

Digitaltechnik

Wintersemester 2017/2018

15. Vorlesung



TECHNISCHE
UNIVERSITÄT
DARMSTADT





1. Klausurorganisation
2. Klausurinhalt
3. Thematische Wiederholung

Agenda

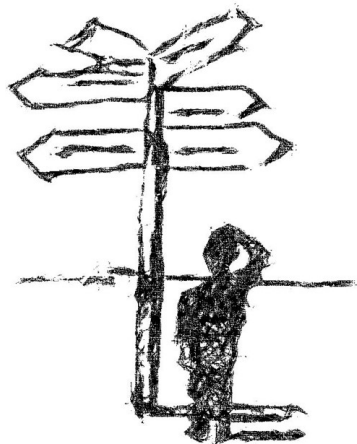


TECHNISCHE
UNIVERSITÄT
DARMSTADT

1. Klausurorganisation

2. Klausurinhalt

3. Thematische Wiederholung





- ▶ Sprechstunden für Vorbereitungsfragen
 - ▶ Raad Bahmani (S4|14 R4.1.05): 08.+22.02.18 jeweils 10:00 - 12:00
 - ▶ Andreas Engel (S2|02 E106): 07.+14.+21.02.18 jeweils 14:00 - 16:00



- ▶ Sprechstunden für Vorbereitungsfragen
 - ▶ Raad Bahmani (S4|14 R4.1.05): 08.+22.02.18 jeweils 10:00 - 12:00
 - ▶ Andreas Engel (S2|02 E106): 07.+14.+21.02.18 jeweils 14:00 - 16:00
- ▶ spätestens KW 8: Raumeinteilung im Moodle bekannt gegeben
 - ▶ S1|01 A1, A4, A5, A01, A03, A04
 - ▶ S1|05 122 (Maschinenhaus)
 - ▶ S2|02 C205 (Piloty)
 - ▶ S2|06 030 (Physik)



- ▶ Sprechstunden für Vorbereitungsfragen
 - ▶ Raad Bahmani (S4|14 R4.1.05): 08.+22.02.18 jeweils 10:00 - 12:00
 - ▶ Andreas Engel (S2|02 E106): 07.+14.+21.02.18 jeweils 14:00 - 16:00
- ▶ spätestens KW 8: Raumeinteilung im Moodle bekannt gegeben
 - ▶ S1|01 A1, A4, A5, A01, A03, A04
 - ▶ S1|05 122 (Maschinenhaus)
 - ▶ S2|02 C205 (Piloty)
 - ▶ S2|06 030 (Physik)
- ▶ ggf. **Nachteilsausgleich** bei Prüfungskommission (Tim Neubacher) anmelden



- ▶ Sprechstunden für Vorbereitungsfragen
 - ▶ Raad Bahmani (S4|14 R4.1.05): 08.+22.02.18 jeweils 10:00 - 12:00
 - ▶ Andreas Engel (S2|02 E106): 07.+14.+21.02.18 jeweils 14:00 - 16:00
- ▶ spätestens KW 8: Raumeinteilung im Moodle bekannt gegeben
 - ▶ S1|01 A1, A4, A5, A01, A03, A04
 - ▶ S1|05 122 (Maschinenhaus)
 - ▶ S2|02 C205 (Piloty)
 - ▶ S2|06 030 (Physik)
- ▶ ggf. Nachteilsausgleich bei Prüfungskommission (Tim Neubacher) anmelden
- ▶ Durchführung am 27.02.2018 ab 11:00 Uhr



- ▶ Sprechstunden für Vorbereitungsfragen
 - ▶ Raad Bahmani (S4|14 R4.1.05): 08.+22.02.18 jeweils 10:00 - 12:00
 - ▶ Andreas Engel (S2|02 E106): 07.+14.+21.02.18 jeweils 14:00 - 16:00
- ▶ spätestens KW 8: Raumeinteilung im Moodle bekannt gegeben
 - ▶ S1|01 A1, A4, A5, A01, A03, A04
 - ▶ S1|05 122 (Maschinenhaus)
 - ▶ S2|02 C205 (Piloty)
 - ▶ S2|06 030 (Physik)
- ▶ ggf. Nachteilsausgleich bei Prüfungskommission (Tim Neubacher) anmelden
- ▶ Durchführung am 27.02.2018 ab 11:00 Uhr
- ▶ etwa KW 12: Ergebnisse in TUCaN verfügbar
- ▶ KW 13 oder 14: Klausureinsicht

