Kap.1
Hardwareentwurf

Enwurfsautomatisierung
EDA
electronic design
automation

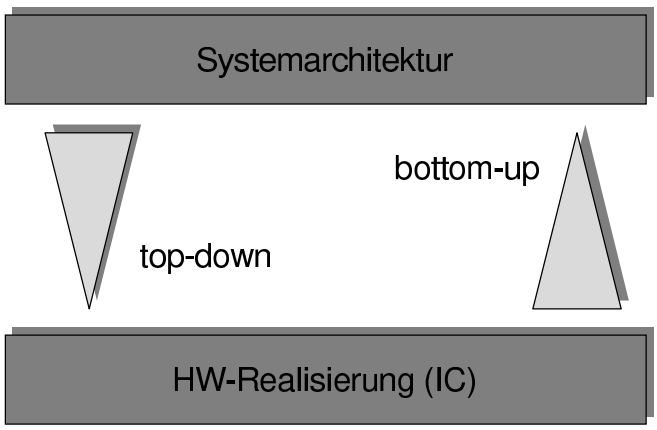
RA Überblick

- Einleitung
- Hardwareentwurf
- Befehlsschnittstelle
- Mikroarchitektur
- Speicherarchitektur
- Ein-/Ausgabe
- Multiprozessorsysteme, ...

1 Hardwareentwurf

- **1.1** Hardwareentwurfsschritte
- **1.2** Hardwarebeschreibungssprachen
- **1.3** Hardwareverifikation
- **1.4** Hardwaresynthese
- **1.5** Technologieabbildung

Von der Architektur zur HW



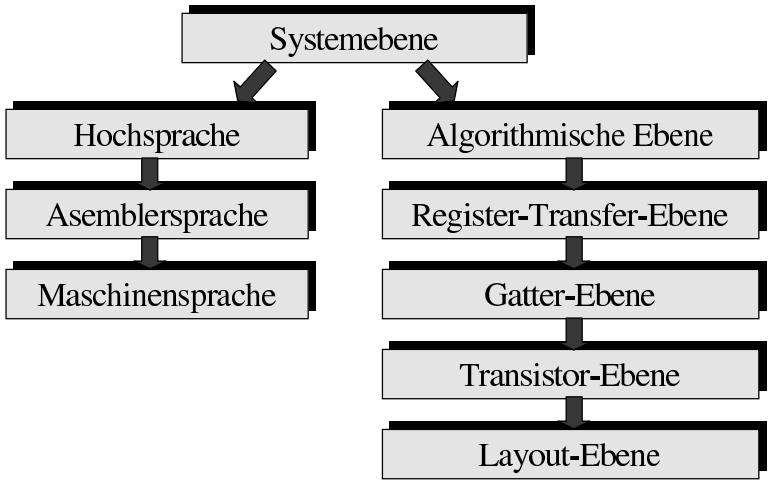
Entwurfsweisen

- top-down
  - abstrakte Beschreibung des Systems wird sukkzessive verfeinert bis zur HW
- bottom-up
  - bereits entworfene Komponenten werden zu komplexeren Einheiten zusammengefügt

Realität:

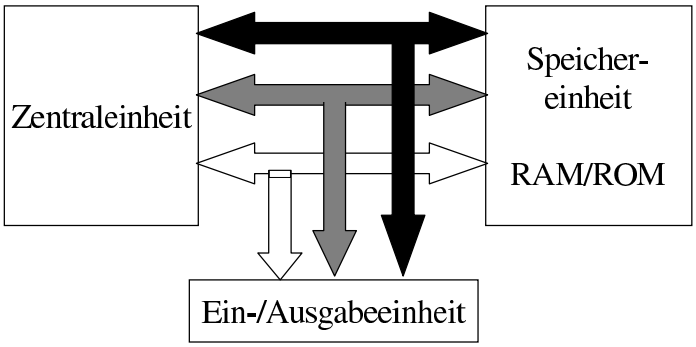
Michung aus beiden Prozessen

Abstraktionsebenen



Systemebene

- Funktionale Einheiten die miteinander kommunizieren



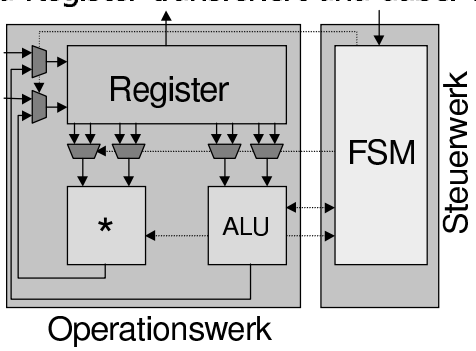
Algorithmische Ebene

- Die Funktionen einzelner Blöcke wird durch Algorithmen in einer HW-Beschreibungssprache spezifiziert

```
int dgl(int x,y,u,dx,a) {
  int x1, y1, u1;
  do
    x1 = x+dx;
    u1 = u-(3*x*u*dx)-(3*y*dx);
    y1 = y+(u * dx);
    x = x1; u = u1; y = y1;
  while (x1<=a);
  return y;
}
```

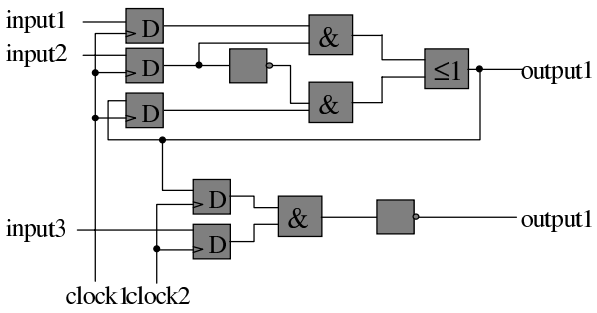
Register-Transfer Ebene

- Darstellung der Funktionalen Einheiten durch Datenpfad- und Kontrollpfad (Daten werden von Register zu Register transferiert und dabei Verarbeitet)



Gatterebene

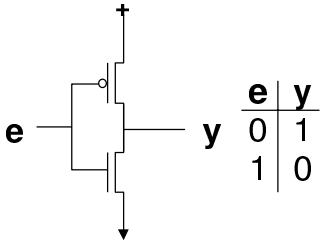
- Es gibt nur noch Boole'sche Signale, Boole'sche Gatter und einfache FlipFlops



Transistorebene

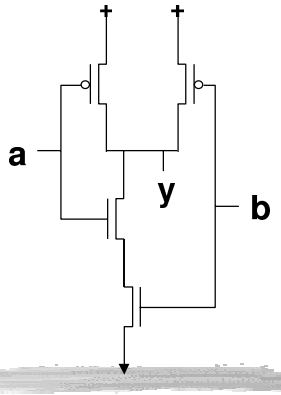
- Realisierung Boolescher Elemente durch Transistoren

