



Grundlagen der Rechnerarchitektur

[CS3100.010]

Wintersemester 2014/15

Heiko Falk

Institut für Eingebettete Systeme/Echtzeitsysteme
Ingenieurwissenschaften und Informatik
Universität Ulm



Kapitel 4

Technologische Grundlagen

Inhalte der Vorlesung

1. Einführung
2. Kombinatorische Logik
3. Sequentielle Logik
- 4. Technologische Grundlagen**
5. Rechnerarithmetik
6. Grundlagen der Rechnerarchitektur
7. Speicher-Hardware
8. Ein-/Ausgabe

Inhalte des Kapitels (1)

4. Technologische Grundlagen

- Halbleiter-Bauelemente
 - Halbleiterdiode
 - Halbleiter
 - Digitale Diodenschaltungen
 - Transistor
 - Aufbau
 - Digitale Transistorschaltungen
 - TTL
 - MOS-Feldeffekttransistor
 - Aufbau
 - CMOS-Schaltungen
- ...

Inhalte des Kapitels (2)

4. Technologische Grundlagen

- ...
- Programmierbare Logikbausteine
 - *Tri-State*-Puffer
 - *Wired-OR* / *Wired-AND*-Schaltung
 - ROM / PROM
 - EPROM / EEPROM / Flash
 - PAL
 - GAL
 - PLA
 - CPLD
 - FPGA

Einordnung

Bisher

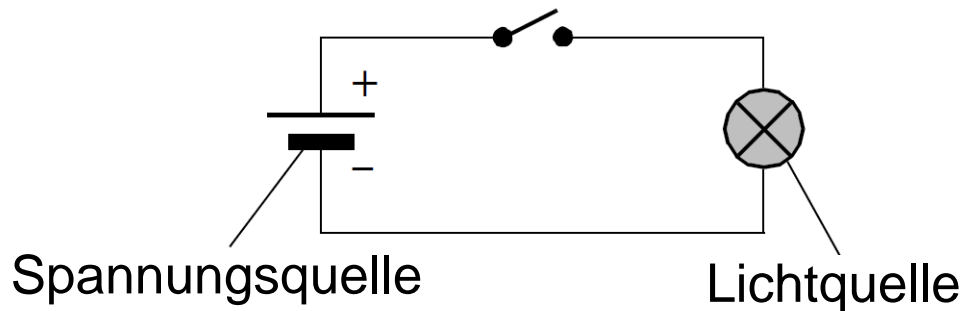
- Schaltungen auf logischer Ebene
- Implementierung vernachlässigt

Jetzt

- Einblick in die technische Realisierung elektronischer Schaltungen

Halbleiterdiode (1)

Grundlage: elektrische Schaltungen



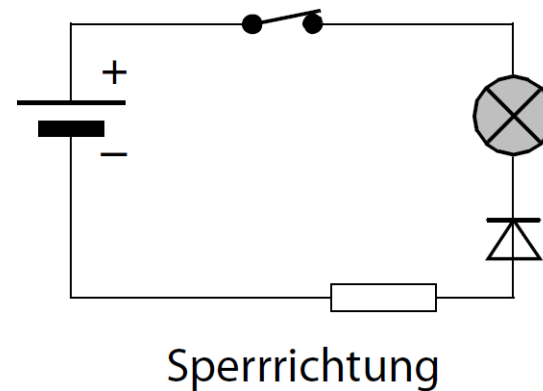
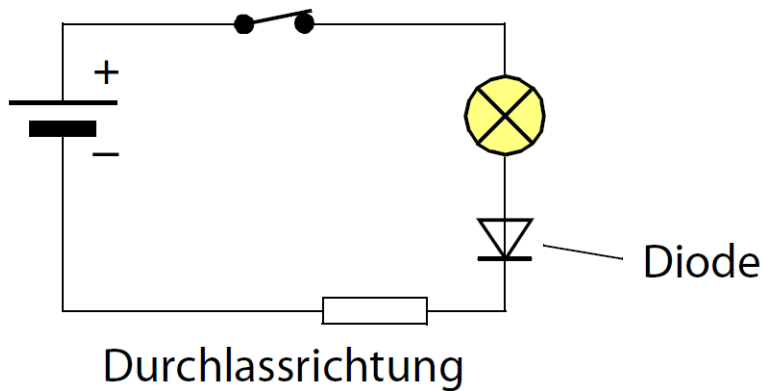
☞ Strom

- Fließt bei geschlossenem Stromkreis
- Konzeptionell:
 - Strom fließt von „Plus“ nach „Minus“
- Physikalisch:
 - Elektronen fließen von „Minus“ nach „Plus“

Halbleiterdiode (2)

Diode

- Spezielles Bauteil mit zwei Anschlüssen



- Strom kann nur in eine Richtung durch die Diode fließen

Aufbau früher

- Röhrendiode
 - Glaskolben mit Vakuum

Halbleiterdiode (3)

Aufbau heute

- Halbleiterdiode

Leiter

- Freie Elektronen können fließen

Isolator

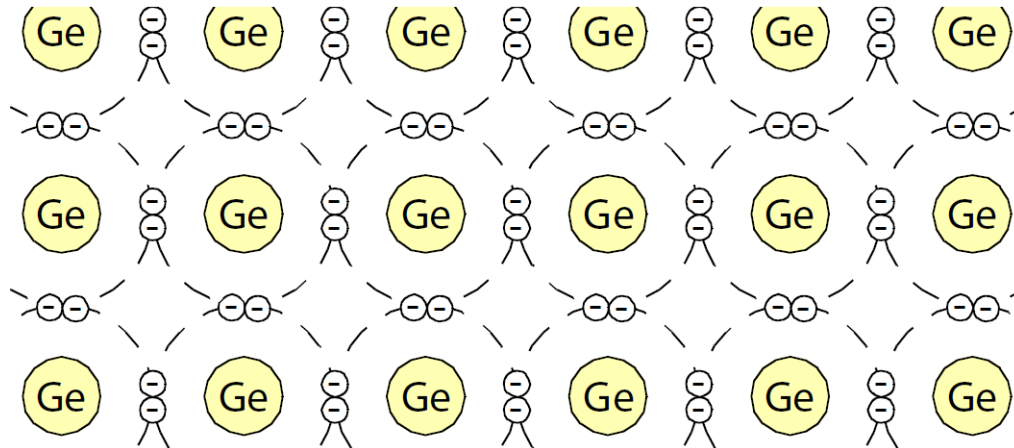
- Keine freien Elektronen

Halbleiter

- Schicht zwischen Leiter und Isolator
- Material, das mal leitend, mal isolierend sein kann
- Z.B. Germanium (Ge) oder Silizium (Si)

Halbleiter (1)

Elektronenanordnung bei Halbleiterkristallen

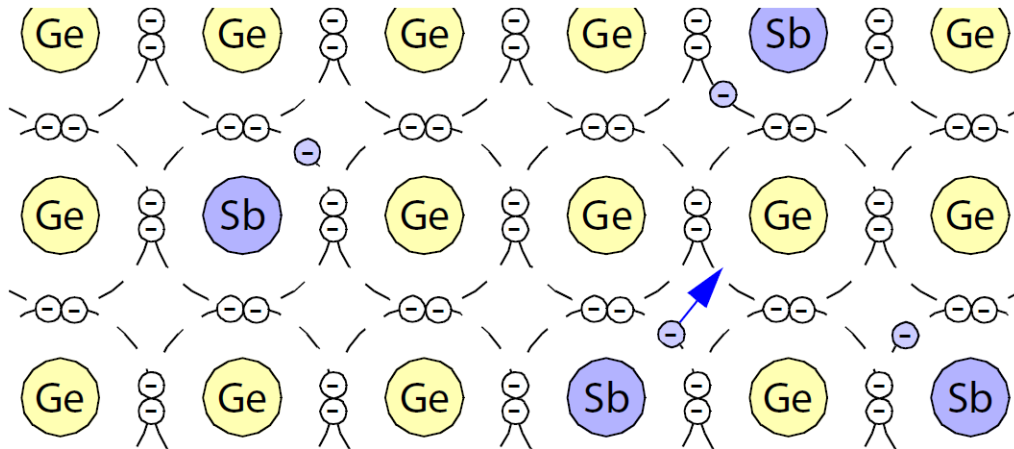


- Je vier Elektronen auf äußerster Schale
- Stabiler Zustand durch Verzahnung der Schalen benachbarter Atome
- Gelegentliche Verunreinigungen
 - Ein Elektron zu viel oder zu wenig auf der äußeren Bahn
 - Geringer Stromfluss möglich (Halbleiter)

Halbleiter (2)

Dotieren der Halbleiter mit anderen Materialien

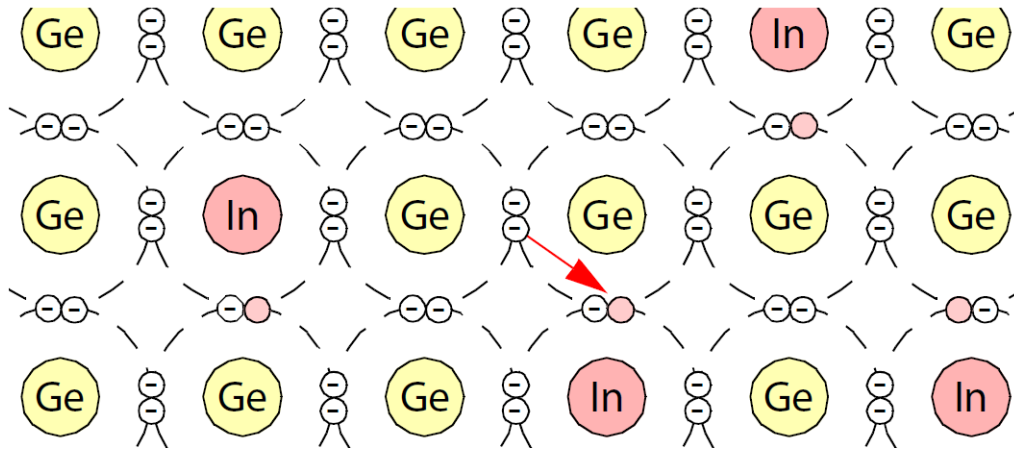
- Dotieren = Einbringen gezielter Verunreinigungen
- Beispiel: Antimon (Sb)



- Ein Elektron mehr auf der äußeren Schale
- ☞ Elektronenüberschuss
- ☞ **N-Leitfähigkeit** des Kristalls (negativ)

Halbleiter (3)

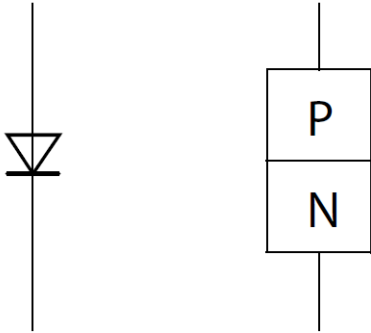
– Beispiel: Indium (In)



- Ein Elektron weniger auf der äußeren Schale
- ☞ Elektronenmangel (dargestellt durch Loch auf der äußeren Schale)
- ☞ **P-Leitfähigkeit** des Kristalls (positiv)

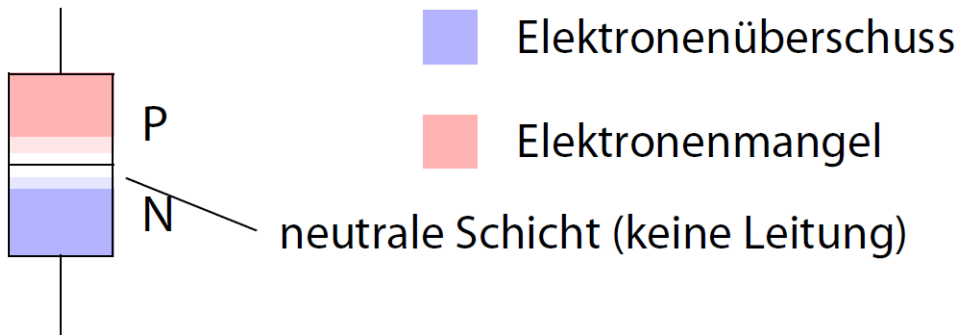
Aufbau der Halbleiterdiode (1)

Aufbau mit N- und P-leitfähigem Bereich



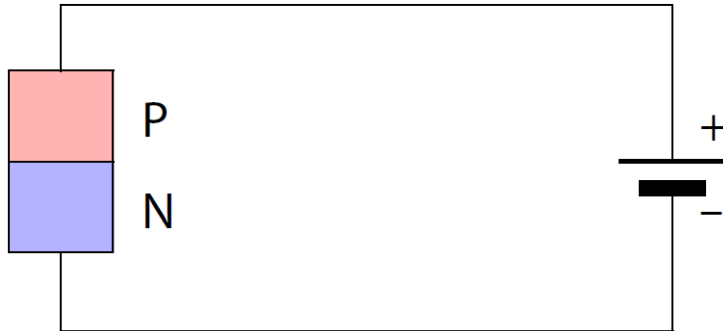
Grenzbereich (PN-Übergang)

- Freie Elektronen füllen Löcher



Aufbau der Halbleiterdiode (2)

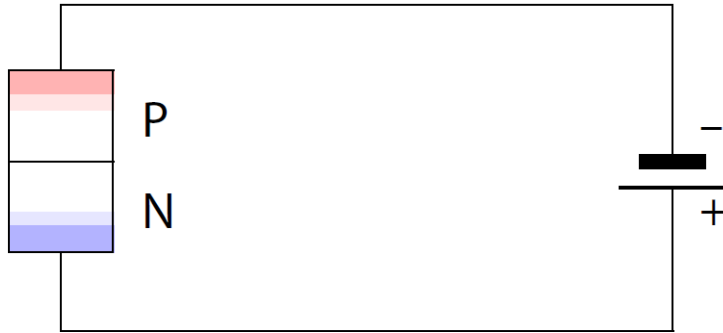
Durchlassrichtung



- Elektronen der Spannungsquelle drücken in den N-Bereich
- Spannungsquelle zieht Elektronen aus dem P-Bereich ab
- ☞ PN-Übergang wird kleiner
- ☞ Strom kann fließen

Aufbau der Halbleiterdiode (3)

Sperrrichtung



- Elektronen der Spannungsquelle drücken in den P-Bereich und füllen Löcher auf
- Spannungsquelle zieht Elektronen aus dem N-Bereich ab
- ☞ PN-Übergang wird größer
- ☞ Strom kann nicht fließen

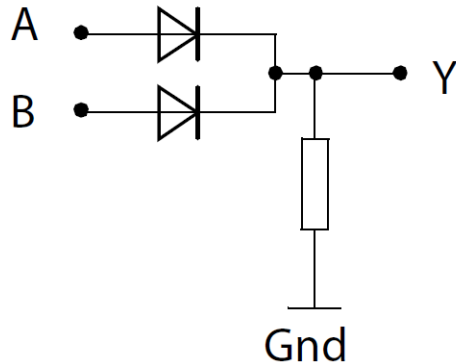
Digitale Diodenschaltungen

Abbildung der Wahrheitswerte (positive Logik)

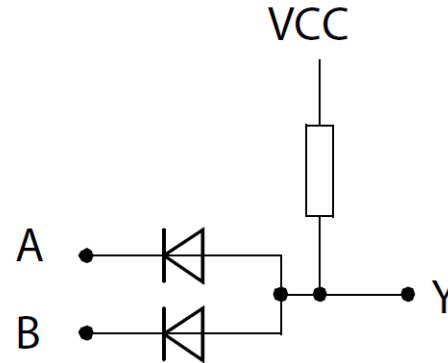
- 1: Stromfluss / positive Spannung
- 0: kein Stromfluss / keine Spannung

Aufbau einfacher Gatter

$$Y = A + B$$

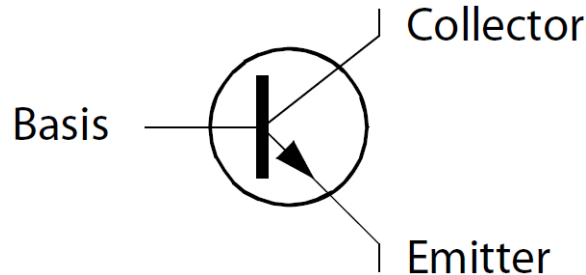


$$Y = A \cdot B$$



Transistor (1)

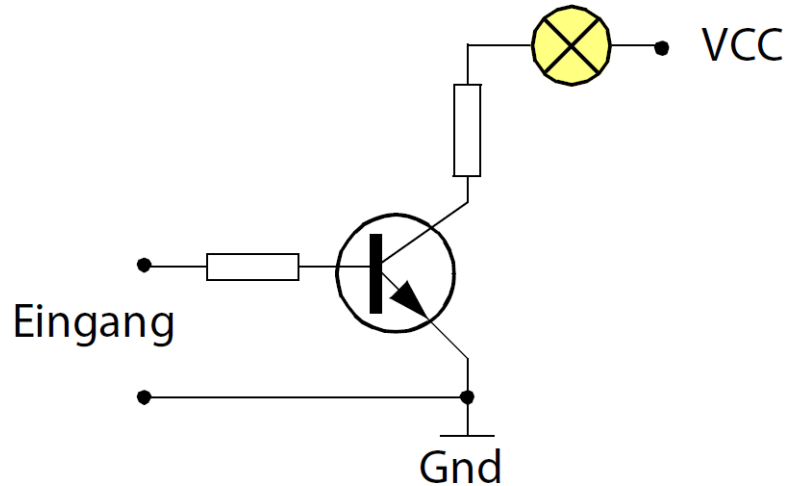
Halbleiterbauteil mit drei Anschlüssen (bipolare Transistoren)



- Zwischen Basis und Emitter sowie zwischen Basis und Kollektor wirkt Transistor wie eine Diode
- Zwischen Emitter und Kollektor fließt zunächst kein Strom
- Durch geringen Strom an der Basis wird Transistor zwischen Kollektor und Emitter leitend

Transistor (2)

