Vorlesung
Hardware-Beschreibungssprachen

Dr.-Ing. S. Arlt

Fakultät EI

Fachbereich Elektronische Schaltungen und Systeme

Vorlesung
Hardware-Beschreibungssprachen

Kapitel 1 : Einführung

Motivation

- Lücke zwischen Integrationsdichte und Design
- Steigende Komplexität
  - 1970 Intel CPU 4004
    4 Bit, $10\mu m, 600mm^2, 0,74MHz, 2,25k$ Transistoren
  - 1997 Intel CPU Pentium 2
    32 Bit, $0,35\mu m, 203mm^2, 300MHz, 7.500k$ Transistoren
  - 2005 Intel CPU Pentium 4 6xx
    64 Bit, $0,09\mu m, 135mm^2, 3587,8MHz, 169.000k$ Transistoren
- Abstraktion, Generik
- Time to Market
- Intellectual Properties (IP) & Reuse

Was ist eine HDL?

- Modellierungssprachen für die Simulation und Implementierung elektronischer Schaltungen
- Dokumentationsmittel speziell auf höherem Abstraktionsniveau
- Übliche HDLs: VHDL, ABEL, Verilog, Altera-HDL, UDL/I
- Unterschiede zu Hochsprachen wie C, C++, Pascal:
  - Beschreibung von Parallelität
  - strukturelle Beschreibung (Entwurf des Steuer- und Datenflusses)
  - Existenz entsprechender Datentypen für die Modellierung von Signalvektoren und Signalpegeln

Anwendungen

- Spezifikation
- Verifikation, Modellierung $\Rightarrow$ Simulation
- Synthese
- Dokumentation

Historie VHDL

VHSIC **V**ery **H**igh **S**peed **I**ntegrated **C**ircuits - Programm des DoD
VHDL- **H**ardware **D**escription **L**anguage

- 1981 erste Anforderungen
- 1983-1985 VHDL-Entwicklung von Intermetrics, IBM, TI
- 1987 IEEE Standard 1076-1987 VHDL-87
- 1991 IEEE Standard 1164 (neunwertige Logik)
- 1993 IEEE Standard 1076-1993 VHDL-93
- 1995 IEEE Standards 1076.3 1076.4

Historie Verilog

- 1985 Verilog- Spache und Simulator von Gateway Automation entwickelt
- 1985-1989 wachsende Zunahme der Verwendung von Verilog bei ASIC-Herstellern
- 1989 Verilog wird von Cadence übernommen
- 1990 Formulierung von *Open Verilog International* (OVI)
- 1995 IEEE Standard 1364