Klausurdurchführung am 27.02.18 ab 11:00



TECHNISCHE
UNIVERSITÄT
DARMSTADT

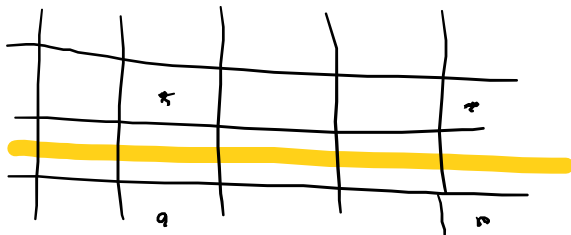
- ▶ mitbringen: Lichtbildausweis, Studierendenausweis, dokumentenechter Stift
- ▶ nicht mitbringen: Taschenrechner, Schreibblock

Klausurdurchführung am 27.02.18 ab 11:00



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ mitbringen: Lichtbildausweis, Studierendenausweis, dokumentenechter Stift
- ▶ nicht mitbringen: Taschenrechner, Schreibblock
- ▶ Sitzordnung: jeweils eine Reihe und zwei Spalten Platz halten



Klausurdurchführung am 27.02.18 ab 11:00



TECHNISCHE
UNIVERSITÄT
DARMSTADT

- ▶ mitbringen: Lichtbildausweis, Studierendenausweis, dokumentenechter Stift
- ▶ nicht mitbringen: Taschenrechner, Schreibblock
- ▶ Sitzordnung: jeweils eine Reihe und zwei Spalten Platz halten
- ▶ Verlesen der Durchführungsregeln (siehe Rückseite Deckblatt)
- ▶ gemeinsames Durchgehen aller Aufgaben
- ▶ erst danach beginnt 90 min Bearbeitungszeit



- ▶ **Anwesenheitskontrolle** *während* der Bearbeitung
 - ▶ Lichtbildausweis und Studierendenausweis griffbereit halten
 - ▶ Mitschreiben „unter Vorbehalt“, falls nicht auf Zulassungsliste
 - ⇒ Teilnahmeberechtigung im Nachgang klären



- ▶ Anwesenheitskontrolle *während* der Bearbeitung
 - ▶ Lichtbildausweis und Studierendenausweis griffbereit halten
 - ▶ Mitschreiben „unter Vorbehalt“, falls nicht auf Zulassungsliste
 - ⇒ Teilnahmeberechtigung im Nachgang klären
- ▶ Tutoren stehen für Verständnisfragen bereit
- ▶ Lösungen (inkl. Lösungsweg) direkt auf Aufgabenblättern eintragen
- ▶ bei Bedarf werden Zusatzblätter ausgegeben
- ▶ vorzeitige Abgabe bis max. 15 min vor Ende der Bearbeitungszeit
- ▶ keine Aufgabenblätter mitnehmen

Deckblatt vor der Bearbeitung ausfüllen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Name (Nach-, Vorname)	
Matrikel-Nr.	
Unterschrift	
Prüfungsordnung Bitte ankreuzen	<input type="checkbox"/> Bachelor of Science – PO 2015 <input type="checkbox"/> Bachelor of Science – PO 2009 <input type="checkbox"/> Bachelor of Science – PO 2007 <input type="checkbox"/> Bachelor of Science – PO 2004 <input type="checkbox"/> Bachelor of Education <input type="checkbox"/> Sonstiger Abschluss:
Anzahl abgegebene Zusatzblätter:	

Hilfsblatt: Einheitenvorsätze



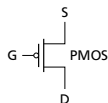
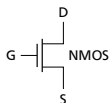
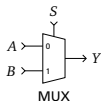
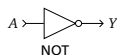
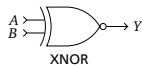
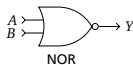
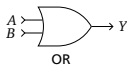
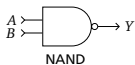
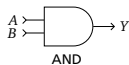
TECHNISCHE
UNIVERSITÄT
DARMSTADT