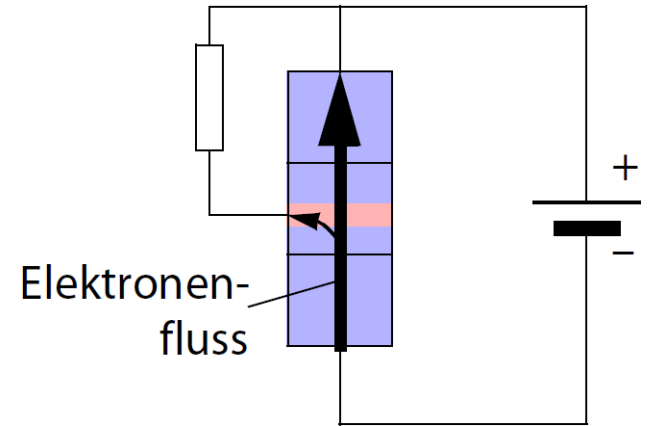
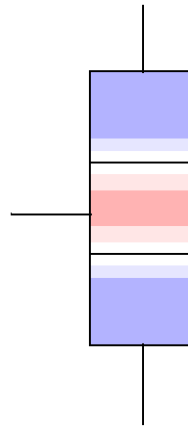
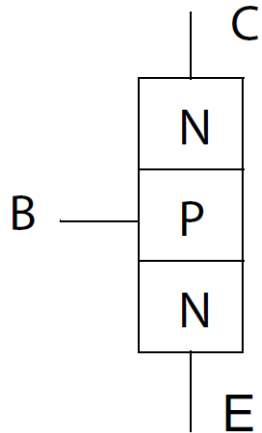
Transistor als Schalter und Verstärker



- Kleiner Schaltstrom an der Basis
- Großer Ausgangsstrom zwischen Kollektor und Emitter
- Verstärkung zwischen Basis- und Kollektor-Kreis

Transistor (3)

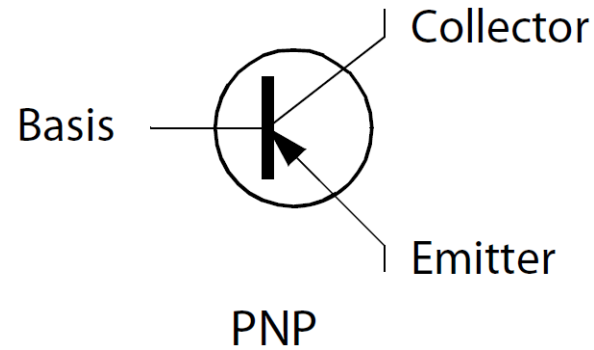
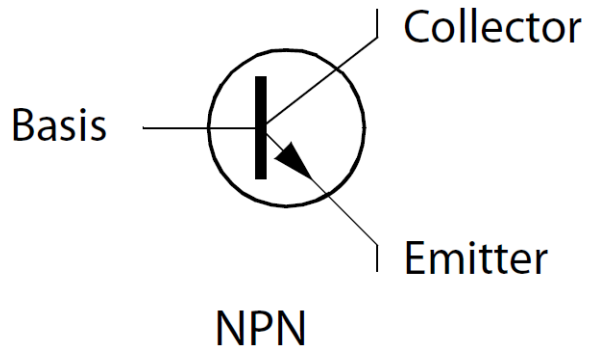
Interner Aufbau



- Drei Schichten: NPN
- Ruhezustand: PN-Übergänge
- Geringer Basisstrom
 - Verringerung des PN-Übergangs zwischen Emitter und Basis
 - Verringerung des PN-Übergangs zwischen Basis und Kollektor

Transistor (4)

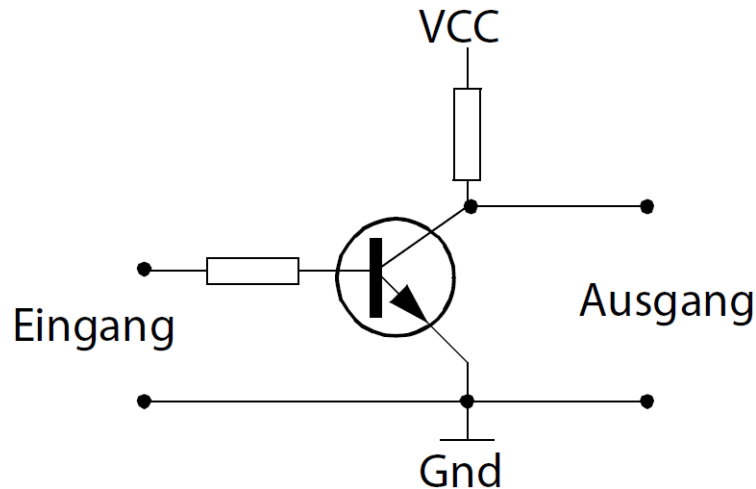
NPN- und PNP-Transistoren



- Prinzipiell gleiche Funktionsweise
- Verschiedene Polung

Digitale Transistorschaltungen (1)

Einfacher Inverter

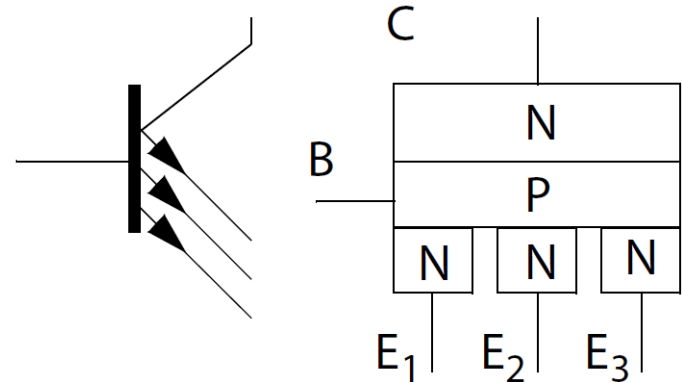


- Liegt Masse (0) am Eingang an, sperrt Transistor
 - Am Ausgang liegt (fast vollständige) Versorgungsspannung (1)
- Liegt Versorgungsspannung (1) am Eingang, schaltet Transistor durch
 - Am Ausgang liegt nur geringe Spannung an (0)

Digitale Transistorschaltungen (2)

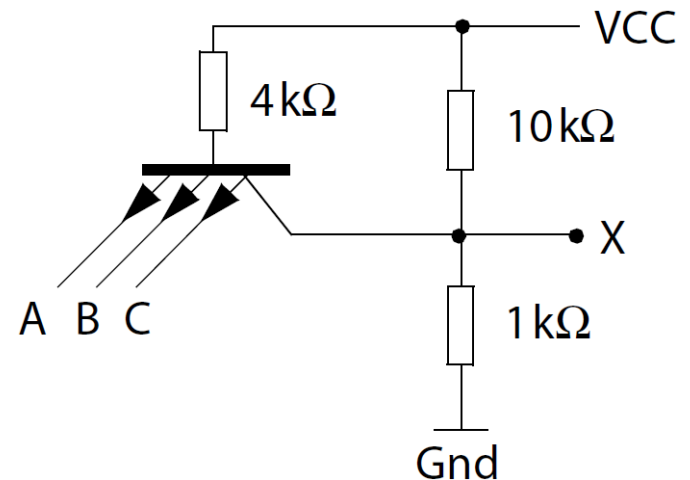
Multi-Emitter-Transistoren

- Je eine Basis- und Kollektor-Zone
- Mehrere Emitterzonen



AND-Gatter mit Multi-Emitter-Transistor

- Transistor schaltet stets durch
- Einer von A, B, C auf Masse
 - ☞ X hat nur geringe Spannung
- Alle A, B, C auf VCC
 - ☞ X hat hohe Spannung



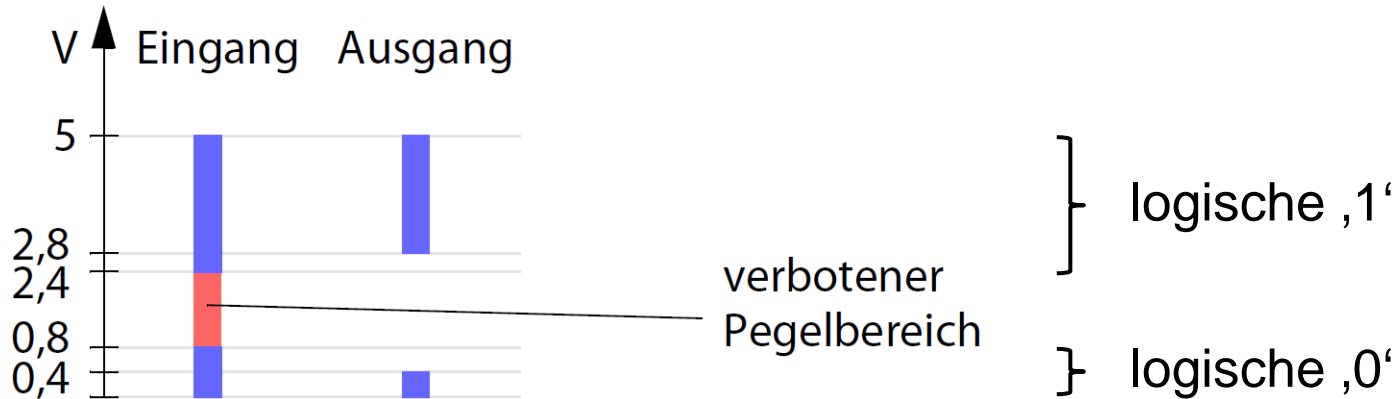
TTL (1)

Transistor-Transistor-Logik (TTL)

- Gatteraufbau ausschließlich mit bipolaren Transistoren

Pegelbereiche

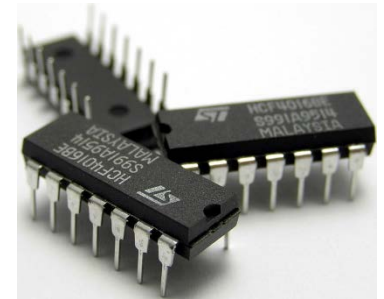
- Darstellung der logischen Werte ,0‘ und ,1‘ durch Bereiche von zulässigen elektrischen Spannungen



TTL (2)

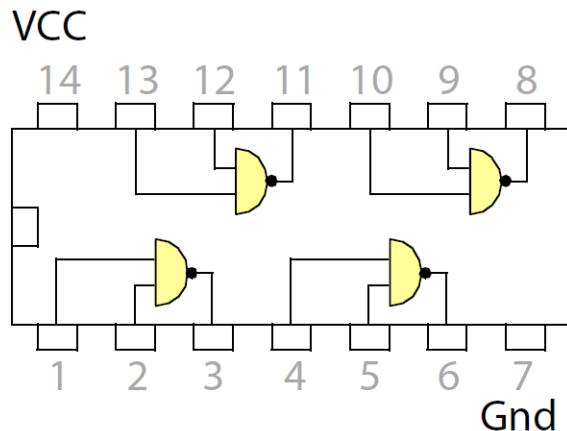
TTL-Schaltkreise

- Meist als *Dual in-Line* Gehäuse (DIL) alias *Dual in-Line Package* (DIP)
- *Pins* eines Bausteins sind zweireihig auf beiden Seiten des Gehäuses angeordnet



[de.wikipedia.org]

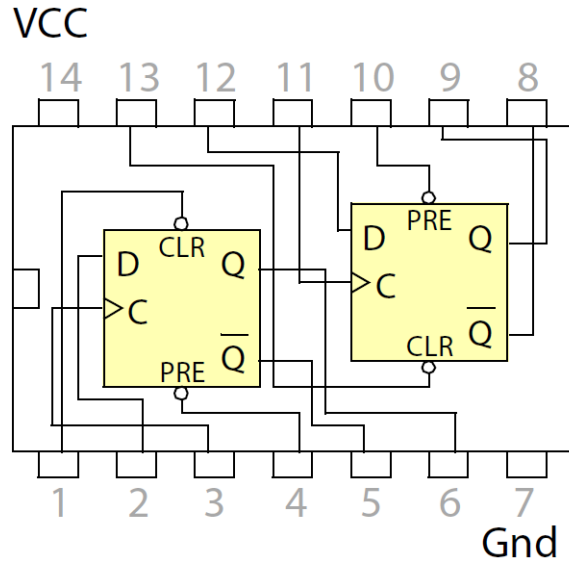
Beispiel: Baustein 7400



Vier NAND-Gatter mit je zwei Eingängen

TTL (3)

Beispiel: Baustein 74LS74



Zwei positiv-flankengetriggerte
D-Flip-Flops

- CLR = explizites asynchr. Rücksetzen
- PRE = explizites asynchr. Setzen

TTL (4)

TTL-Familien

	TTL	LS-TTL	ALS-TTL	F-TTL	AS-TTL
Bezeichnung	74xx	74LSxx	74ALSxx	74Fxx	74ASxx
Spannung	5 V				
Leistung pro Gatter	10 mW	2 mW	1 mW	4 mW	22 mW
Schaltzeit	10 ns	9 ns	4 ns	2,5 ns	1,5 ns
max. Frequenz	40 MHz	50 MHz	100 MHz	125 MHz	230 MHz

Problem

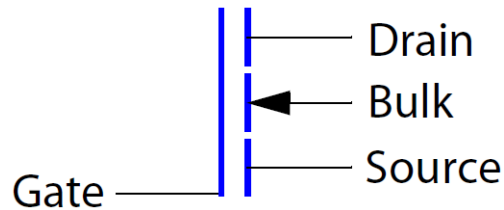
- Erhöhung der Schaltgeschwindigkeit führt zu höherer Leistungsaufnahme
- Spezielle „*low power*“ Transistoren und Schaltungen im Einsatz

MOS-Feldeffekttransistor (1)

MOS-FET (*Metal-Oxide-Semiconductor*, Metall-Oxid-Halbleiter)

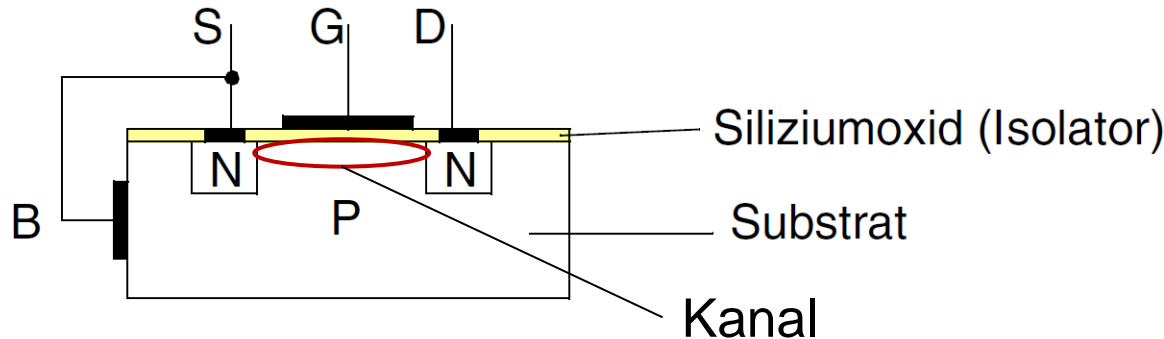
Elektrisches Feld steuert Leitfähigkeit im Halbleiterkristall

- Schaltbild



Selbstsperrender n-Kanal MOS-FET

- *Bulk* (Substrat) üblicherweise mit *Source* verbunden
- Aufbau



MOS-Feldeffekttransistor (2)

Funktionsweise

- Im Ruhezustand ist Transistor gesperrt: zwischen *Drain* und *Source* hoher Widerstand, kein Stromfluss (selbstsperrend)
- Positive Spannung zwischen Steuerleitung *Gate* und *Source*
 - *Gate*, Isolator und Substrat wirken wie ein Kondensator, der aufgeladen wird
 - Diese Aufladung führt zu einem elektrischen Feld im Substrat
 - Durch das elektrische Feld wandern Elektronen innerhalb des Substrats in den NP-Grenzbereich und sammeln sich dort an
 - Ab einer gewissen Schwellenspannung am *Gate* befinden sich so viele Elektronen im Kanal des Transistors, dass der Kanal N-leitfähig wird
- ☞ Stromfluss zwischen *Source* und *Drain* möglich

MOS-Feldeffekttransistor (3)

Vorteil

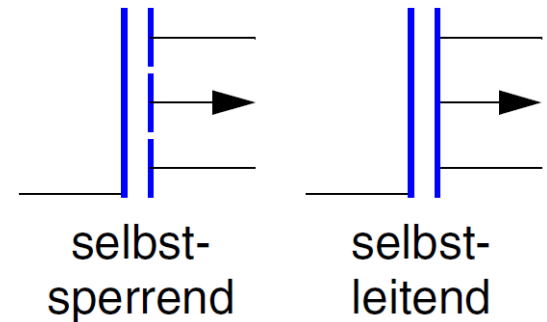
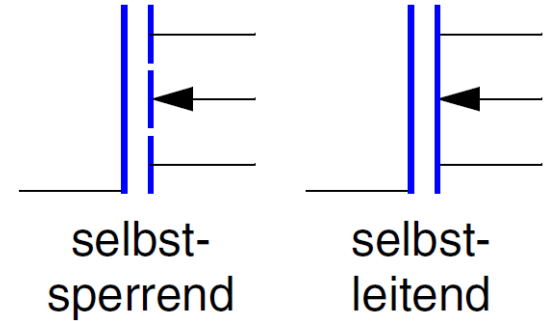
- Elektrisches Feld benötigt keinen Stromfluss (nur Steuerspannung)
- Leistungsloser Betrieb
- Lediglich Umschalten erfordert Wechsel der Ladungszustände (kostet Energie)
- Leistungsaufnahme von Umschaltfrequenz abhängig

MOS-Feldeffekttransistor (4)

MOS-FET Varianten

- n-Kanal MOS-FET
 - Selbstsperrend (*Gate-Source-Spannung* schaltet durch)
 - Selbstleitend (*Gate-Source-Spannung* sperrt)