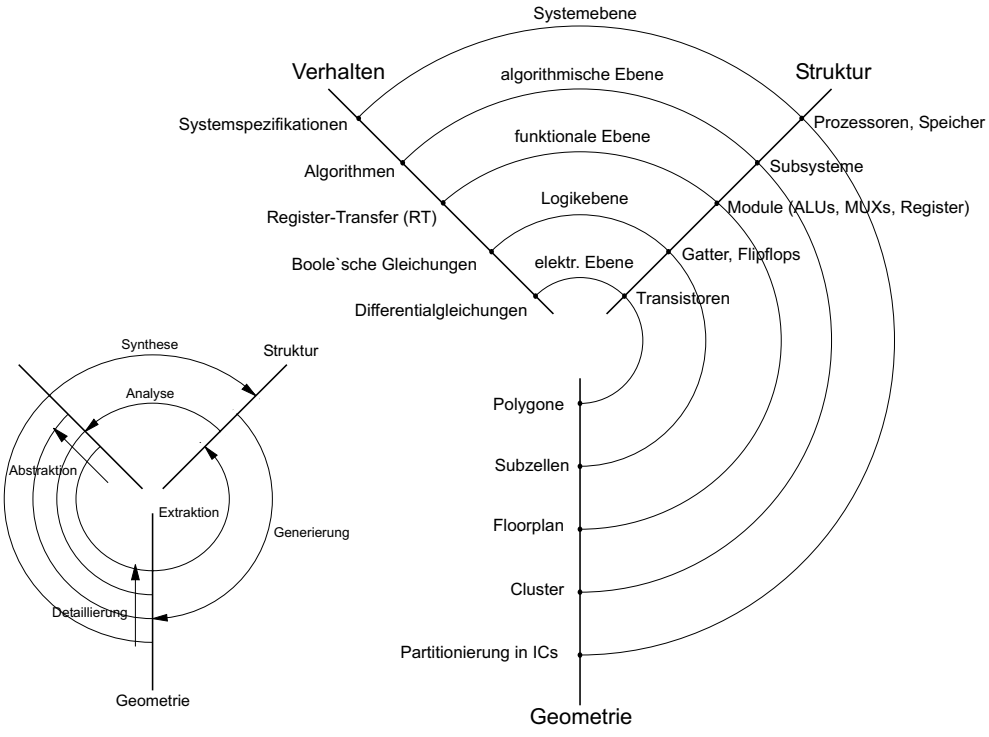
Kurzvergleich

- VHDL bietet mehr lexikalische Elemente (erweiterte Typdefinitionen, Variablen)
- Vorteile von Verilog liegen bei der Simulation von Netzlisten mit Timingmodellen, ist an C angeleht, keine strenge Typprüfung
- Die Favorisierung einer Sprache ist bei den heute verfügbaren Simulations- und Synthesewerkzeugen nicht mehr notwendig
- Europa: ca. 90% Nutzung von VHDL
- USA: ca.50% Nutzung von VHDL
- Verilog oft effizienter als VHDL