Bezeichnung	Kürzel	Wert	Bezeichnung	Kürzel	Wert	Bezeichnung	Kürzel	Wert
Milli	m	10^{-3}	Kilo	k	10^3	Kibi	Ki	2^{10}
Mikro	μ	10^{-6}	Mega	M	10^6	Mebi	Mi	2^{20}
Nano	n	10^{-9}	Giga	G	10^9	Gibi	Gi	2^{30}
Piko	p	10^{-12}	Tera	T	10^{12}	Tebi	Ti	2^{40}

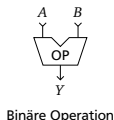
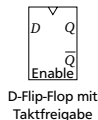
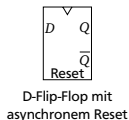
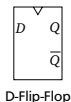
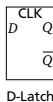
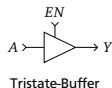
Hilfsblatt: Schaltsymbole



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Transmission Gate



Hilfsblatt: Boolesche Algebra



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Axiom	Dual	Bedeutung	Theorem	Dual	Bedeutung
A1 $B \neq 1 \Rightarrow B = 0$	A1' $B \neq 0 \Rightarrow B = 1$	Dualität	T1 $A \cdot 1 = A$	T1' $A + 0 = A$	Neutralität
A2 $\bar{0} = 1$	A2' $\bar{1} = 0$	Negieren	T2 $A \cdot 0 = 0$	T2' $A + 1 = 1$	Extremum
A3 $0 \cdot 0 = 0$	A3' $1 + 1 = 1$	Und / Oder	T3 $A \cdot A = A$	T3' $A + A = A$	Idempotenz
A4 $1 \cdot 1 = 1$	A4' $0 + 0 = 0$	Und / Oder	T4 $\bar{\bar{A}} = A$		Involution
A5 $0 \cdot 1 = 1 \cdot 0 = 0$	A5' $1 + 0 = 0 + 1 = 1$	Und / Oder	T5 $A \cdot \bar{A} = 0$	T5' $A + \bar{A} = 1$	Komplement
Theorem	Dual		Dual		Bedeutung
T6 $A B = B A$	T6' $A + B = B + A$				Kommutativität
T7 $A (B C) = (A B) C$	T7' $A + (B + C) = (A + B) + C$				Assoziativität
T8 $A (B + C) = (A B) + (A C)$	T8' $A + (B C) = (A + B) (A + C)$				Distributivität
T9 $A (A + B) = A$	T9' $A + (A B) = A$				Absorption
T10 $(A B) + (A \bar{B}) = A$	T10' $(A + B) (A + \bar{B}) = A$				Zusammenfassen
T11 $(A B) + (\bar{A} C) + (B C) = (A B) + (\bar{A} C)$	T11' $(A + B) (\bar{A} + C) (B + C) = (A + B) (\bar{A} + C)$				Konsensus
T12 $\overline{A B C \dots} = \bar{A} + \bar{B} + \bar{C} \dots$	T12' $A + B + C \dots = \bar{A} \bar{B} \bar{C} \dots$				De Morgan

Hilfsblatt: SystemVerilog



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Modul Deklaration

```
module modul_ID
#(parameter param_ID = wert)
  (input  datentyp /*[n:m]*/ in_port_ID,
   output datentyp /*[n:m]*/ out_port_ID);

// lokale Signale
datentyp /*[n:m]*/ signal_ID /*[k:l]*/;

// parallele Anweisungen
assign /* #delay */ signal = ausdruck;
always sequentielle_anweisung
submodule #(parameter_map) instanz (port_map);

// generische Anweisungen
genvar id;
generate
  if (bedingung) begin
    // lokale Signale, parallele Anweisungen
  end
  for (init; cond; step) begin
    // lokale Signale, parallele Anweisungen
  end
endgenerate
endmodule
```

Sequentielle Anweisungen

```
// Zuweisung
signal = ausdruck; // blockierend
signal <= ausdruck; // nicht-blockierend

// verzögerte Anweisungen
#delay anweisung
@(ausdruck) anweisung
@(posedge ausdruck) anweisung
@(negedge ausdruck) anweisung
@* anweisung

// bedingte Anweisungen
if (bedingung) anweisung1 else anweisung2
case (ausdruck)
  wert1 : anweisung1
  wert2 : anweisung2
  default: anweisung3
endcase

// wiederholte Anweisung
for (init; cond; step) anweisung

// kombinierte Anweisung
begin anweisung1 anweisung2 ... end
```