Inverter



e	y
0	1
1	0

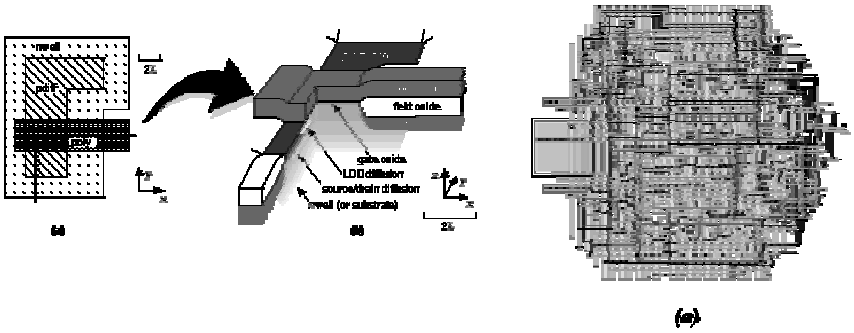
NAND



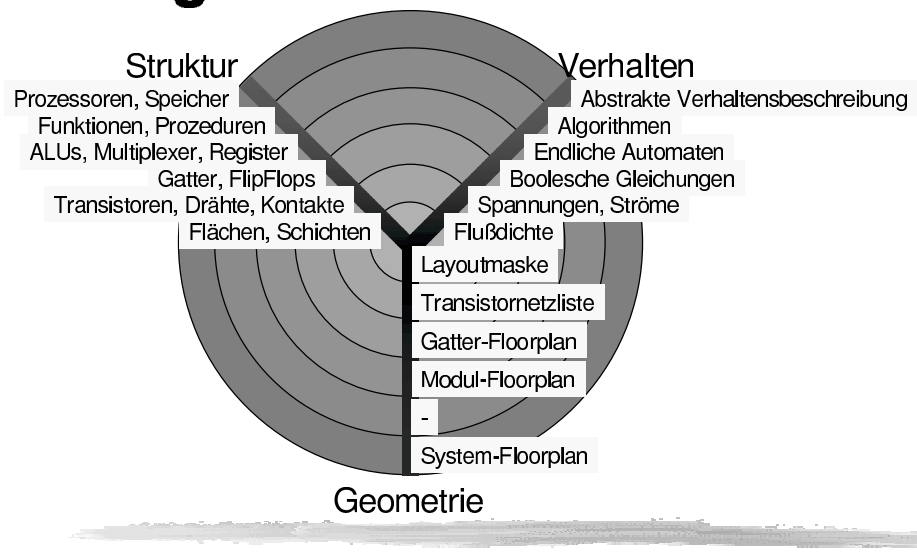
a	b	y
0	0	1
0	1	1
1	0	1
1	1	0

Layoutebene

- Realisierung von Transistoren durch dotierte Bereiche und isolierenden Schichten auf dem IC



Y-Diagramm

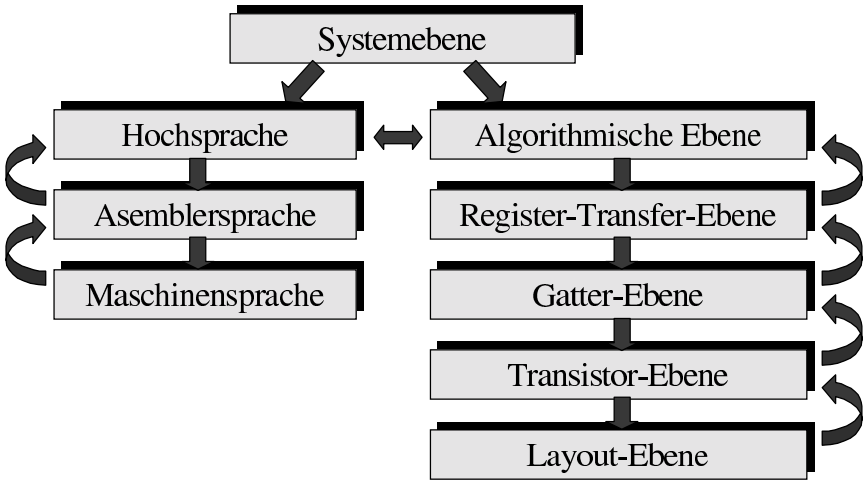


Systementwurf

Ziel des Entwurfsprozesses

- Beschreibungen von einer hohen Abstraktionsebene in eine niedere Abstraktionsebene zu transformieren
  - durch Verfeinerung
  - unter Beibehaltung der Funktionalität (eventuell auch des Zeitverhaltens)
- Verifikation/Validation der Beschreibung auf den unterschiedlichen Ebenen

Systementwurf in der Realität



Automatisierung

- Text-/Grafikeditoren
- Simulation
  - Testbenchgeneratoren
  - Coverage-Analysatoren
- Verifikation
  - Äquivalenzprüfung
  - Modellprüfung
  - linting (design rule checking)
- Timing Analyse
- Synthese
  - Verhaltenssynthese
  - Logiksynthese
- Plazierung und Verdrahtung
- Automatic testpattern generator (ATPG)