Vorlesung
Hardware-Beschreibungssprachen

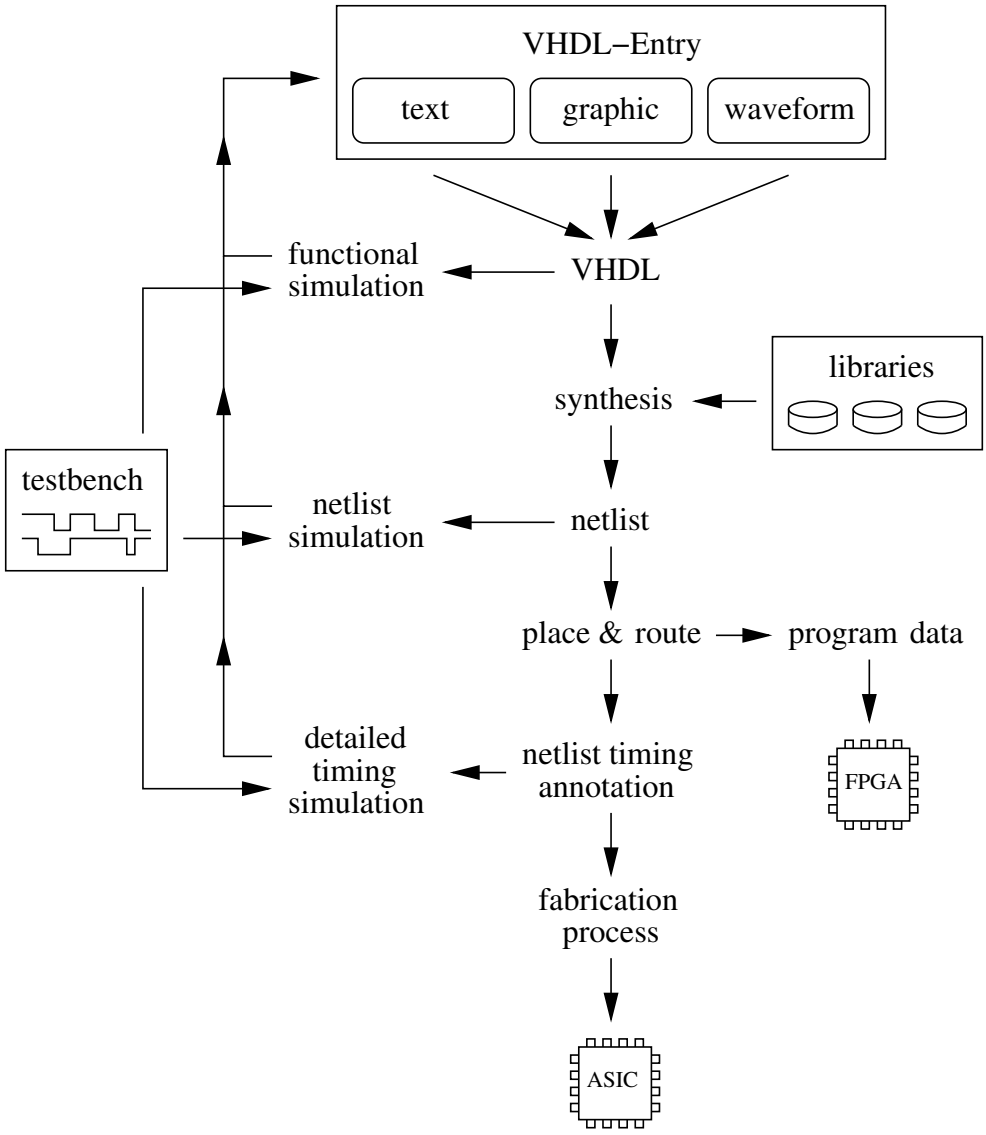
Kapitel 2 : Entwurfsmethodik

Systematisierung des Entwurfs



Y-Diagramm nach Gajski-Kuhn

Entwurfsablauf

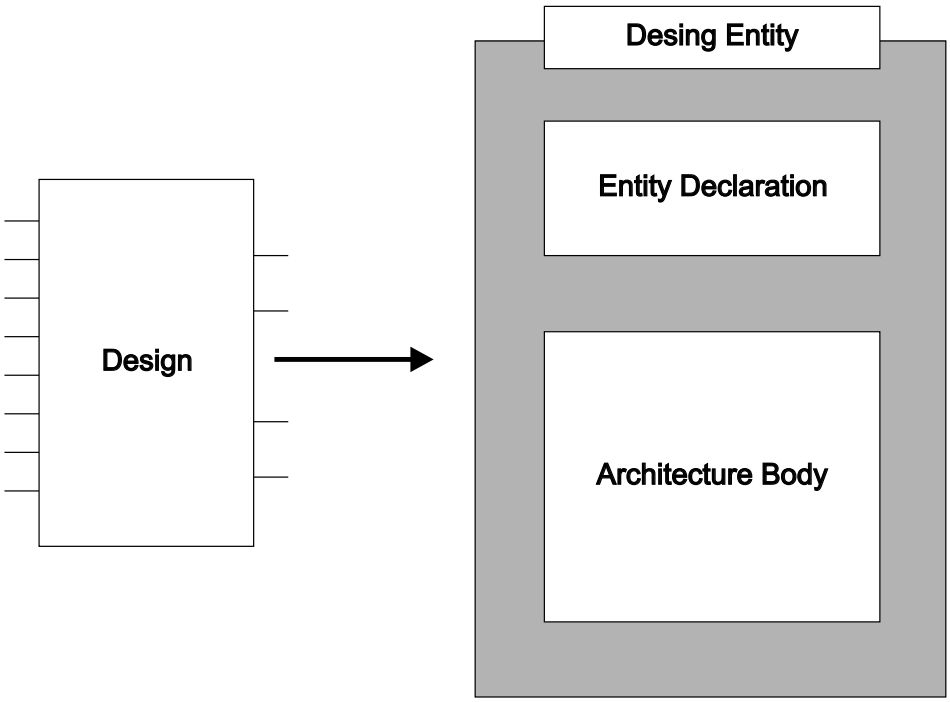


VHDL Design Flow

Vorlesung
Hardware-Beschreibungssprachen

Kapitel 3 : Aufbau eines VHDL Modells

Grundstruktur eines VHDL Modells



Syntax

library <library_name>[, <library_name_n>] ;
use<*library_name* > [. < *package_name*>]. > *all*/*element_name*>;

entity <entity_name> **is**
    [**generic**(< *parameter_declaration* >);]
port(< *input*/*output_declaration* >);
    [<*entity_declaration*>;]
end[*entity*] < *entity_name*>;