Hilfsblatt: SystemVerilog



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Operator	Bedeutung
[]	Zugriff auf Vektorelement
~	bitweise NOT
!	logisches NOT
-	unäre Negation
&	unäre Reduktion mit AND
	unäre Reduktion mit OR
^	unäre Reduktion mit XOR
~&	unäre Reduktion mit NAND
~	unäre Reduktion mit NOR
~^	unäre Reduktion mit XNOR
**	Exponentialfunktion
*	Multiplikation
/	Division
%	Modulo
+ -	Addition, Subtraktion
<< >>	logischer Shift
<<< >>>	arithmetischer Shift
<	kleiner als
<=	kleiner oder gleich
>	größer als
>=	größer oder gleich
==	gleich
!=	ungleich
===	bitweise gleich
!==	bitweise ungleich
& ~&	bitweise AND, NAND
^ ~^	bitweise XOR, XNOR
~	bitweise OR, NOR
&&	logisches AND
	logisches OR
?:	ternärer Operator
{ }	Konkatenation

Vertikale Gruppierung nach Präzedenz, beginnend mit der höchsten

Numerische Literale

```
// Bitbreite 'Basis Ziffernfolge
64'h0123456789abcd // hexadezimal
27'd0123456789      // dezimal
24'o01234567        // oktal
4'bxz01             // binär
// x - unbekannt/ungültig
// z - hochomig
```

Elementare Datentypen

```
bit // zweiwertige Logik
logic // vierwertige Logik
byte // 8 bit signed
integer // 32 bit signed
longint // 64 bit signed
time // 64 bit signed for Zeitwerte
real // Gleitkomma-Werte
```

System Funktionen

```
// Basis und Genauigkeit der Simulationszeit setzen
'timescale base / precision;
$time // aktuelle Systemzeit (als int)
$realtime // aktuelle Systemzeit (als real)

$log2(num) // Logarithmus zur Basis 2
$dumpfile(pfad); // VCD Ausgabedatei setzen
$dumpvars; // (alle) Signale beobachten
$finish; // Simulation beenden
$display(format, ausdrücke); // Meldung ausgeben
// %b binary format
// %c ASCII character format
// %d decimal format
// %h hex format
// %o octal format
// %s string format
// %t time format
```



- ▶ 90 Punkte verteilt auf acht Aufgaben
 - ▶ 1 Punkt / Minute
 - ▶ erreichbare Punkte für (Teil-)Aufgaben sind angegeben
 - ▶ Tipp: zu Beginn *Bearbeitungsreihenfolge individuell* festlegen (Hauptaufgaben hängen nicht voneinander ab)

- ▶ 90 Punkte verteilt auf acht Aufgaben
 - ▶ 1 Punkt / Minute
 - ▶ erreichbare Punkte für (Teil-)Aufgaben sind angegeben
 - ▶ Tipp: zu Beginn *Bearbeitungsreihenfolge individuell* festlegen (Hauptaufgaben hängen nicht voneinander ab)

▶	Notenskala	1,0	1,3	1,7	2,0	2,3	2,7	3,0	3,3	3,7	4,0	5,0
	benötigte Punkte	81	77	73	69	65	61	57	53	49	45	0

- ▶ für Studienleistung (alte PO): bestanden ab 45 Punkten
- ▶ es wird *nicht* „runterkorrigiert“
- ▶ wer beginnt, wird bewertet

Agenda

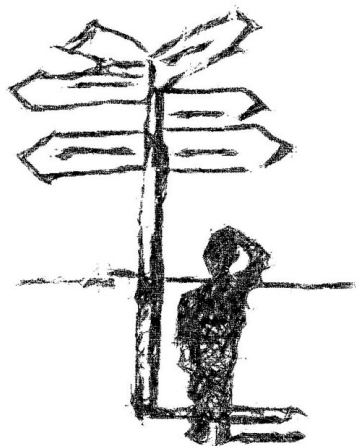


TECHNISCHE
UNIVERSITÄT
DARMSTADT

1. Klausurorganisation

2. Klausurinhalt

3. Thematische Wiederholung





- ▶ Prüfungsrelevanter Stoff
 - ▶ V1 bis V14
 - ▶ Ü1 bis Ü13 (ohne Zusatzaufgaben)