- p-Kanal MOS-FET
 - Selbstsperrend
 - Selbstleitend

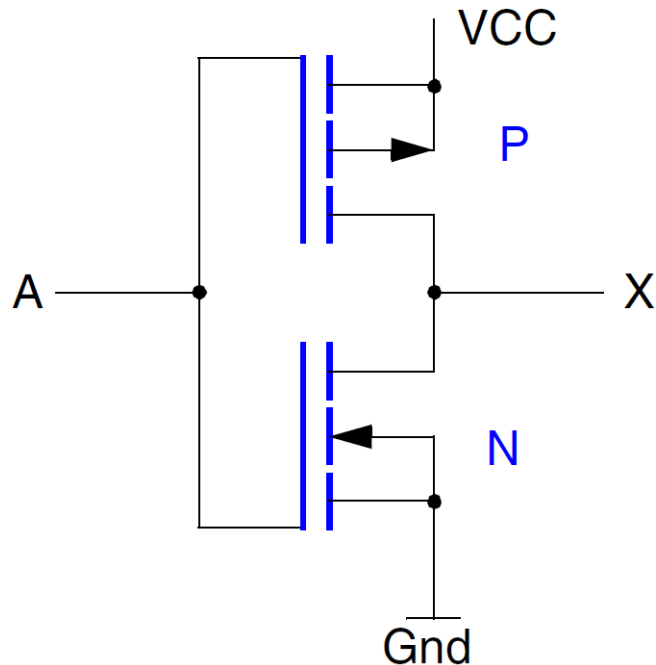


CMOS-Schaltung (1)

CMOS (*Complementary-symmetry Metal-Oxide-Semiconductor*)

- Komplementär-symmetrischer MOS-Halbleiter
- Bei CMOS werden üblicherweise komplementäre MOS-FETs (d.h. p- und n-Kanal MOS-FETs) symmetrisch miteinander verschaltet

Beispiel: NICHT-Gatter

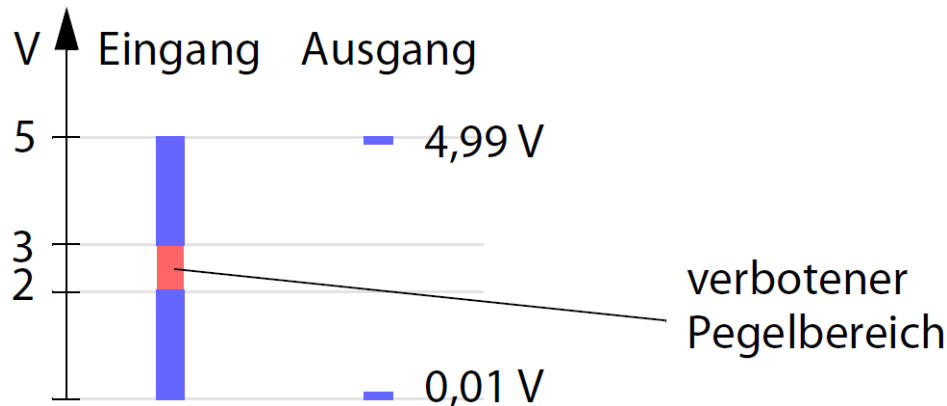


- Einer der Transistoren ist immer gesperrt
- Niedrige Leistungsaufnahme
- Kaum Stromfluss durch beide Transistoren

CMOS-Schaltung (2)

Versorgungsspannung

- Kann in weiten Grenzen schwanken
- TTL-kompatible Pegel möglich
 - Beispiel



Schaltungen empfindlich gegen Überspannungen (Entladungen)

- Schutzschaltungen an den Eingängen integrierter Bausteine

Roter Faden

4. Technologische Grundlagen

- Halbleiter-Bauelemente
 - Halbleiterdiode
 - Halbleiter
 - Digitale Diodenschaltungen
 - Transistor
 - Aufbau
 - Digitale Transistorschaltungen
 - TTL
 - MOS-Feldeffekttransistor
 - Aufbau
 - CMOS-Schaltungen
- Programmierbare Logikbausteine

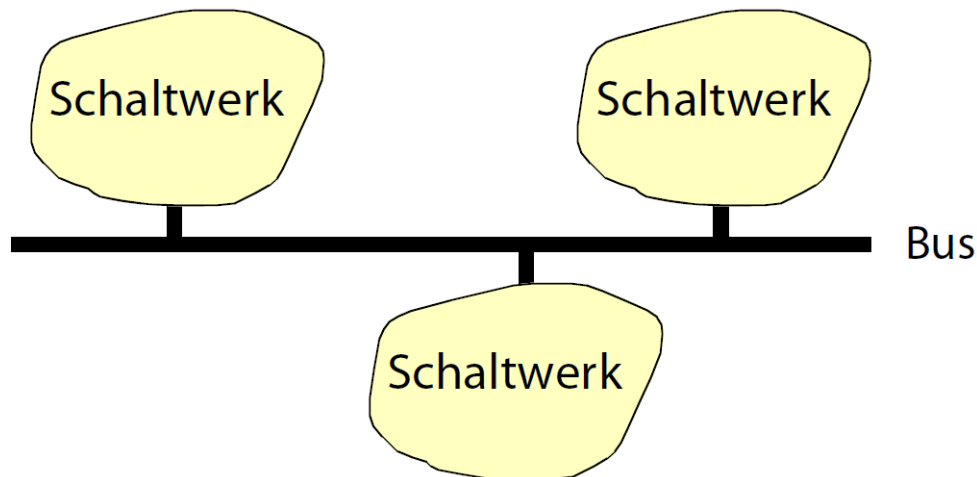
Verknüpfung von Schaltungen (1)

Ausgabeleitungen elektronischer Digitalschaltungen

- 0: geringe Spannung, geringer Stromfluss
- 1: hohe Spannung, hoher Stromfluss

Ausgang soll sich gelegentlich elektrisch neutral verhalten

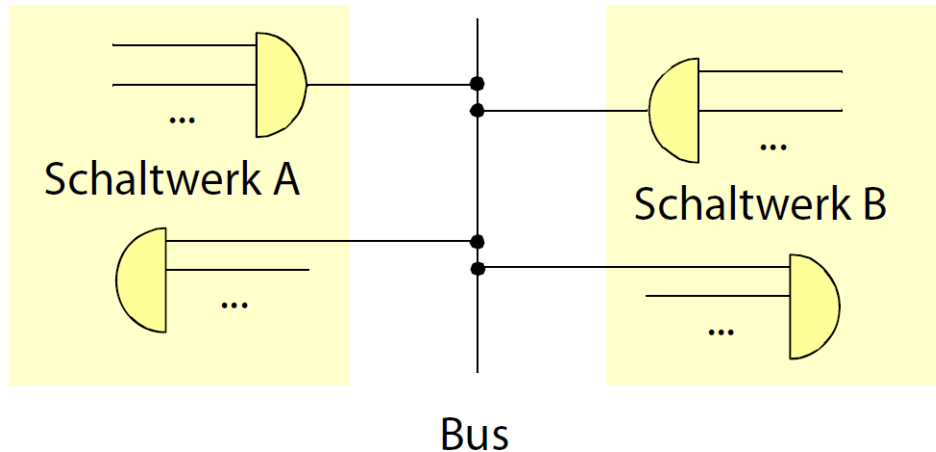
- Beispiel: Bussystem verbindet Schaltwerke innerhalb eines Rechners



- Busleitungen sollen zur Eingabe und zur Ausgabe benutzt werden

Verknüpfung von Schaltungen (2)

Situation

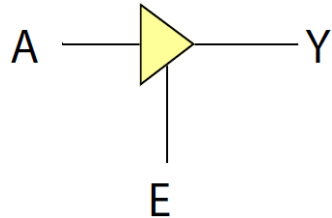


Problem: Zwei Ausgänge auf der selben Leitung

- Elektrisch: Kurzschluss je nach Ausgabewert, bzw. Spannungen im verbotenen Bereich

Tri-State-Ausgabelogik (1)

Lösung: Tri-State-Puffer

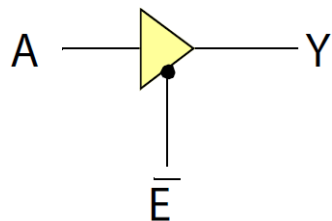


A	E	Y
0	0	X
0	1	0
1	0	X
1	1	1

X = Ausgabe elektrisch neutral
(floating, hochohmig)

E = enable

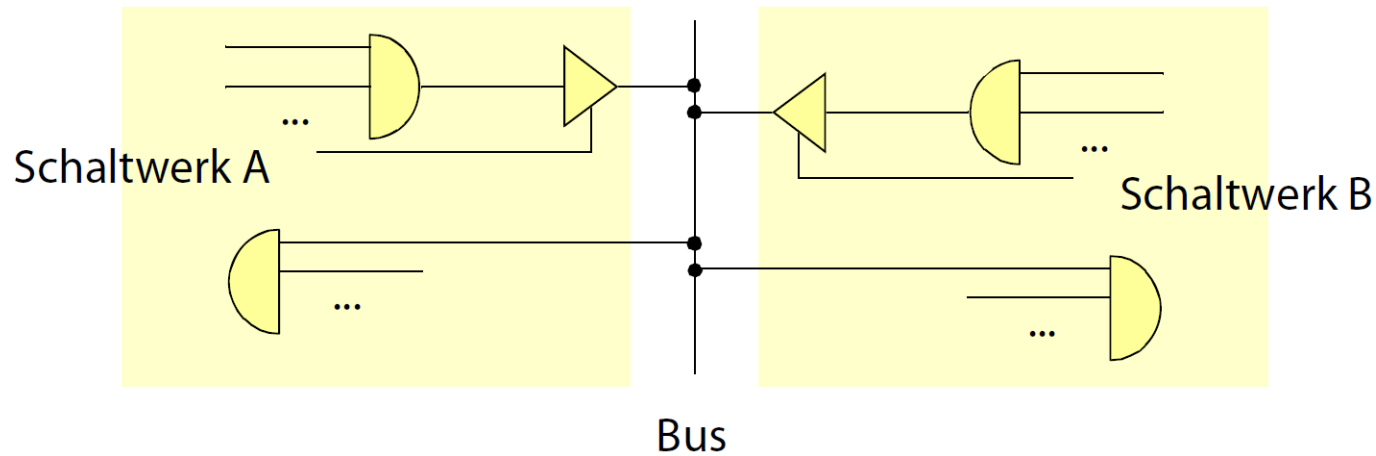
– Tri-State-Puffer auch mit invertiertem E-Eingang gebräuchlich



A	E	Y
0	0	0
0	1	X
1	0	1
1	1	X

Tri-State-Ausgabelogik (2)

Zugang zum Bus nur über *Tri-State*-Puffer

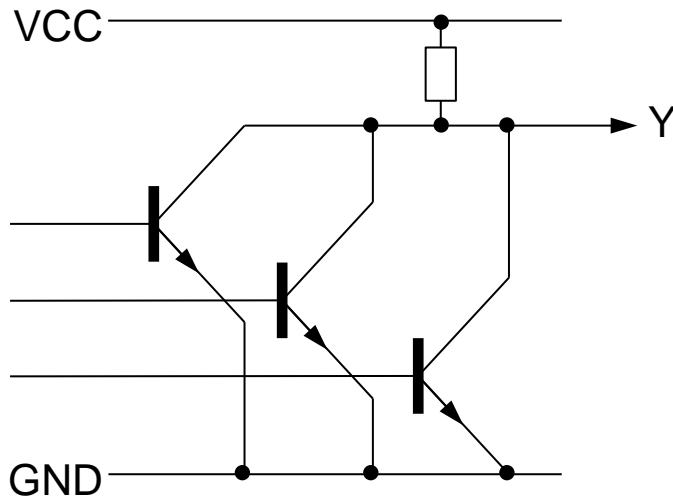


- Schaltwerke können Ausgang jeweils vom Bus nehmen
- Bestimmter Datenpfad kann gezielt ausgewählt werden

Wired-And / Wired-Or-Schaltung (1)

Alternative

- Ausnutzen der elektrotechnischen Grundlagen
- Kollektor eines Transistors als Ausgang (*open collector*-Schaltung)



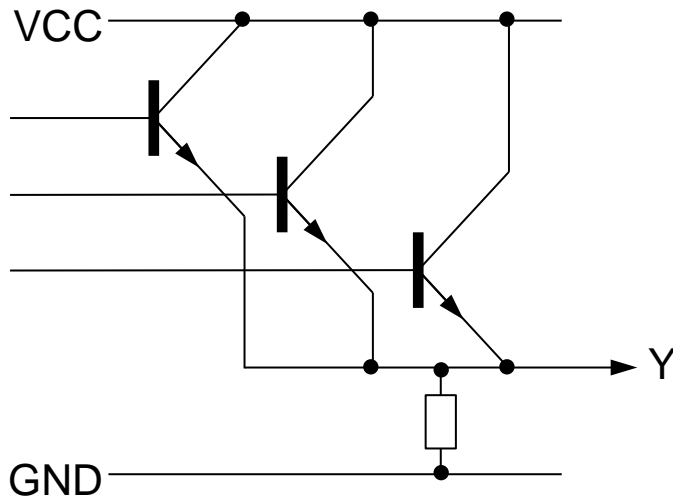
Wired-And

- Selbstleitende Transistoren
- ‚0‘ an einer Basis lässt einen Transistor leiten
- ☞ $Y = ,0‘$
- ‚1‘ an jeder Basis sperrt alle Transistoren
- ☞ $Y = ,1‘$

Wired-And / Wired-Or-Schaltung (2)

Alternative

- Ausnutzen der elektrotechnischen Grundlagen
- Kollektor eines Transistors als Ausgang (*open collector*-Schaltung)



Wired-Or

- „Umdrehen“ eines *Wired-And*
- Selbstsperrende Transistoren
- ‚1‘ an einer Basis lässt einen Transistor leiten
- ☞ $Y = ,1‘$
- ‚0‘ an jeder Basis sperrt alle Transistoren
- ☞ $Y = ,0‘$

ROM / PROM (1)

Read-Only Memory (ROM)

- Festwertspeicher
- Werte bei der Herstellung festgelegt

Programmable Read-Only Memory (PROM)

- Festwertspeicher
- Einmalig durch Anwender programmierbar

Simple Programmable Logic Devices (SPLD)

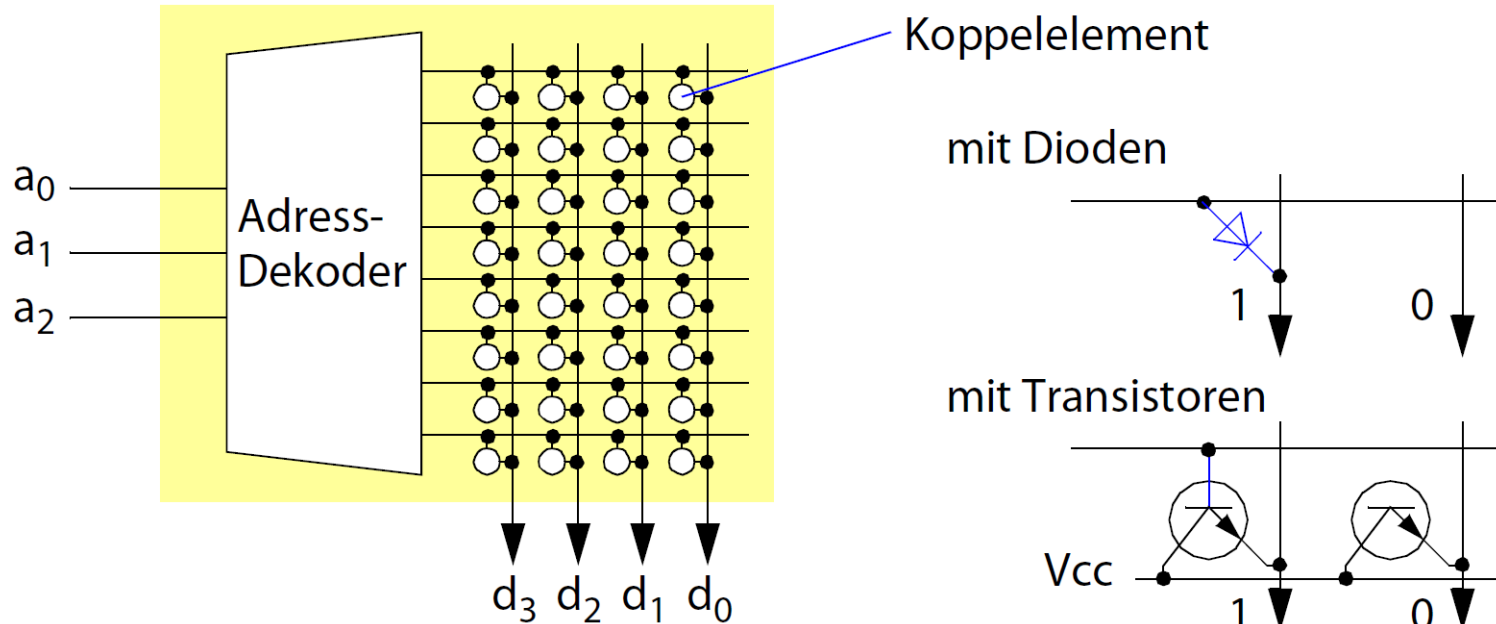
- ROM und PROM gehören beide zur Familie der SPLDs
- Einfache programmierbare Logikbausteine

ROM / PROM (2)

Funktionsweise

- Adressleitungen selektieren Speicherzelle
- Inhalt an Ausgabeleitung abgreifbar

Beispiel: 8 x 4 Bit ROM



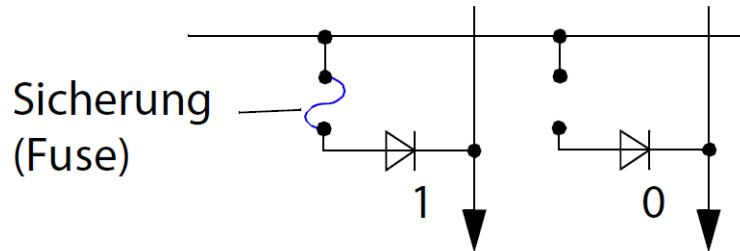
ROM / PROM (3)