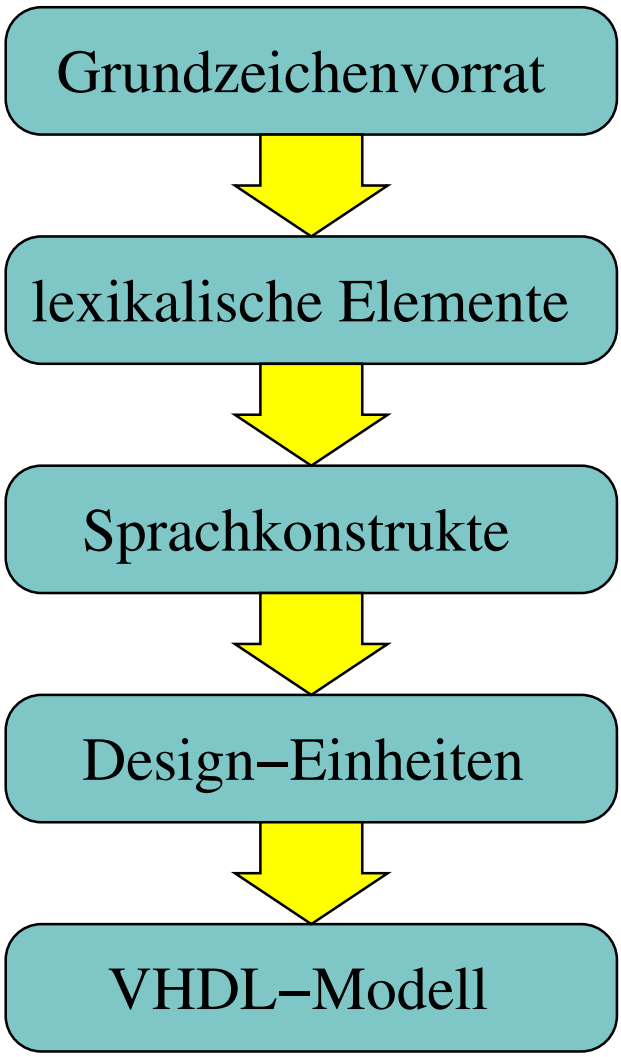
architecture <architecture_name> **of** <entity_name> **is**
    [<*architecture_declaration*>;]
begin
    <*VHDL_statements*>;
end[*architecture*] < *architecture_name*>;

configuration <configuration_name> **of** <identifier_name> **is**
<*configuration_statements*>
end[*configuration*][<*configuration_name*>];

Vorlesung
Hardware-Beschreibungssprachen

Kapitel 4 : VHDL Sprachelemente

VHDL Sprachaufbau



VHDL-87 7-Bit Zeichensatz

VHDL-93 8-Bit ASCII Zeichensatz

Lexikalische Elemente

= Wörter, lat. Lexeme

- Kommentare
- Bezeichner (identifier)
- Literale
- Reservierte Wörter

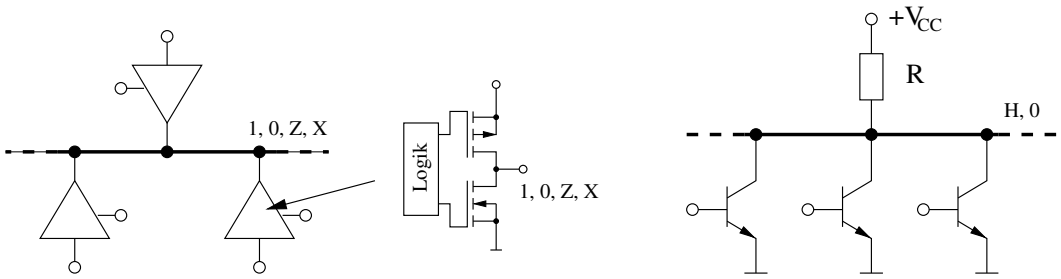
Literale

Literal= (lat. litera) = Buchstabe

Literale sind von einer Programmiersprache definierte Zeichenfolgen zur Darstellung der Werten von Basistypen

- Abstract
  - Decimal (-Mantisse-Exponent)
  - Based (Basis-Mantisse-Exponent)
    - Integer/Real
- Character
- String
- Bit String