- ▶ Prüfungsrelevanter Stoff
 - ▶ V1 bis V14
 - ▶ Ü1 bis Ü13 (ohne Zusatzaufgaben)
- ▶ Aufgabentypen
 - ▶ 1 × Verständnisfragen
 - ▶ 6 × ähnlich Übungsaufgaben
 - ▶ 1 × Transferaufgabe



- ▶ Vorlesungsfolien und Übungsblätter
- ▶ Referenzliteratur (hauptsächlich Harris 2013)
- ▶ nicht bereitgestellt werden
 - ▶ Altklausuren
 - ▶ Testate



- ▶ Vorlesungsfolien und Übungsblätter
- ▶ Referenzliteratur (hauptsächlich Harris 2013)
- ▶ nicht bereitgestellt werden
 - ▶ Altklausuren
 - ▶ Testate
- ▶ für (System)Verilog
 - ▶ <https://www.mikrocontroller.net/articles/Verilog>
 - ▶ <http://www.chipverify.com/verilog-tutorial>
 - ▶ <http://www.chipverify.com/system-verilog/system-verilog>

SystemVerilog Tutorial

<http://www.chipverify.com>



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Preface
Basics
Data-types
Arrays
Operators
Parameters
Assignments
Control Flow
Modules
Module Instantiations
Ports
Block Statements
Structured Procedures
Functions
Tasks

Simulation Guide

Introduction
Initial
Always
Assign
More will come soon !

Examples

Hello World!
D Flip-Flop with Sync. Reset
D Flip-Flop with Async. Reset
D Latch
JK Flip-Flop
Positive Edge Detector

```
module tb;
    dff dff0 (.clk (clk),
              .d (d),
              .rstn (rstn),
              .q (q));

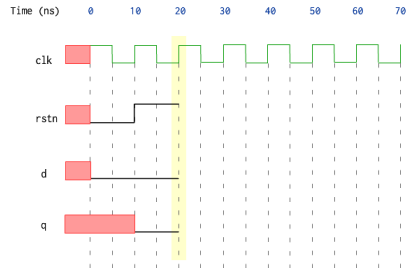
    reg d;
    reg rstn;
    reg clk;

    always #5 clk = ~clk;

    initial begin
        rstn <= 1'b0;
        clk <= 1'b1;
        d <= 1'b0;

        #10 rstn <= 1'b1;
        #10 d <= 1'b1;
        #10 d <= 1'b0;
        #20 d <= 1'b1;
        #20 $finish;
    end
endmodule
```

Simulation of a D Flip-Flop



Agenda

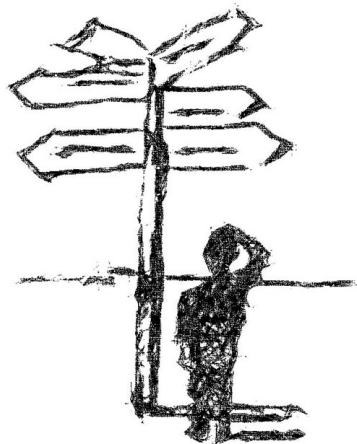


TECHNISCHE
UNIVERSITÄT
DARMSTADT

1. Klausurorganisation

2. Klausurinhalt

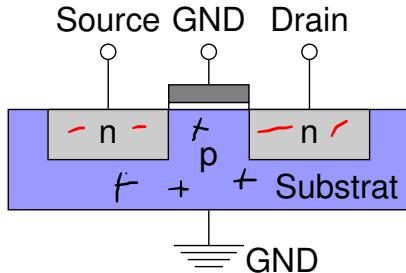
3. Thematische Wiederholung



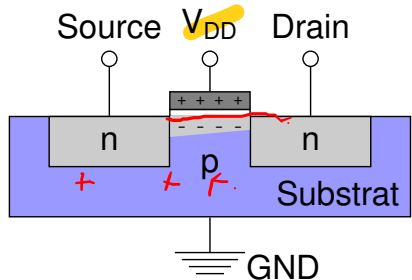
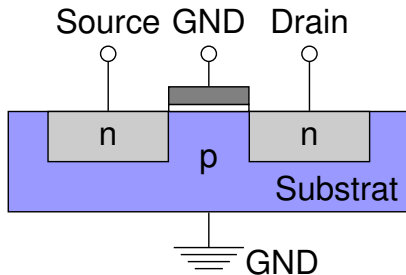


Warum leitet nMOS für Gate = 1 (V_{DD})?