Beispiel: 8 x 4 Bit ROM

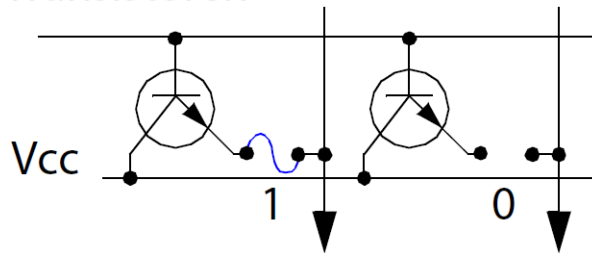
- Einsatz von Sicherungen
- Programmiervorgang zerstört u.U. Sicherungen

☞ Koppellemente für PROMs

mit Dioden



mit Transistoren



EPROM / EEPROM / Flash

Erasable Programmable Read-Only Memory (EPROM)

- Festwertspeicher
- Elektrisch programmierbar (durch Überspannungen)
- Durch UV-Bestrahlung löscher

Electrically Erasable Programmable Read-Only Memory (EEPROM)

- Festwertspeicher
- Elektrisch programmierbar
- Elektrisch löscher

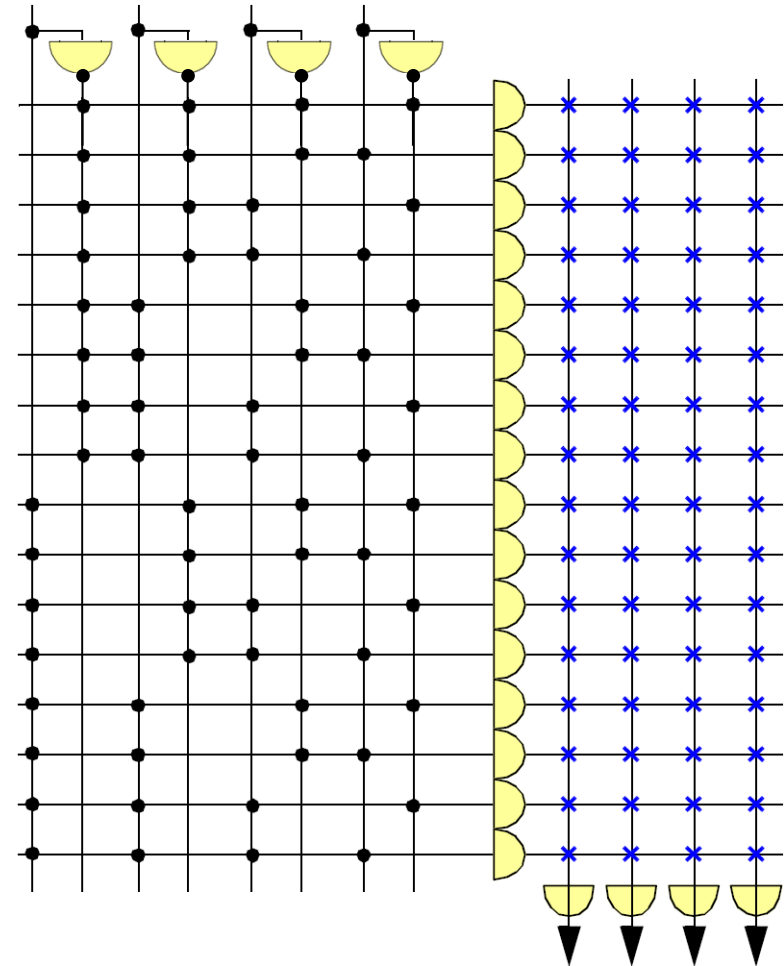
Flash-Speicher

- Festwertspeicher
- Ähnlich EEPROM, aber geringere Größe
- Nur blockweise programmier- und löscher

PROM als Schaltnetz (1)

Schematische Darstellung eines PROMs

- Adressdecodierer als feste UND-Matrix (Minterme über Adressleitungen)
- Koppелеlemente als programmierbare ODER-Matrix



PROM als Schaltnetz (2)

PROM kann Schaltfunktion implementieren

- Wahrheitstabelle in Hardware abgebildet
- PROM mit 2^m Worten à n Bits kann n Schaltfunktionen mit je m Eingängen realisieren
- Keine Minimierung
 - Pro Wertekombination der Eingänge wird ein Ergebnis direkt programmiert

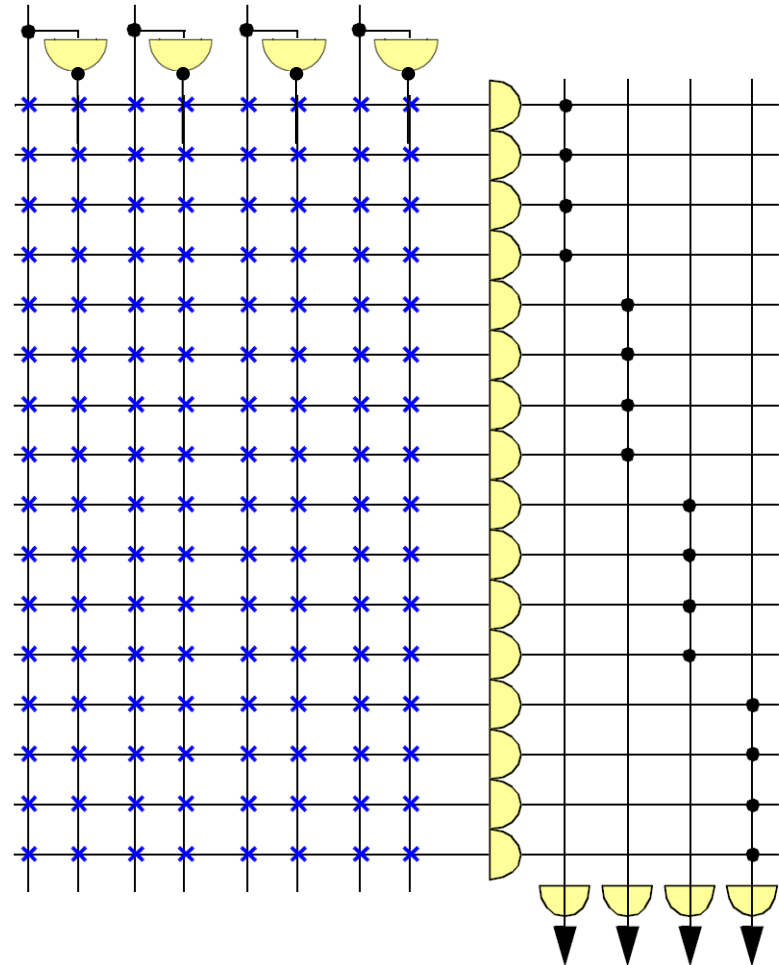
Beispiel: 2x2-Multiplizierer

- Wahrheitstabelle: siehe Kapitel 2, Folie 75
- Werte in PROM „brennen“

PAL (1)

Programmable Array Logic (PAL)

- Frei programmierbare UND-Matrix
- Feste ODER-Matrix
- Anzahl der UND-Gatter pro ODER-Gatter fixiert
- Kann jede minimierte Schaltfunktion realisieren
 - Voraussetzung: Anzahl der Produktterme pro Schaltfunktion klein genug



PAL (2)

Begriff „PAL“

- Eingetragenes Warenzeichen der Firma AMD bzw. Lattice
- PAL \neq „Problem Anderer Leute“ ; -)

Technik

- Ursprünglich in TTL-, später auch in CMOS-Technik
- Ursprünglich einmalig programmierbar
- Heute zum Teil auch löschbare Varianten

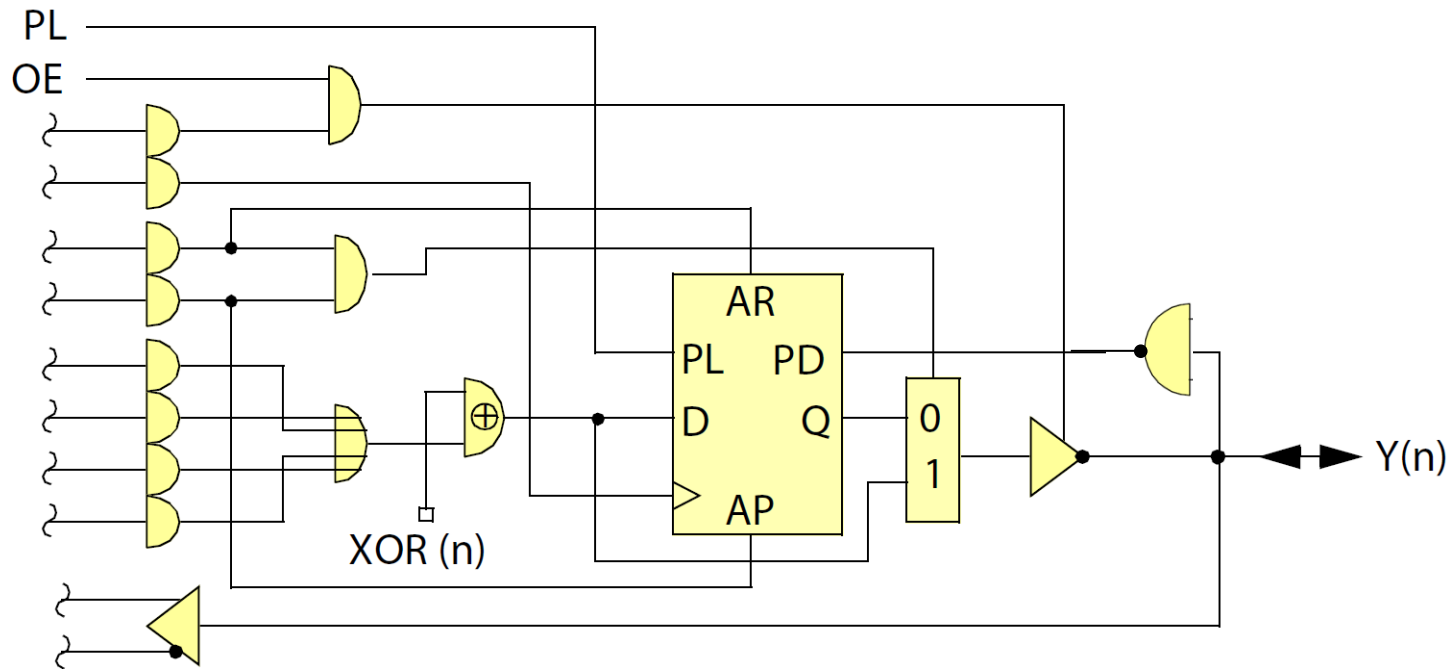
Varianten

- Normaler oder invertierter Ausgang
- Flip-Flops an den Ausgängen
- Getaktete oder ungetaktete Flip-Flops
- Rückkopplung der Ausgänge als interne Eingänge
- *Tri-State*-Ausgänge

GAL (1)

Generic Array Logic (GAL)

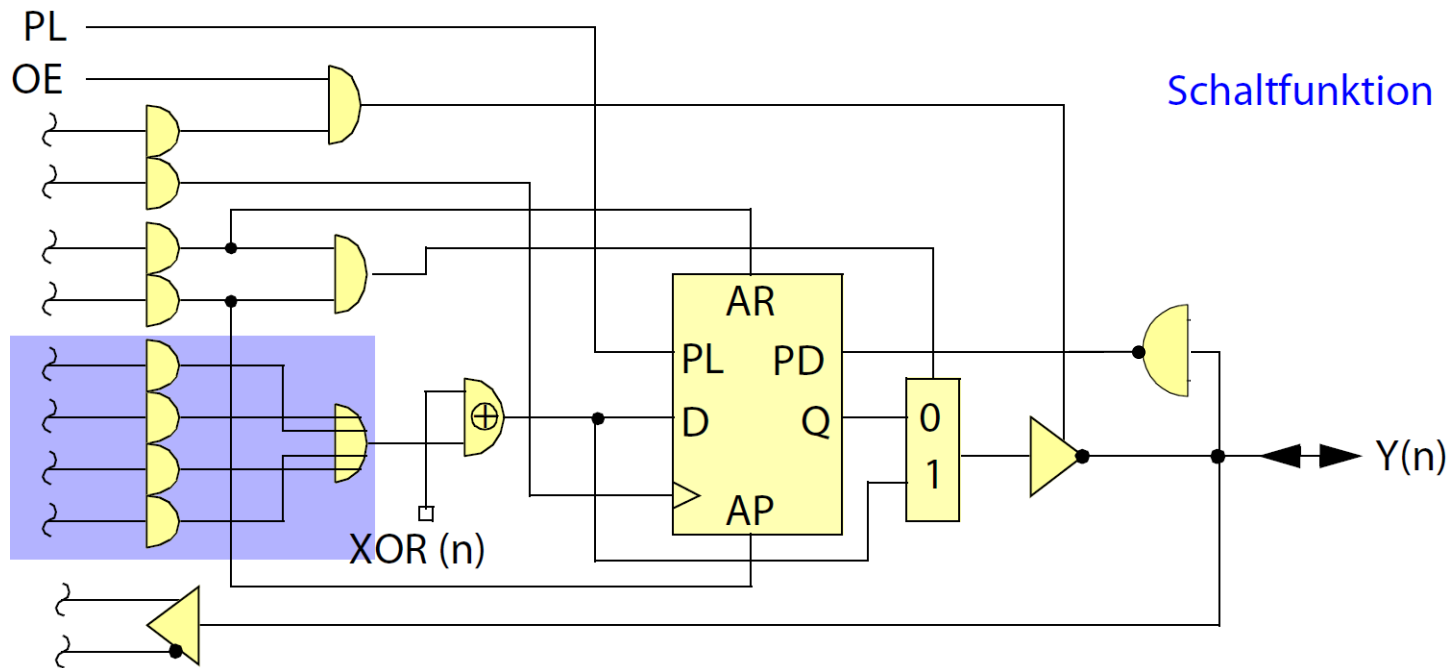
- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (2)

Generic Array Logic (GAL)

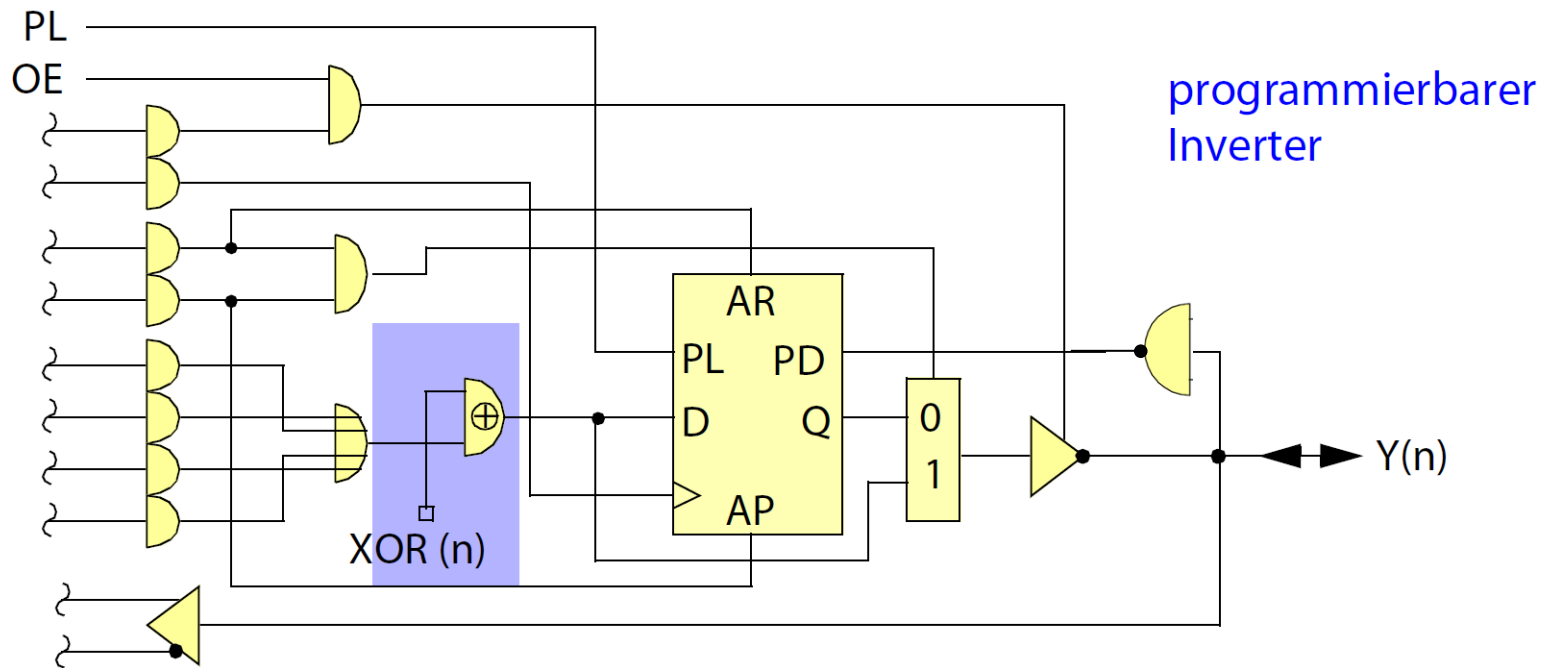
- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (3)

Generic Array Logic (GAL)