Beispiel - Bussysteme



Tri–State–Treiber CMOS–Realisierung Open–Collector–Treiber

std_ulogic

type std_ulogic is (

'U' -- uninitialized

'X' -- forcing unknown, starker unbekannter Wert

'0' -- forcing 0

'1' -- forcing 1

'Z' -- high impedanz

'W' -- weak unknown

'L' -- weak 0

'H' -- weak 1

'-'); -- don't care

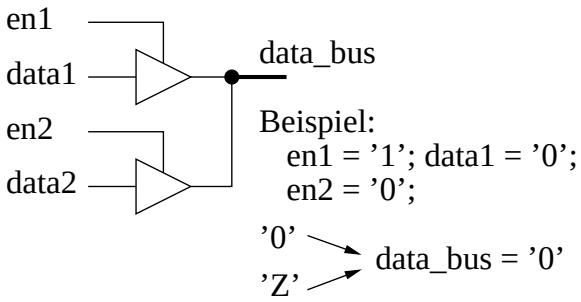
Resolution Function

	U	X	0	1	Z	W	L	H	-
U	'U'	'U'	'0'	'U'	'U'	'U'	'0'	'U'	'U'
X	'U'	'X'	'0'	'X'	'X'	'X'	'0'	'X'	'X'
0	'0'	'0'	'0'	'0'	'0'	'0'	'0'	'0'	'0'
1	'U'	'X'	'0'	'1'	'X'	'X'	'0'	'1'	'X'
Z	'U'	'X'	'0'	'X'	'X'	'X'	'0'	'X'	'X'
W	'U'	'X'	'0'	'X'	'X'	'X'	'0'	'X'	'X'
L	'0'	'0'	'0'	'0'	'0'	'0'	'0'	'0'	'0'
H	'U'	'X'	'0'	'1'	'X'	'X'	'0'	'1'	'X'
-	'U'	'X'	'0'	'X'	'X'	'X'	'0'	'X'	'X'

Synthese mit std_logic und std_ulogic

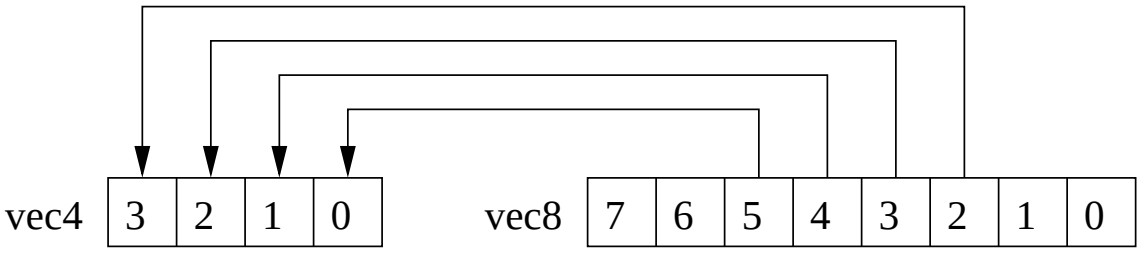
```
library ieee;
use ieee.std_logic_1164.all;
entity stdlog is
  port(en1, en2, data1, data2 : in std_logic;
     data_bus : out std_logic);
end stdlog;
architecture rtl of stdlog is
begin
  data_bus <= data1 when en1 = '1' else 'Z';
  data_bus <= data2 when en2 = '1' else 'Z';
end rtl;
```

```
library ieee;
use ieee.std_logic_1164.all;
entity stdulog is
  port(en1, en2, data1, data2 : in std_ulogic;
     data_bus : out std_ulogic);
end stdulog;
architecture rtl of stdulog is
begin
  data_bus <= data1 when en1 = '1' else 'Z';
  data_bus <= data2 when en2 = '1' else 'Z';
end rtl;
```



⇒ Fehler !!

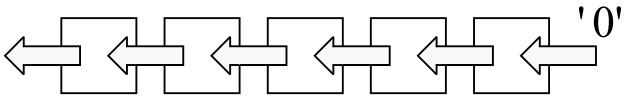
Slice Names



vec4 <= vec8(5 downto 2);

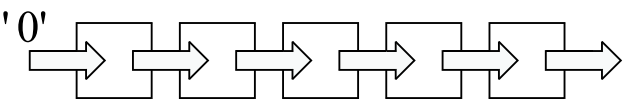
VHDL Shift Operationen

sll: shift left



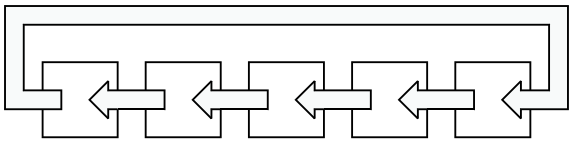

```
a <= "01101";

q1 <= a sll 1;
-- q1 = "11010"
```