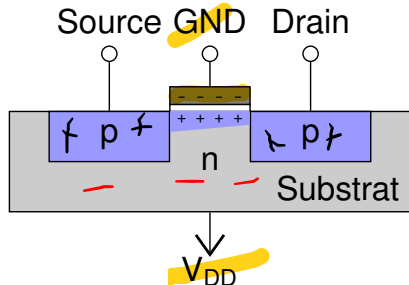
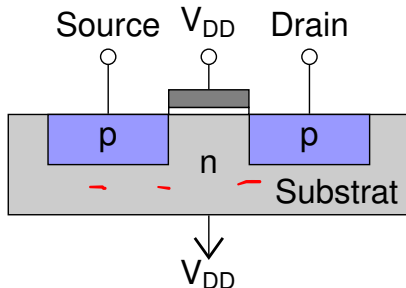
Warum leitet nMOS für Gate = 1 (V_{DD})?



Warum leitet nMOS für Gate = 1 (V_{DD})?



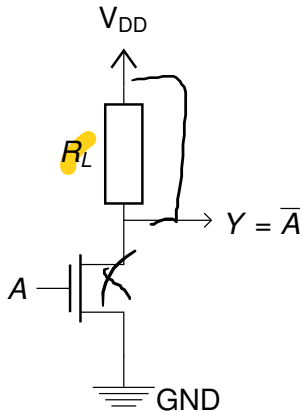
Warum leitet pMOS für Gate = 0 (GND)?



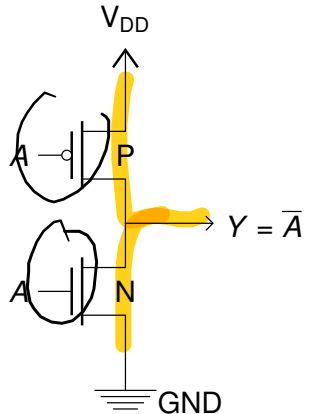
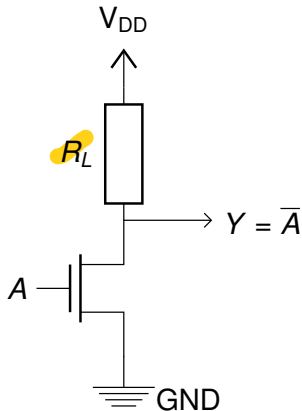


Warum hat MOSFET-Inverter statischen Leistungsumsatz?

Warum hat MOSFET-Inverter **statischen Leistungsumsatz**?



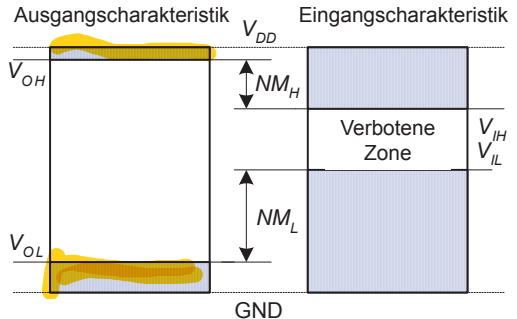
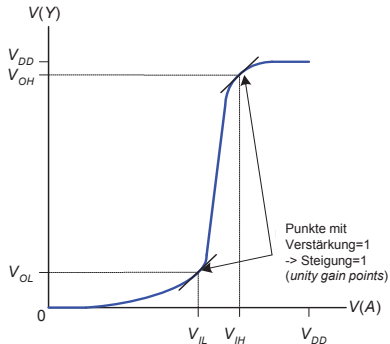
Warum hat MOSFET-Inverter statischen Leistungsumsatz?





Warum t_{pd} und t_{cd} an einem Gatter nicht immer identisch?