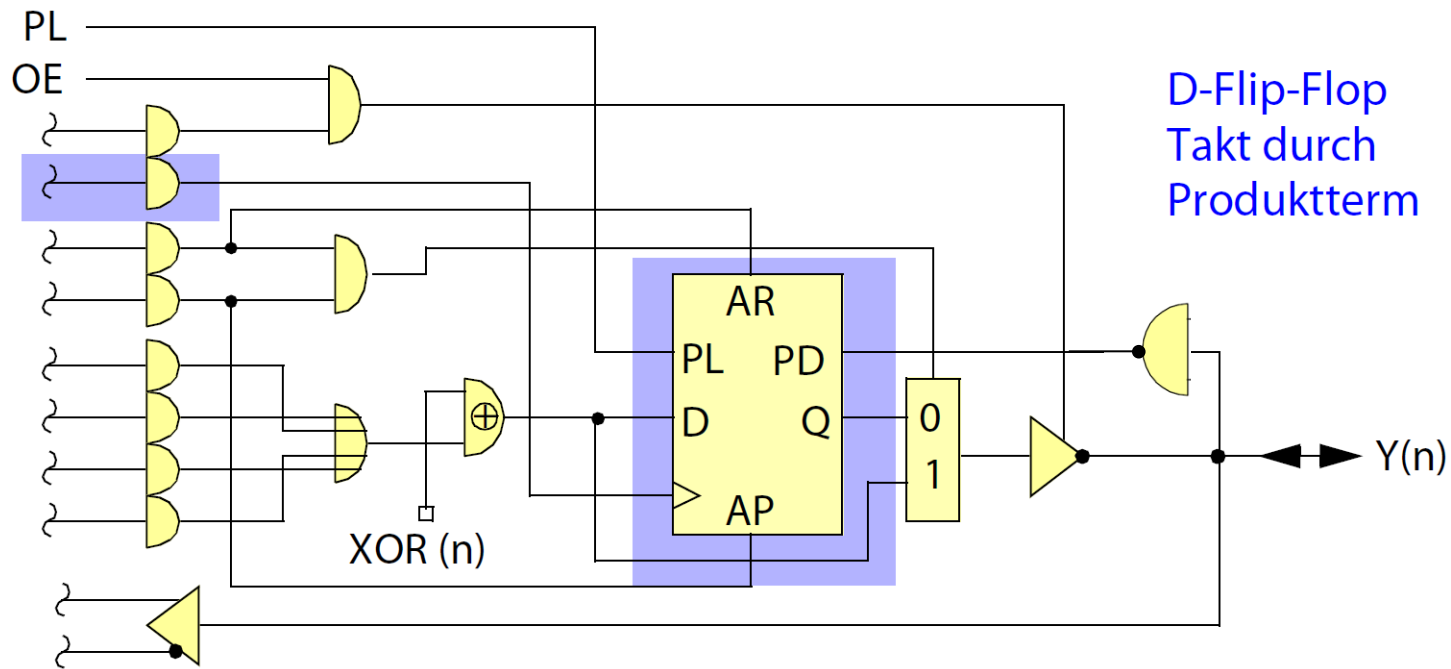
- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (4)

Generic Array Logic (GAL)

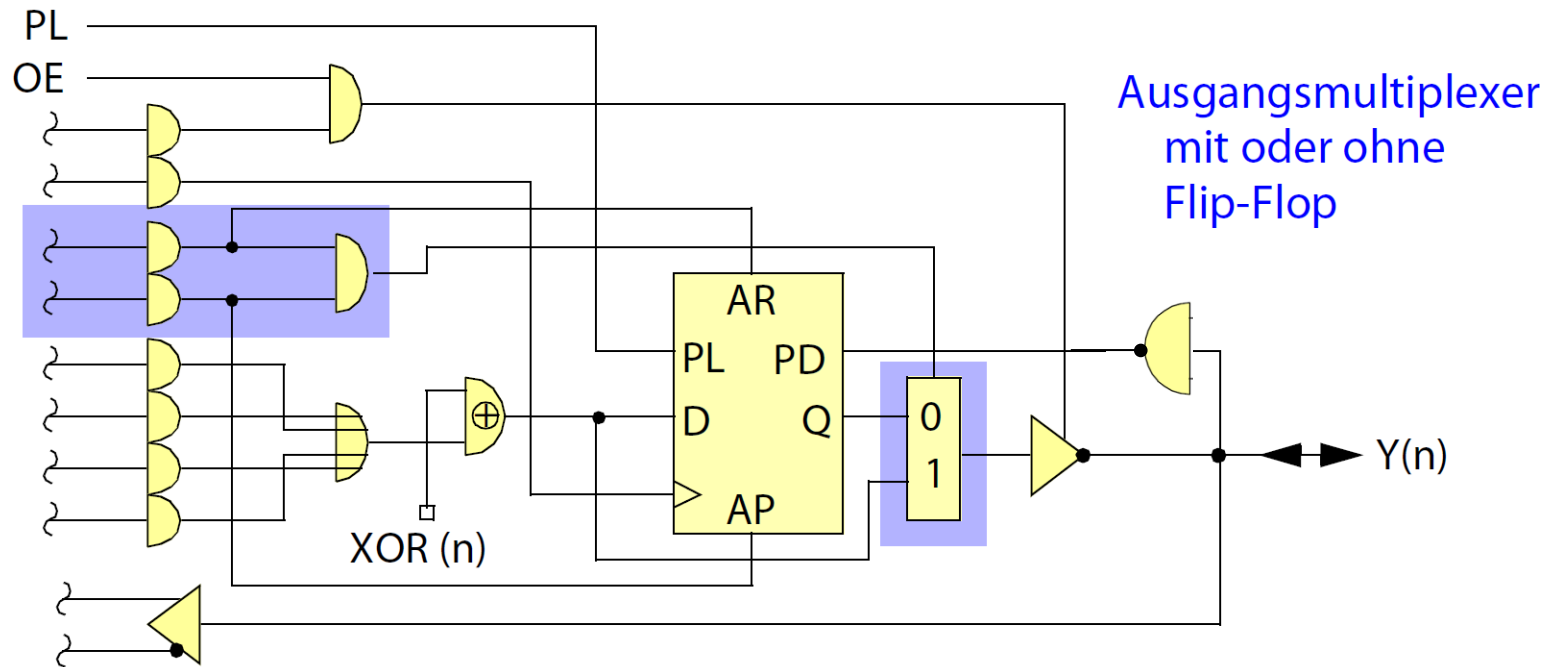
- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (5)

Generic Array Logic (GAL)

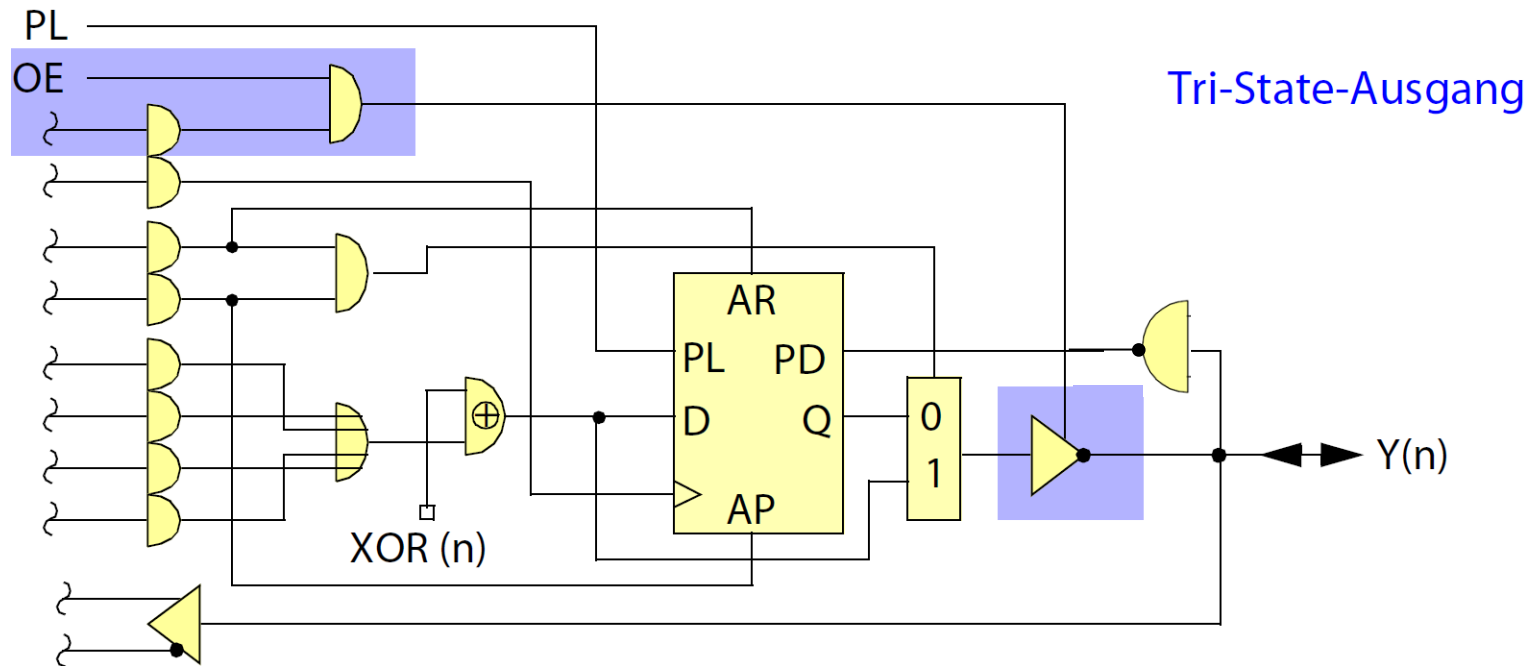
- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (6)

Generic Array Logic (GAL)

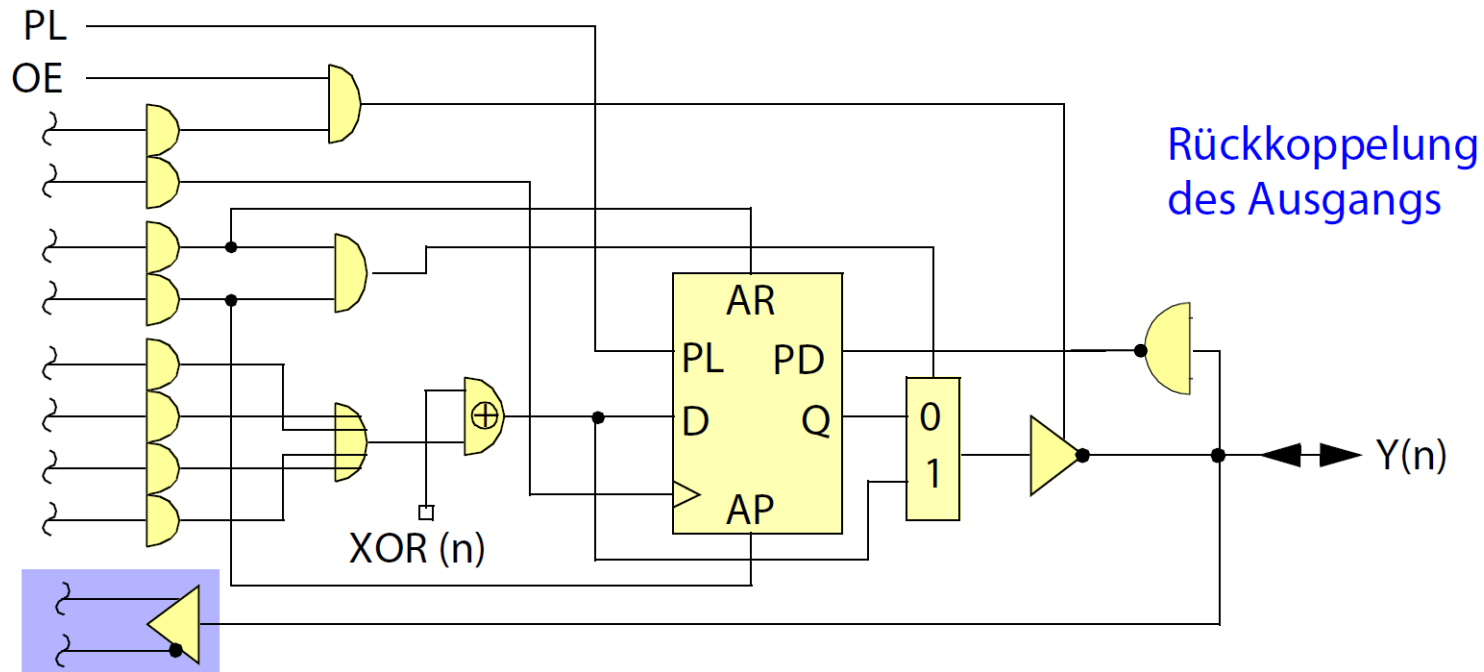
- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (7)

Generic Array Logic (GAL)

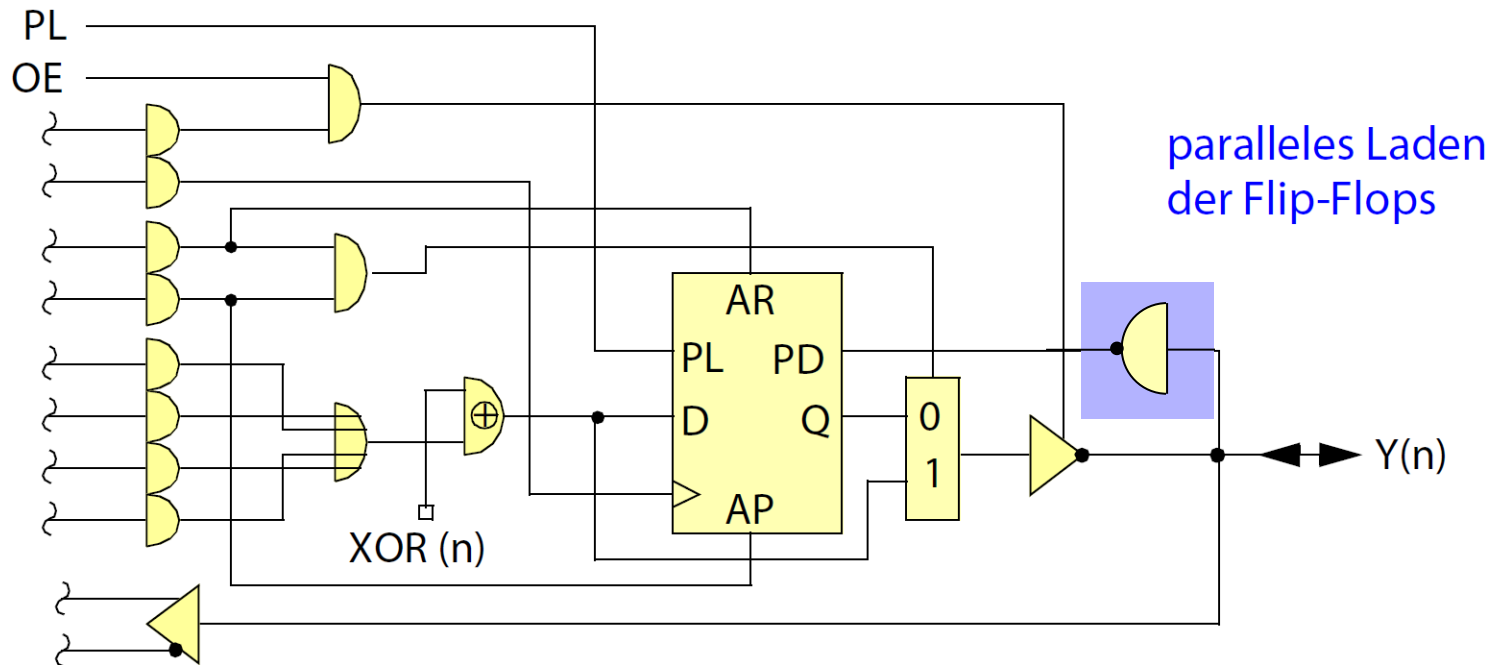
- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (8)

Generic Array Logic (GAL)

- Funktionsweise wie PAL, jedoch flexible Ausgabelogik
- Beispiel: *Output Logic Macro Cell (OLMC)*



GAL (9)

Begriff „GAL“

- Markenzeichen der Firma Lattice

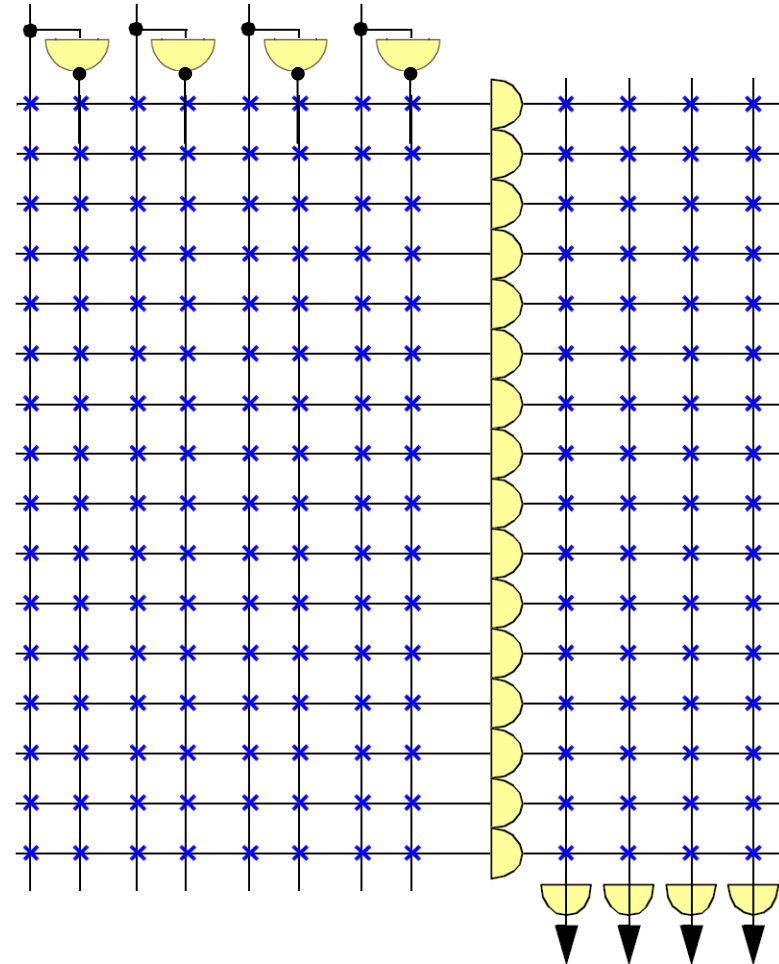
Technik

- Ausschließlich CMOS
- Wiederprogrammierbar durch E²CMOS-Technologie (ähnlich EEPROM)