srl: shift right



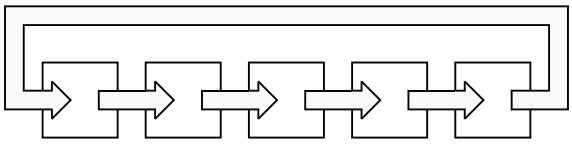

```
q2 <= a srl 3;
-- q2 = "00001"
```

rol: rotate left



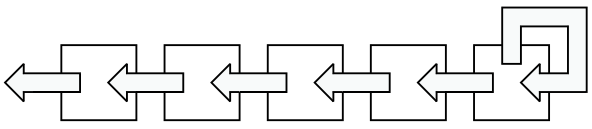
```
q3 <= a rol 2;
-- q3 = "10101"
```

ror: rotate right



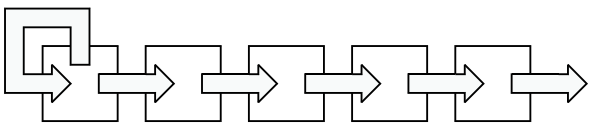
```
q4 <= a ror 1;
-- q4 = "10110"
```

sla: shift left and append value 'right



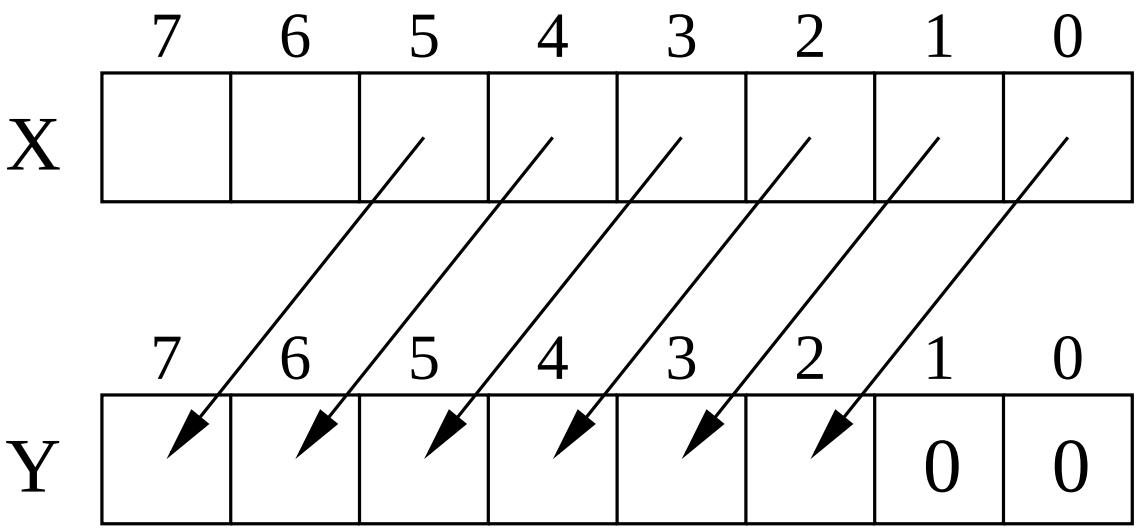
```
q5 <= a sla 2;
-- q5 = "10111"
```

sra: shift right and append value 'left



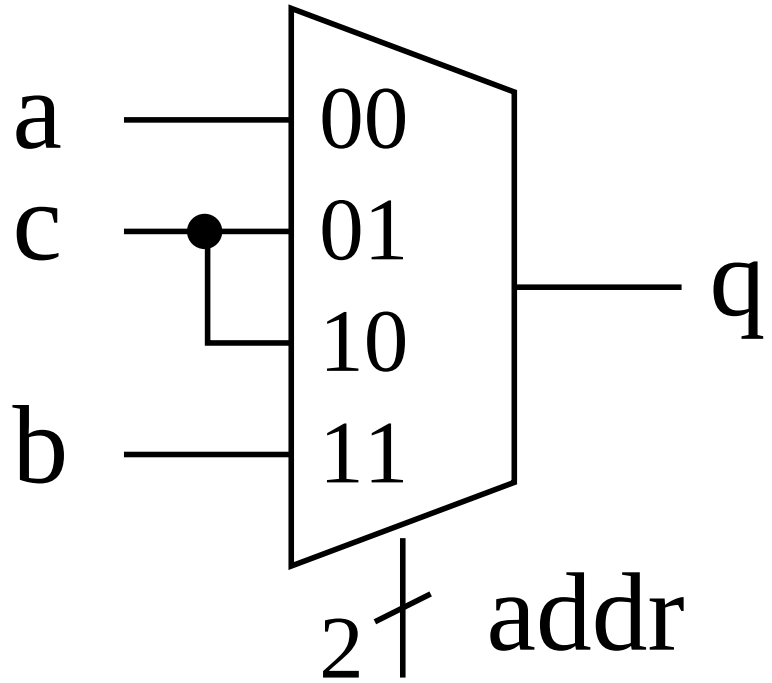
```
q6 <= a sra 1;
-- q6 = "00110"
```

Nachbildung einer Shift Operation

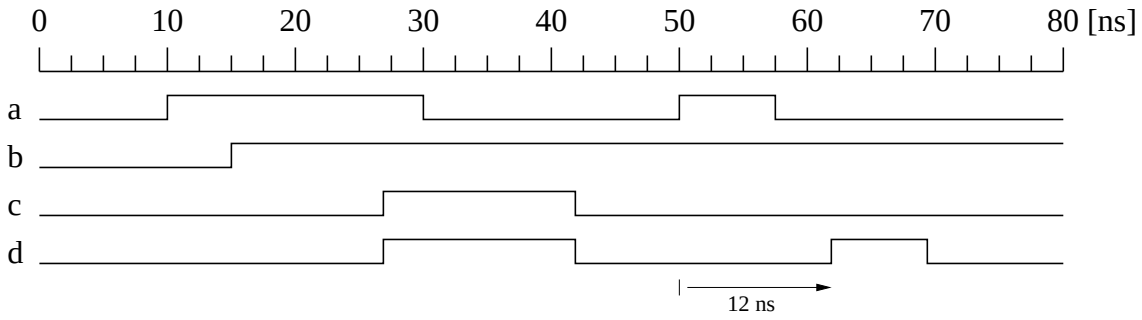


Bedingte Signalzuweisung

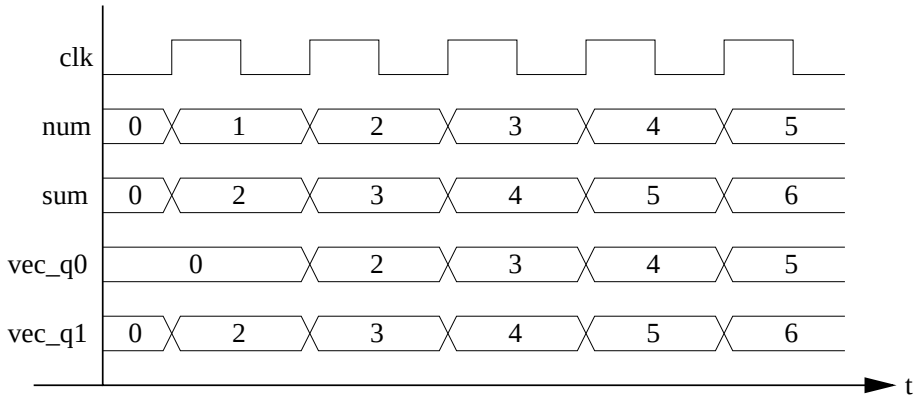
Resultierender Multiplexer:

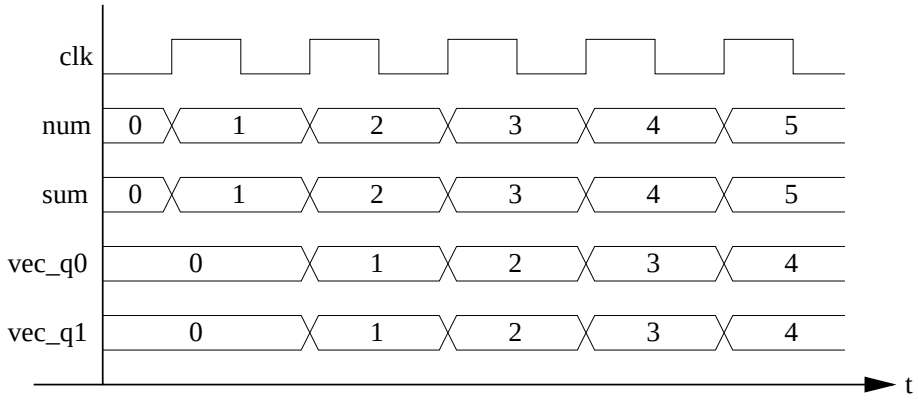


Verzögerungsmodelle



Simulationsergebnis

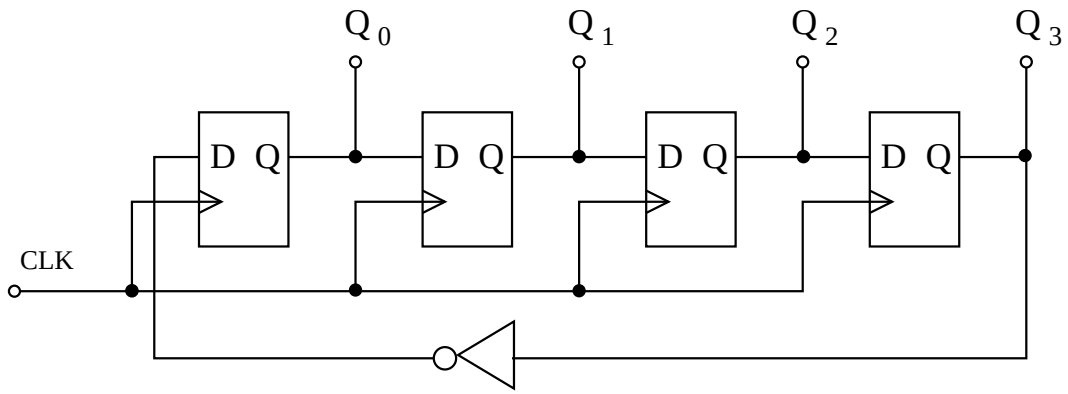




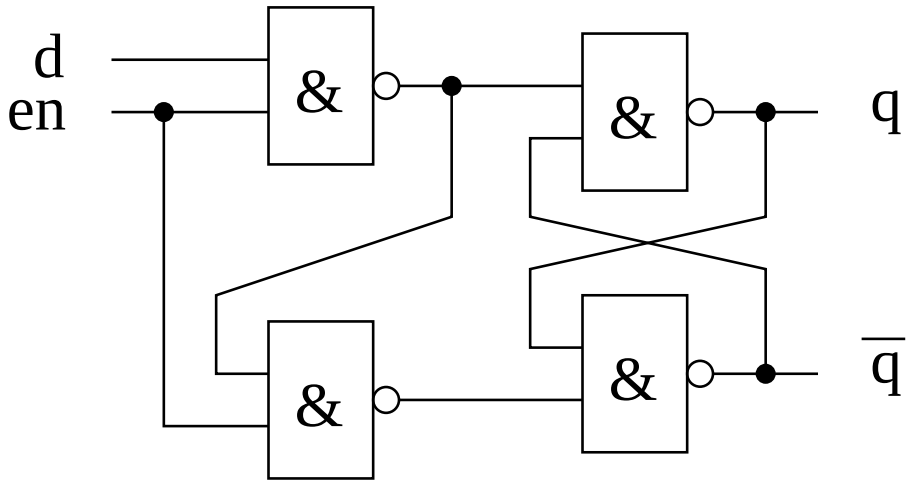
Vorlesung
Hardware-Beschreibungssprachen

Kapitel 5 : Entwurfsbeispiele

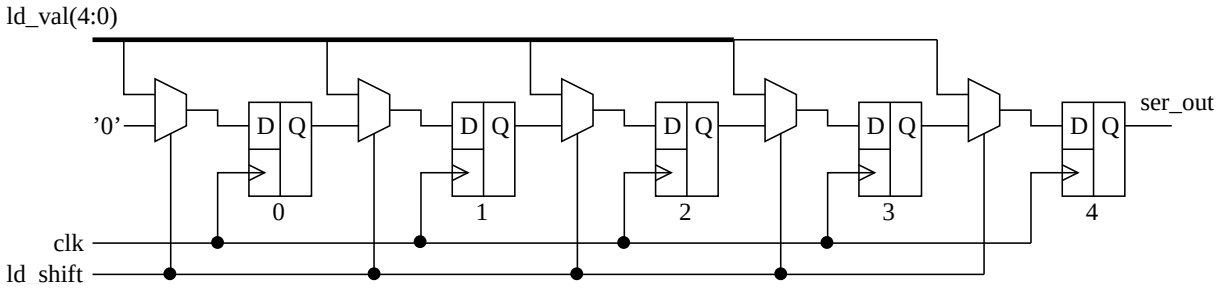
Johnson Zähler



Transparent Latch



Schieberegister



7-Segment-Anzeige