PLA

Programmable Logic Array (PLA)

- Frei programmierbare UND-Matrix
- Frei programmierbare ODER-Matrix
- Kann beliebige Schaltfunktionen realisieren
 - Voraussetzung: Anzahl der Produktterme ausreichend



CPLD (1)

Complex Programmable Logic Device (CPLD)

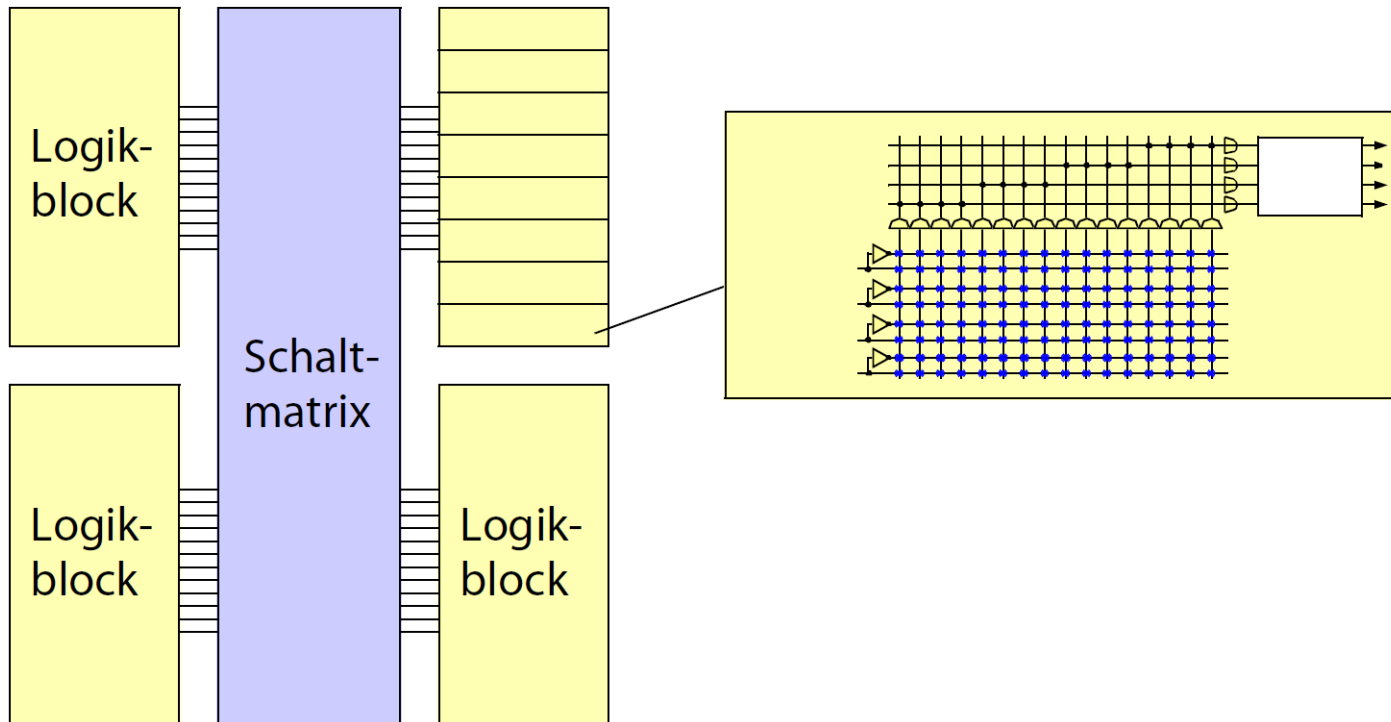
- Alternativbezeichnungen (Markennamen):
EPLD, EEPLD, PEEL, MAX, SuperPAL, MegaPAL, ...

Aufbau

- Zusammenfassung der Funktionalität von bis zu etwa 100 GAL-Bausteinen
 - Makrozellen zur Berechnung der Schaltfunktionen
 - Makrozellen für Ausgabelogik (*I/O-Blocks*)
- Verschaltung der internen GALs mit einer programmierbaren Zuordnungsmatrix (vollständige oder unvollständige Verschaltung)
- „Borgen“ zusätzlicher UND-Glieder für Produkterme aus anderen Makrozellen möglich
- Programmierung wie EPROM, EEPROM oder Flash-ROM

CPLD (2)

Blockschaltbild



Rekonfigurierbare Logik

Motivation

- Speziell angefertigte Hardware (*Application-Specific Integrated Circuit, ASIC*) sehr teuer; Ausführung von „billiger“ Software auf einem Prozessor oft zu langsam
- Wunsch: hohe Geschwindigkeit von Hardware + Flexibilität von Software
 - ☞ Hardware mit programmierbarer Funktionalität und Verschaltung
 - ☞ Einsatz von konfigurierbarer Hardware, im folgenden: FPGAs

Anwendungsgebiete

- Bit-orientierte Algorithmen wie Verschlüsselung, oder
- Schnelle Objekterkennung (z.B. Medizin oder Militär)
- Anpassung von Mobiltelefonen an verschiedene Standards
- *Rapid Prototyping*: schnelle und kostengünstige Erzeugung eines ASIC-Prototypen für Testzwecke (z.B. Straßentests neu entwickelter Kfz)

FPGA (1)

Field-Programmable Gate Array (FPGA)

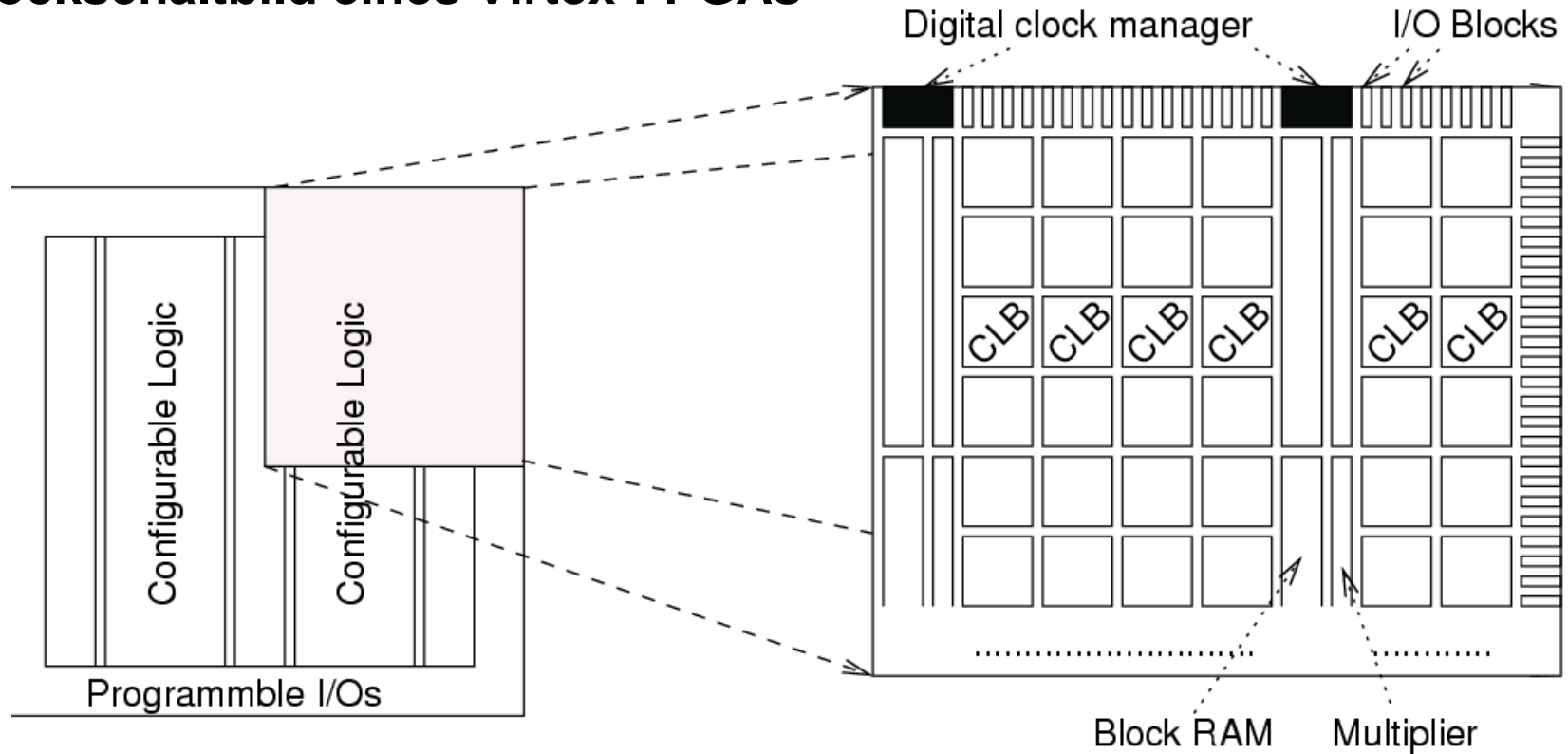
- Alternativbezeichnungen (Markennamen):
LCA, pASIC, FLEX, APEX, ORCA, SPGA, ...

Aufbau

- Bis zu 10.000 Makrozellen für Schaltfunktionen und Flip-Flops
- Große oder kleine Logikzellen
- Programmierbare Kommunikationsinfrastruktur zwischen den Zellen (unvollständig)
- Programmierung wie EPROM, EEPROM oder Flash-ROM
- Programmierung mit flüchtigem Speicher (RAM)
 - Hochfahren des FPGA durch externen Speicher (z.B. PROM)
 - Rekonfiguration (Umprogrammierung) im laufenden Betrieb möglich
- Im folgenden: Virtex-FPGAs von Xilinx

FPGA (2)

Blockschaltbild eines Virtex-FPGAs

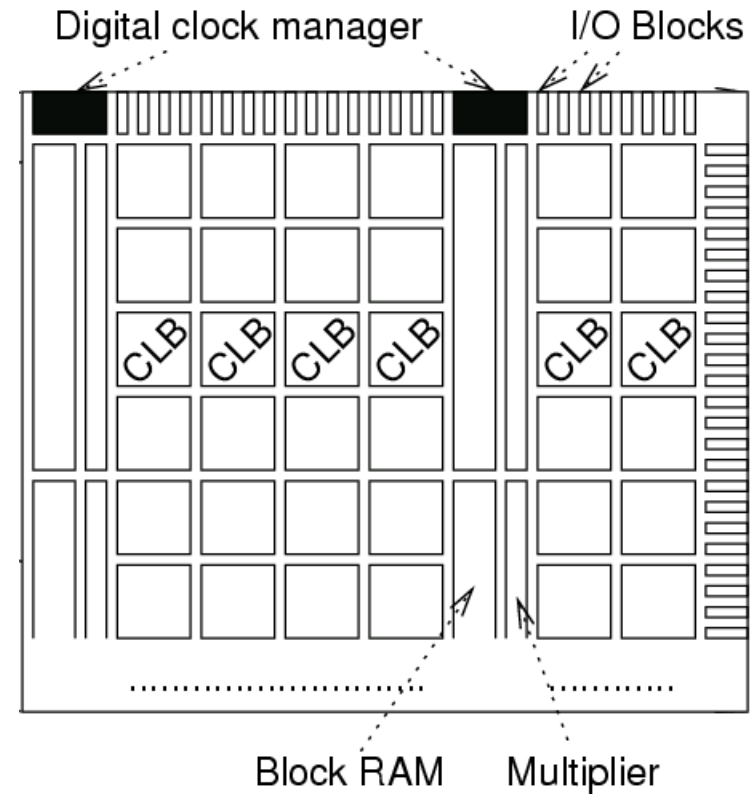


- Bis zu 240 x 108 *Configurable Logic Blocks* (CLBs)
- Bis zu 1.200 frei verfügbare I/O-Pins (I/O-Blocks)

FPGA (3)

Blockschaltbild eines Virtex-FPGAs

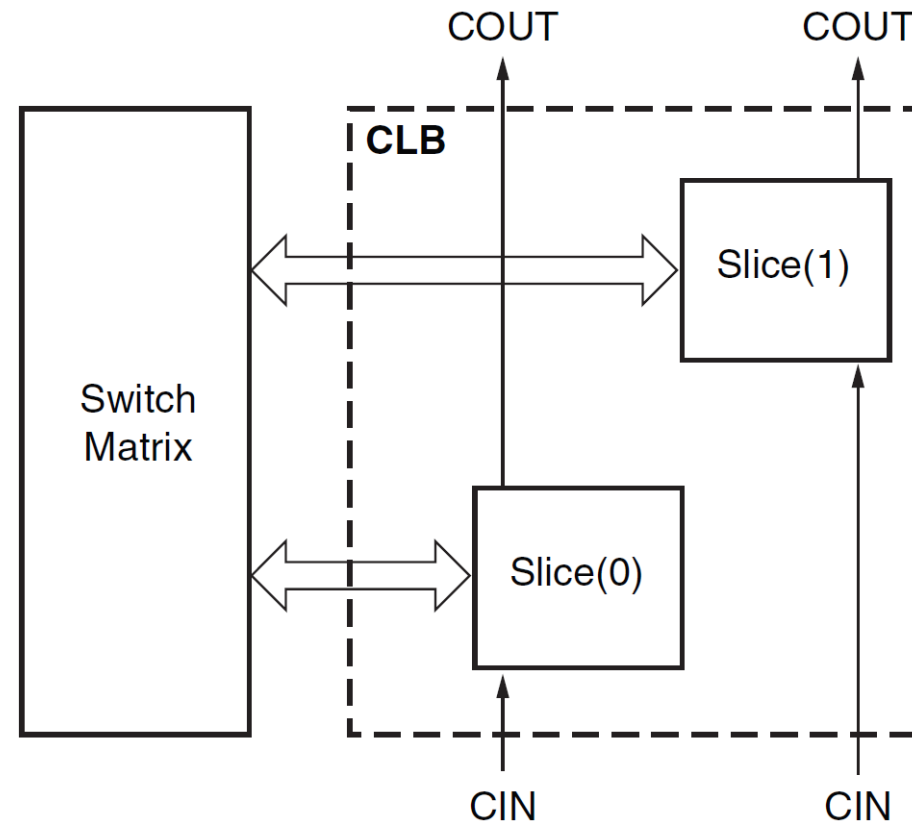
- CLBs können über eine konfigurierbare Schaltmatrix miteinander verbunden werden
- Separate Blöcke stellen Multiplizierer und RAM-Speicher für größere Datenmengen bereit



FPGA (4)

Configurable Logic Blocks (CLBs)

- Jeder CLB enthält zwei oder vier sog. *Slices*
- Über *Carry*-Ein- und -Ausgänge können Überträge von Additionen bzw. Subtraktionen schnell zwischen CLBs von unten nach oben weitergereicht werden
- Jeder CLB ist über eine Schaltmatrix an das Verbindungsnetzwerk des FPGAs angeschlossen

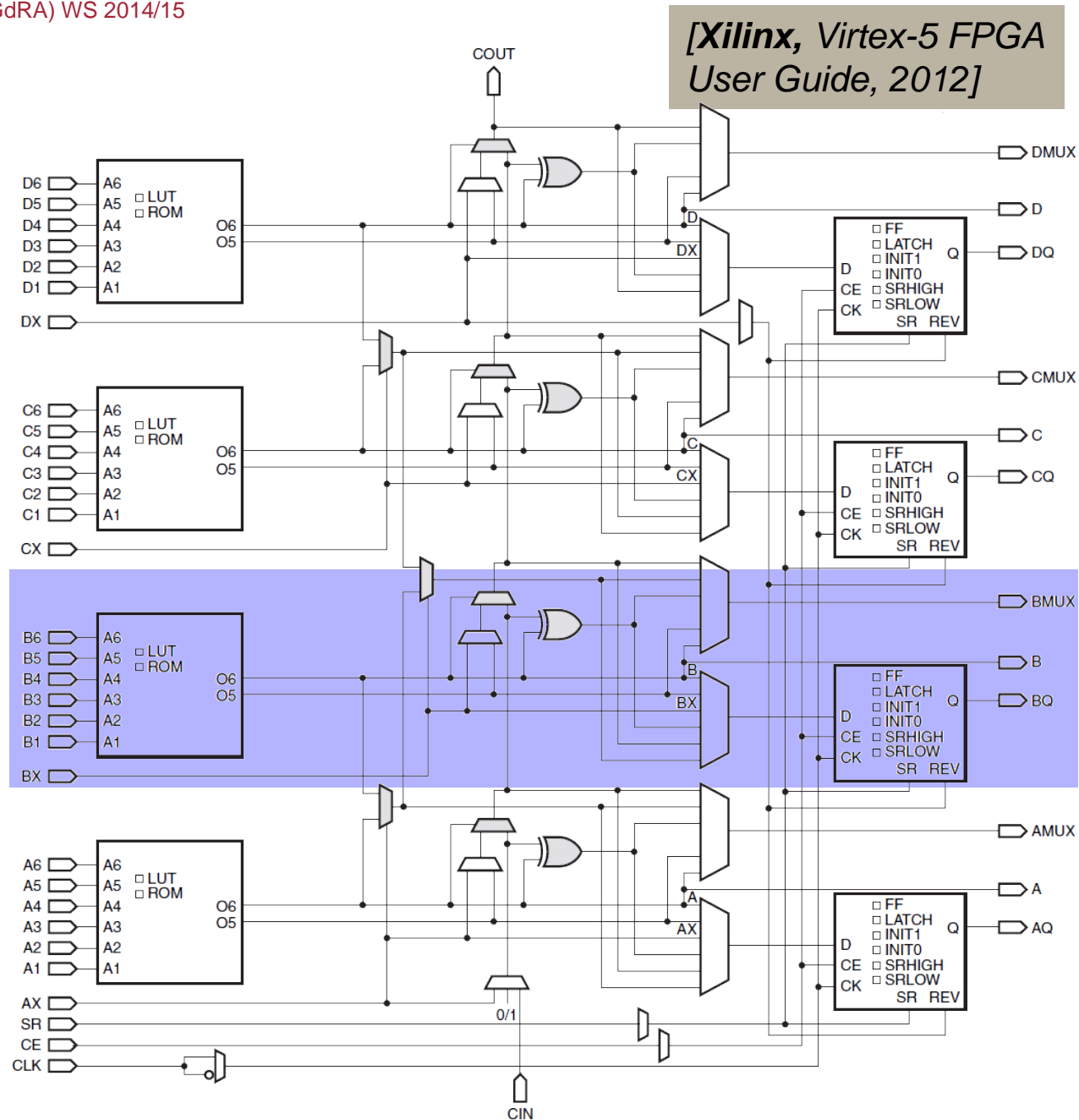


[Xilinx, Virtex-5 FPGA User Guide, 2012]

FPGA (5)

Slices

- Eine *Slice* ist zeilenweise aufgebaut, unten Zeile A, oben Zeile D

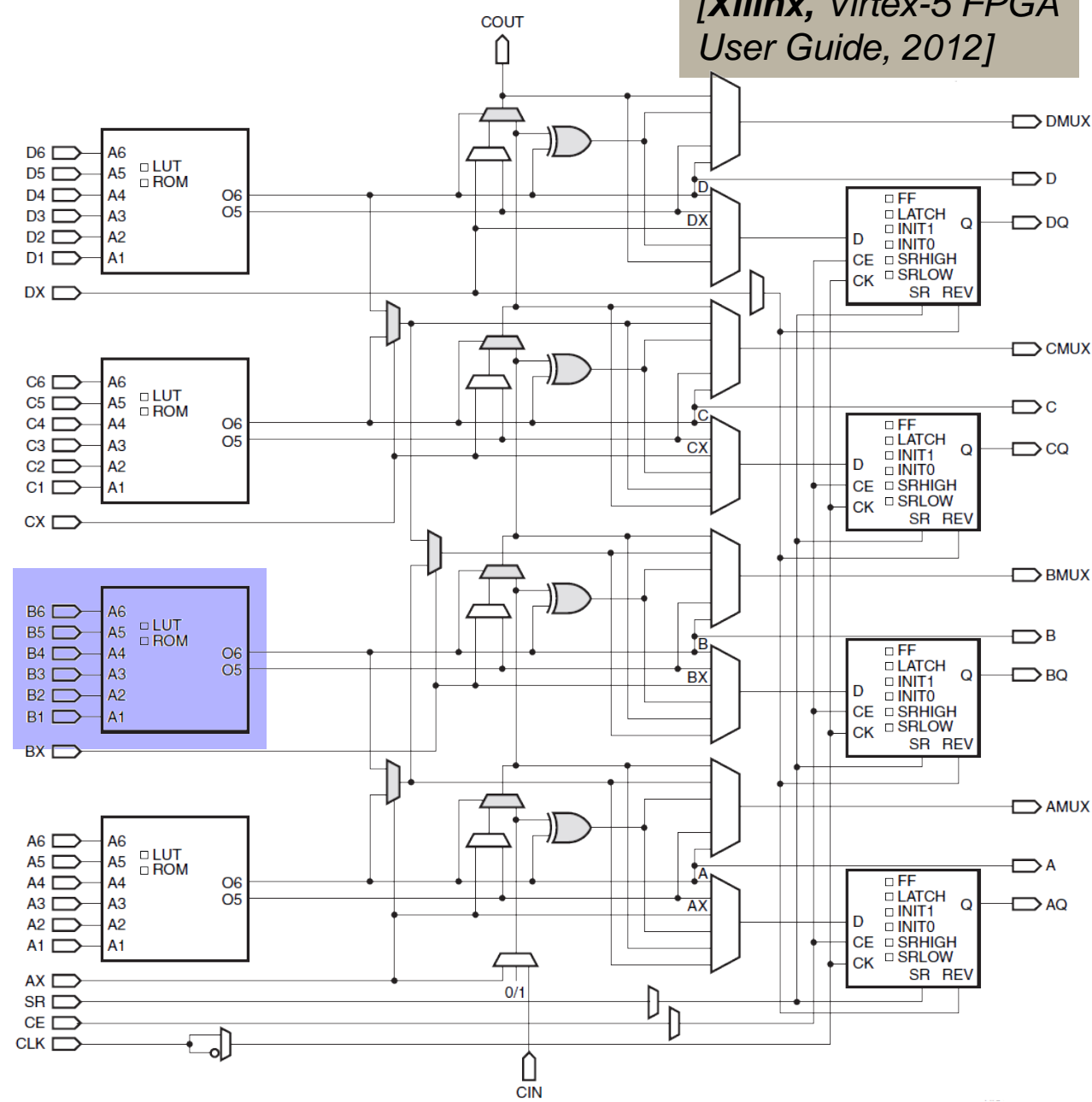


FPGA (6)

Slices

- Jede Zeile enthält einen 64 Bits großen Funktionsgenerator mit je 6 Eingängen und 2 Ausgängen

[Xilinx, Virtex-5 FPGA User Guide, 2012]

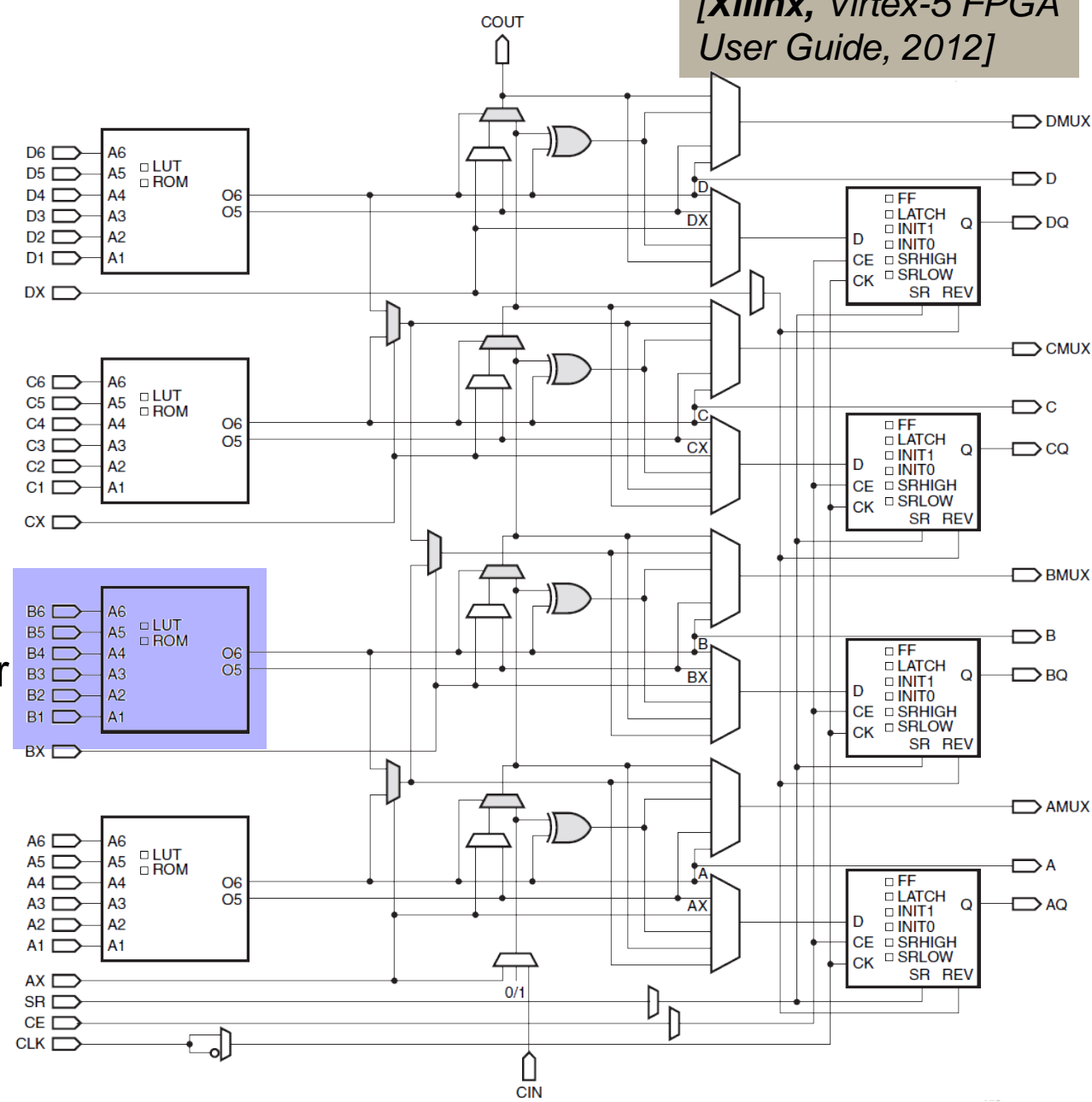


FPGA (7)

Slices

- Verwendung des Funktionsgenerators als *Lookup Table (LUT)*: Realisierung einer Schaltfunktion mit 6 Eingaben und 1 Ausgabe, oder zweier Schaltfunktionen mit 5 Eingaben
- Alternativ: Nutzung als *Read-Only Memory (ROM)*

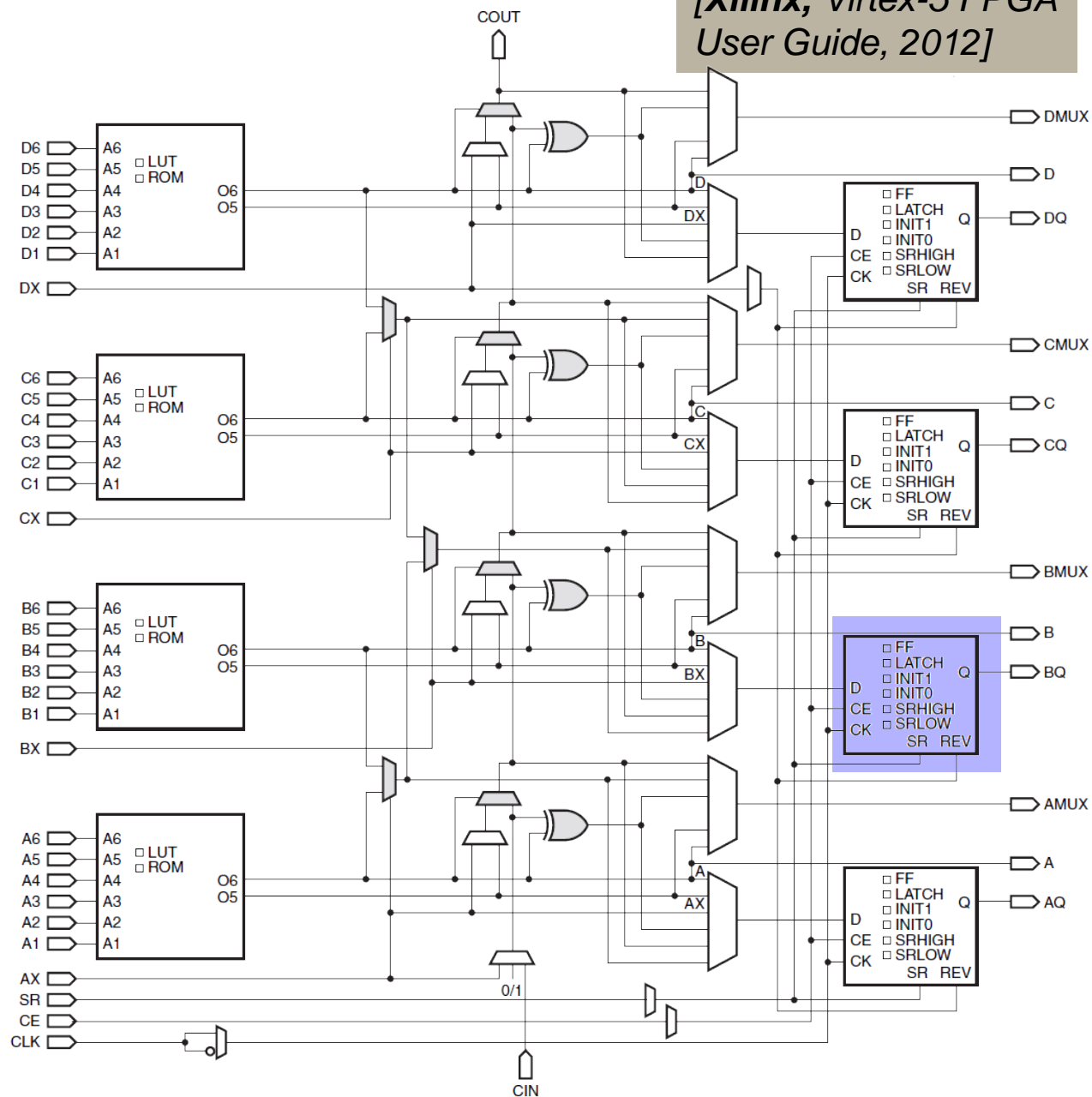
[Xilinx, Virtex-5 FPGA User Guide, 2012]



FPGA (8)

Slices

- Jede Zeile enthält einen 1-Bit Speicher, der als flankengetriggertes D-Flip-Flop oder als pegelgetriggertes *Latch* konfiguriert werden kann
- Alle Flip-Flops einer *Slice* erhalten den gleichen Takt

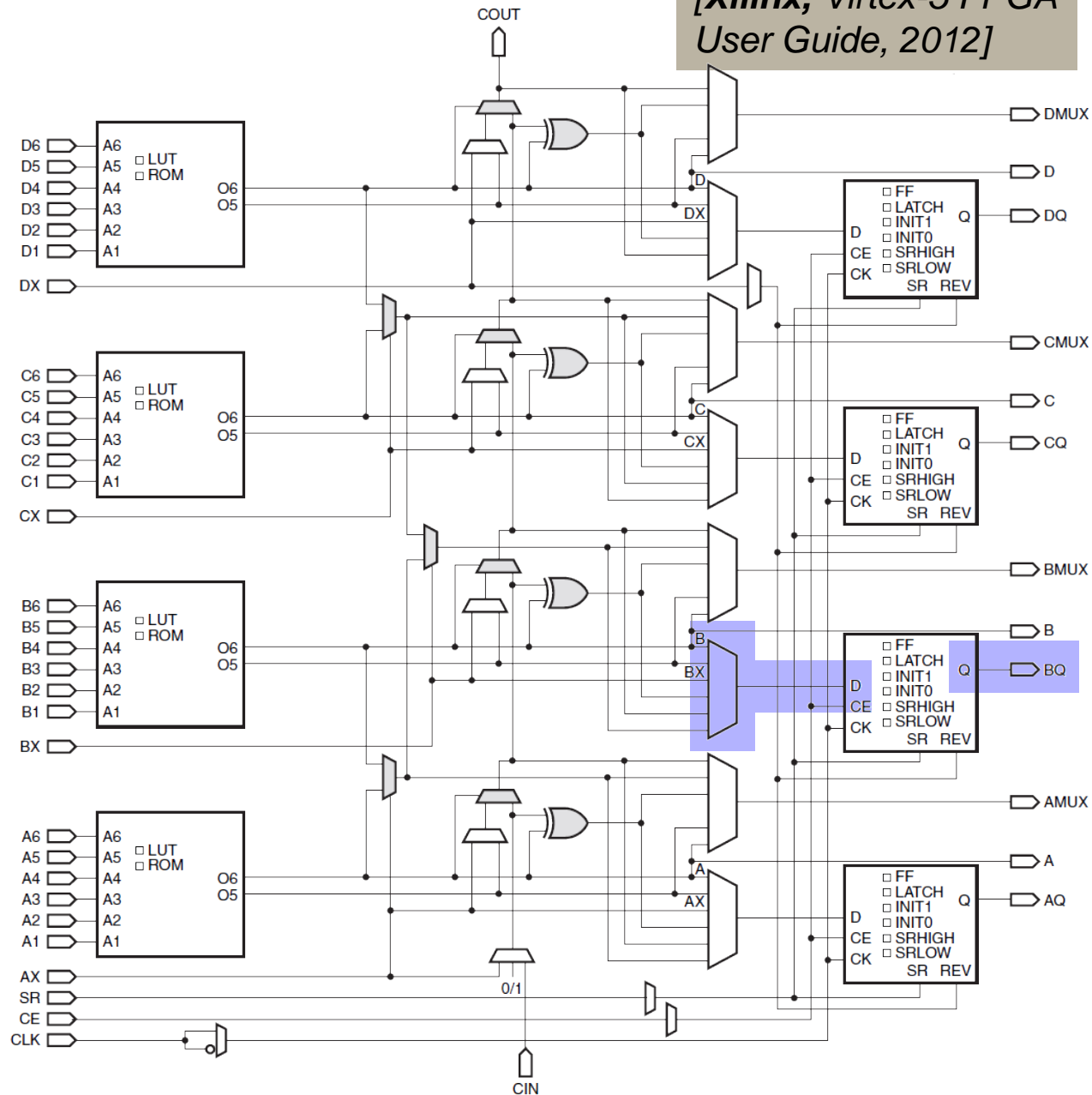


FPGA (9)

Slices

- Der im Flip-Flop / Latch gespeicherte Zustand steht direkt als Ausgang der Slice zur Verfügung
- Über Multiplexer wird konfiguriert, was im Flip-Flop / Latch gespeichert wird (z.B. Ausgänge des Funktionsgenerators)

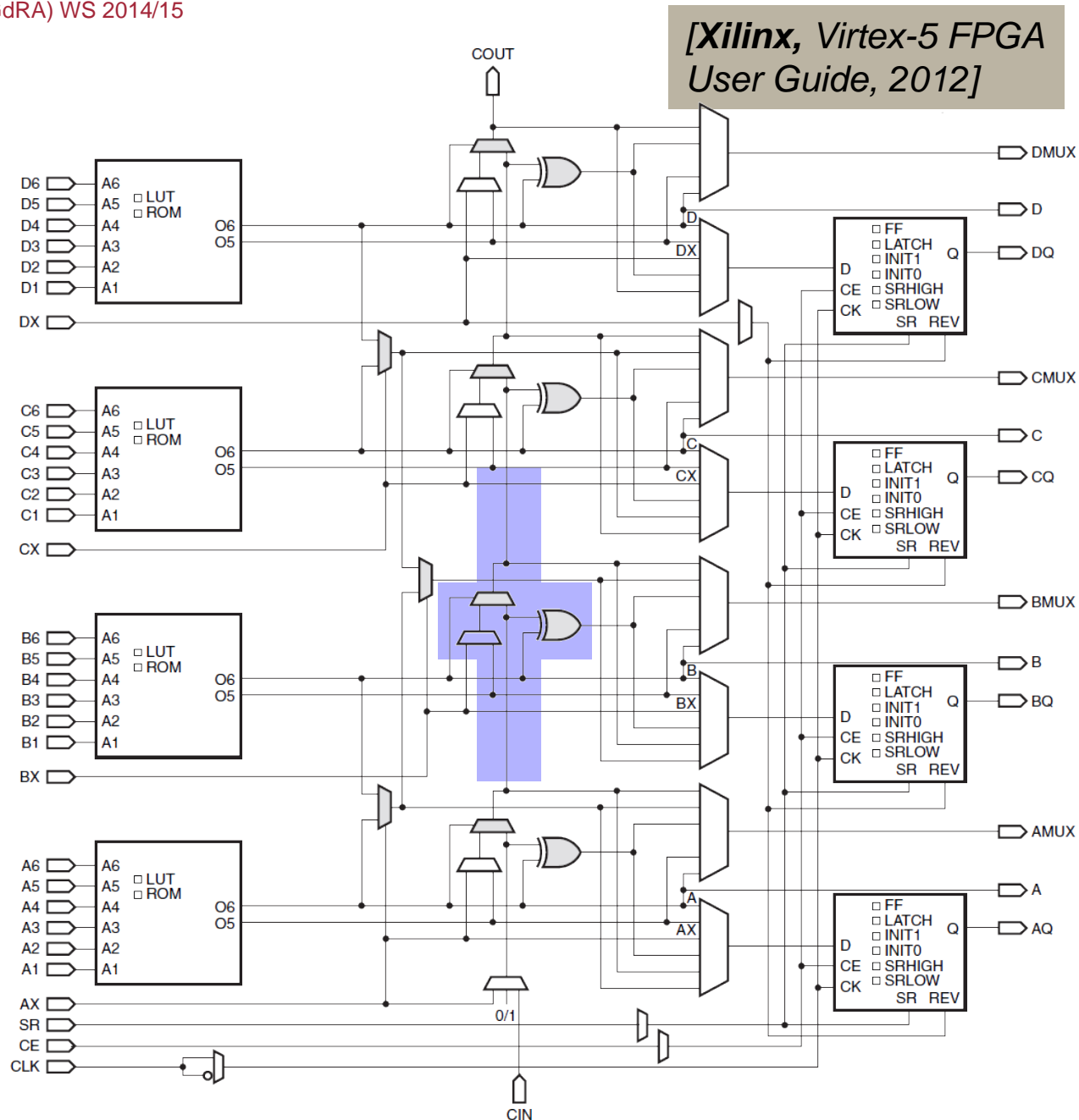
[Xilinx, Virtex-5 FPGA User Guide, 2012]



FPGA (10)

Slices

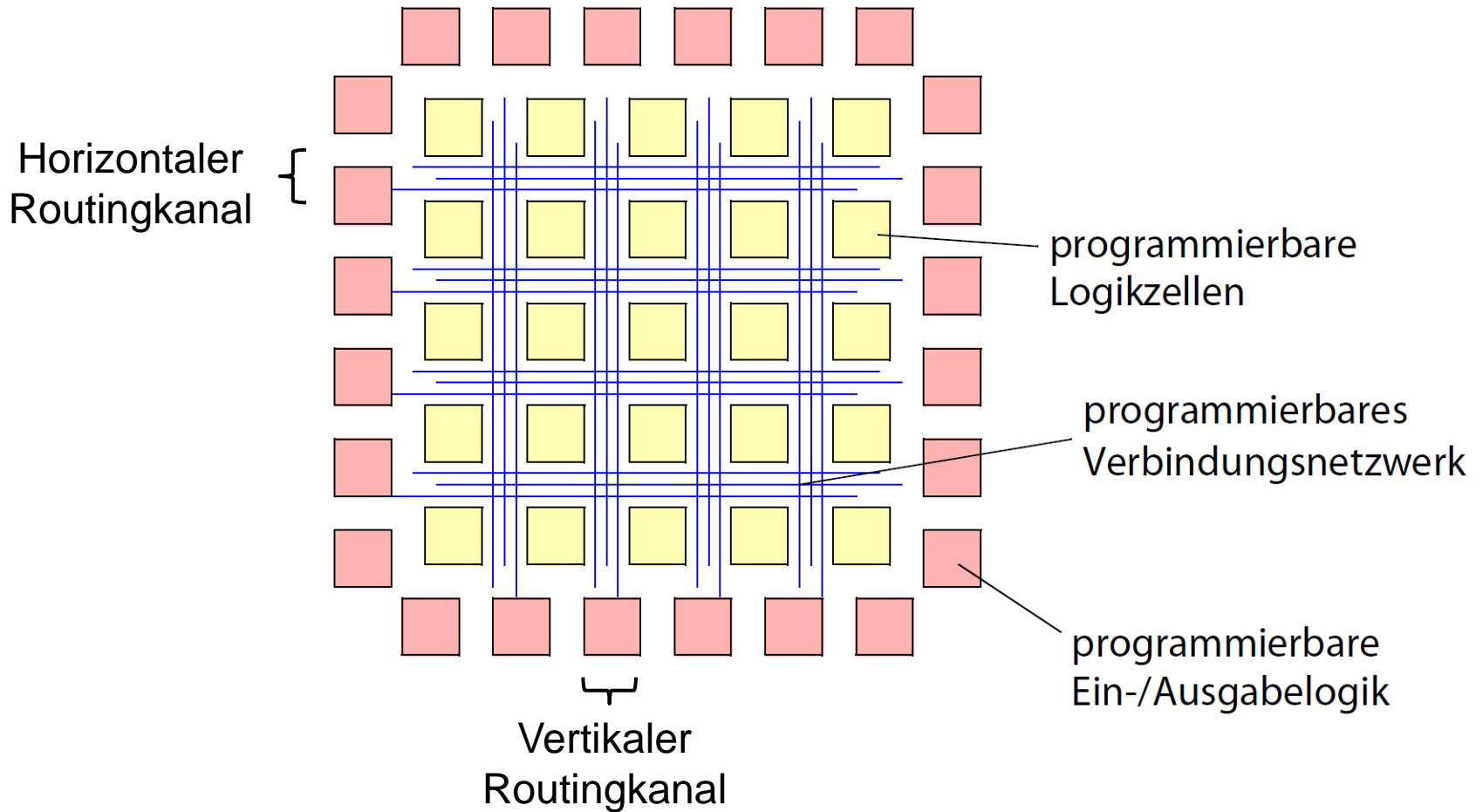
- Über Carry-Logik jeder Zeile können Überträge bei Addition/Subtraktion schnell von unterer zu oberer Zeile propagiert werden
- Separates XOR-Gatter dient zur Addition/Subtraktion eines Operanden mit einem Carry-Bit



[Xilinx, Virtex-5 FPGA User Guide, 2012]

FPGA (11)

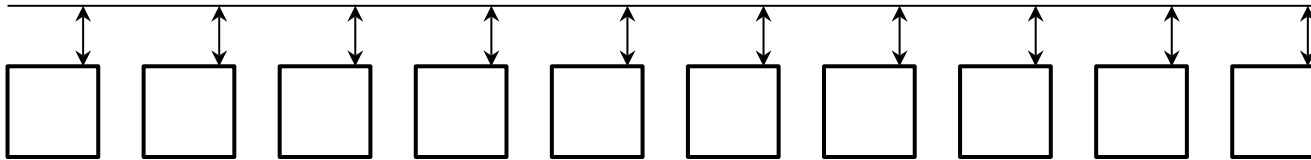
Verbindungsnetzwerk



FPGA (12)

Verbindungsnetzwerk (Xilinx Virtex II)

- *Long Lines*

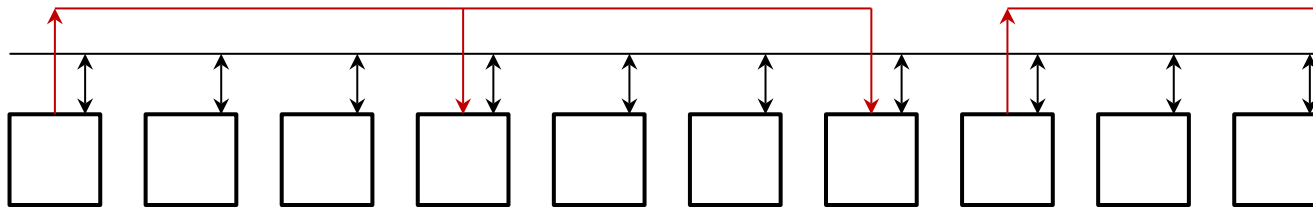


- Überspannen den kompletten FPGA-Chip
- Bidirektionale Verbindung
- 24 *Long Lines* pro horizontalem und vertikalem Kanal

FPGA (13)

Verbindungsnetzwerk (Xilinx Virtex II)

– *Hex Lines*

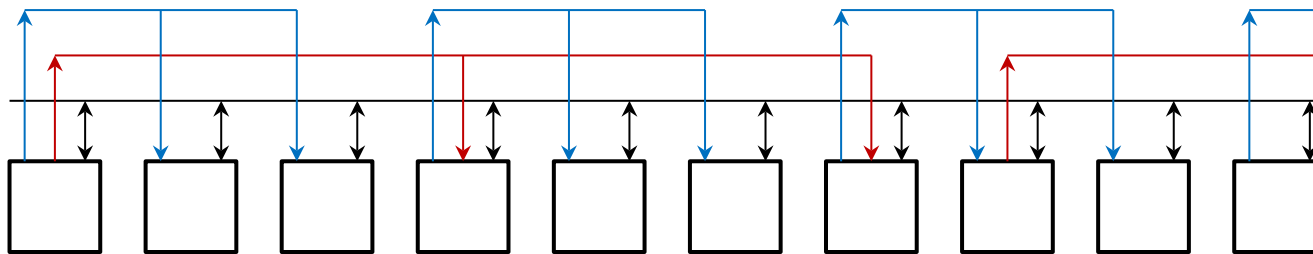


- Verbinden einen CLB mit seinem 3. und 6. Nachbarn
- Unidirektionale Verbindung, kann nur am linken Ende getrieben werden
- 120 *Hex Lines* pro horizontalem und vertikalem Kanal

FPGA (14)

Verbindungsnetzwerk (Xilinx Virtex II)

– *Double Lines*

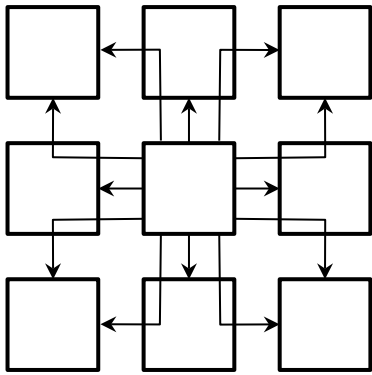


- Verbinden einen CLB mit seinem nächsten und übernächsten Nachbarn
- Unidirektionale Verbindung, kann nur am linken Ende getrieben werden
- 40 *Double Lines* pro horizontalem und vertikalem Kanal

FPGA (15)

Verbindungsnetzwerk (Xilinx Virtex II)

- Direkte Verbindungen

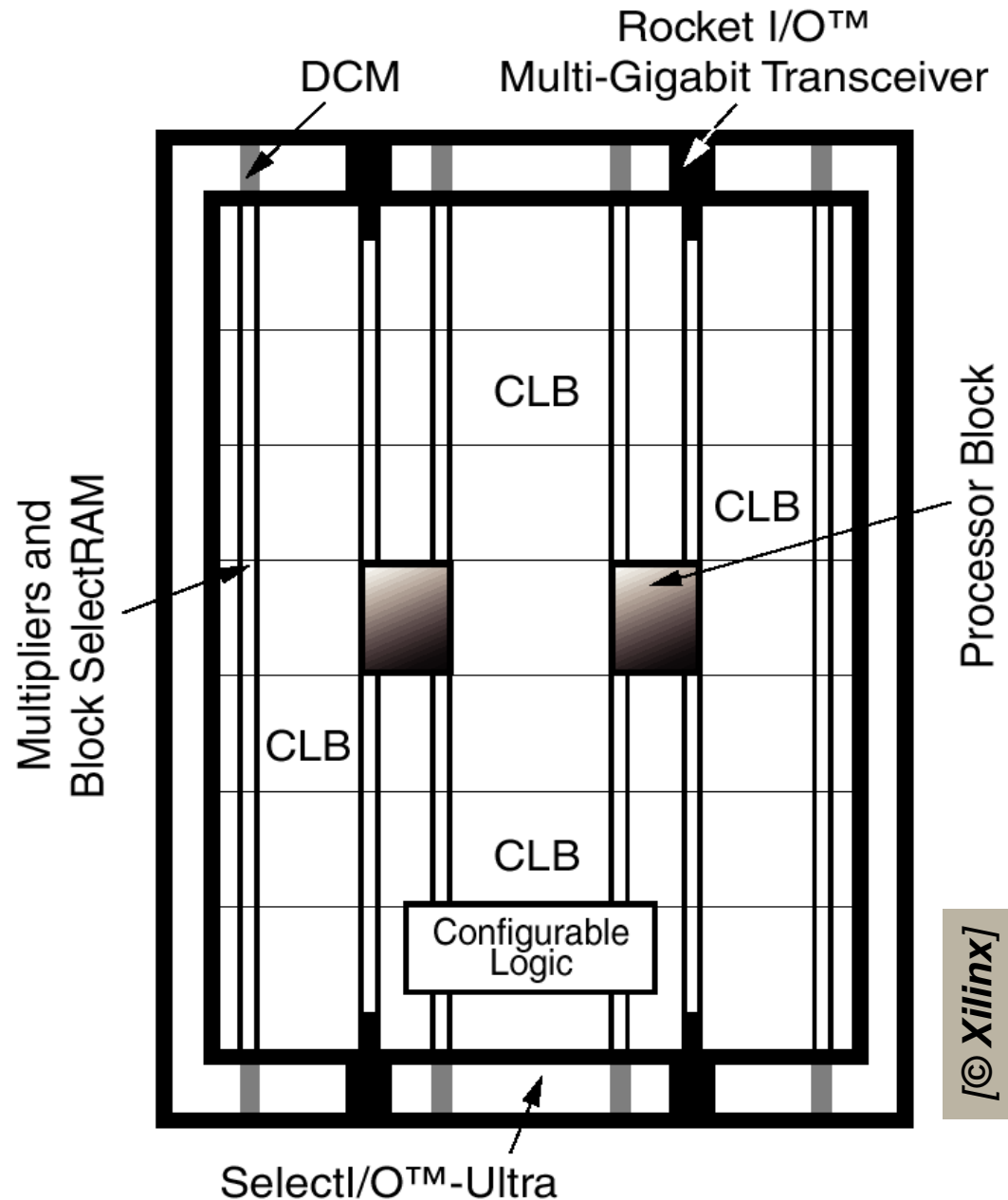


- Verbinden einen CLB mit direkten Nachbarn (horizontal, vertikal, quasi-diagonal)

FPGA (16)

PowerPC Cores

- Virtex-FPGAs können bis zu 4 PowerPC Prozessoren eingebettet in die konfigurierbare Hardware enthalten
- Die PowerPCs können über das Verbindungsnetzwerk die konfigurierbare Hardware des FPGAs mitbenutzen
- ☞ Aufbau sehr effizienter gemischter Hardware- und Software-Systeme möglich!



[© Xilinx]

Zusammenfassung (1)

Halbleiter-Bauelemente

- Halbleiter: Kombination von N- und P-leitfähigen Materialien, so dass gezielt Elektronenmangel bzw. -überschuss entsteht
- Diode: Bauelement, dass Stromfluss nur in einer Richtung zulässt
- Transistor: Bauelement, dass Stromfluss von Kollektor zu Emitter abhängig von Strom an der Basis zulässt oder sperrt
- Aus Transistoren können einfach logische Gatter konstruiert werden
- MOS-FET: Transistor, der aufgrund eines elektrischen Feldes zwischen Gate und Source schaltet
- Vorteil des MOS-FETs: geringer Stromverbrauch, niedrige Leistungsaufnahme
- CMOS: Prinzip, mit komplementären MOS-FETs (n- und p-Kanal MOS-FETs) symmetrische Schaltungen aufzubauen

Zusammenfassung (2)

Programmierbare Logikbausteine

- Wenn mehrere Schaltwerke an einen gemeinsamen Bus angeschlossen sind, sind Kurzschlüsse zu vermeiden
- *Tri-State*-Puffer mit Zuständen 0, 1 und X, bzw. *Wired-OR* / *Wired-AND*-Schaltungen mit Zuständen 0 und X bzw. 1 und X
- Von PROMs bis zu PLAs:
 - AND- und OR-Matrix stellen Verbindungen zwischen Eingabe- und Ausgabeleitungen her
 - Je nach Technologie können einzelne Verbindungen in den Matrizen gesetzt / aufgehoben werden, was die Programmierung ermöglicht
- FPGAs
 - Komplexer Aufbau bestehend aus CLBs, *Slices*, Funktionsgeneratoren und Speichern
 - Kanalweises Routing über kurze, mittlere oder lange Leitungen