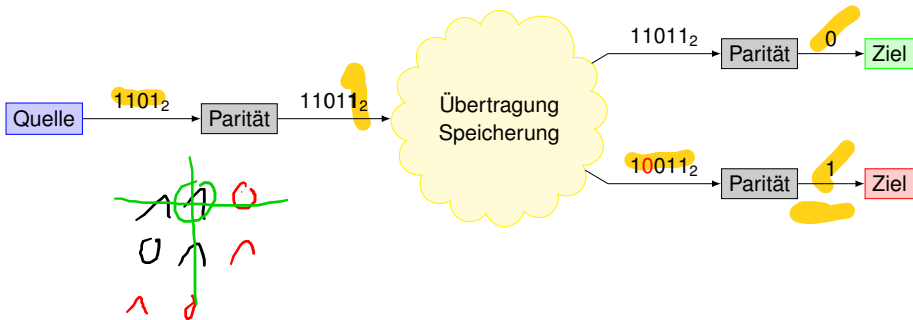
Warum t_{pd} und t_{cd} an einem Gatter nicht immer identisch?





Unterschied zwischen fehlererkennenden und fehlerkorrigierend Paritätsbits?

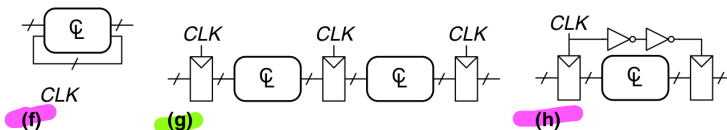
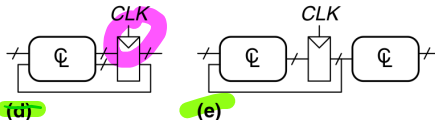
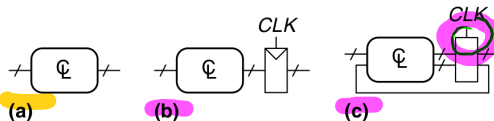
Unterschied zwischen fehlererkennenden und fehlerkorrigierend Paritätsbits?



kombinatorische / sequentielle / synchrone Schaltungen



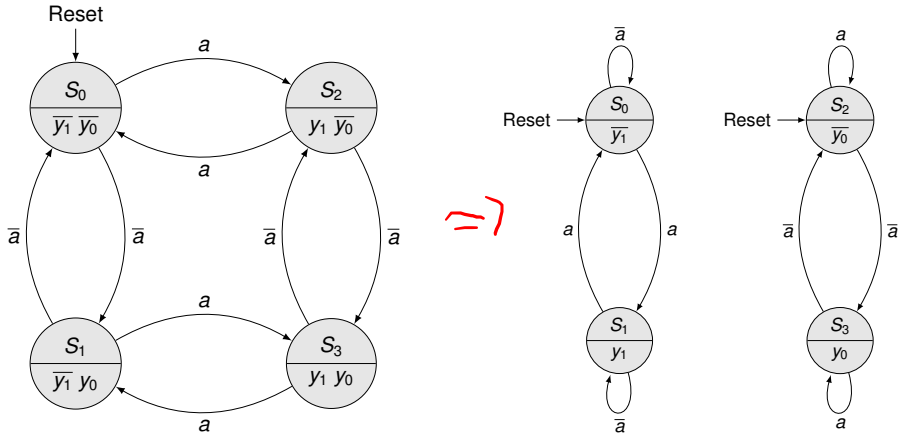
TECHNISCHE
UNIVERSITÄT
DARMSTADT



FSM Zerlegung



TECHNISCHE
UNIVERSITÄT
DARMSTADT



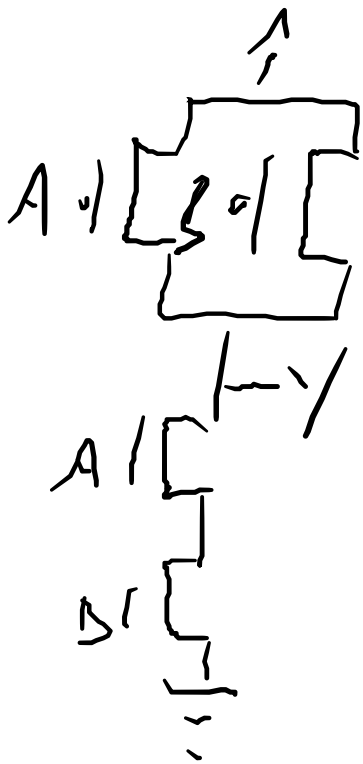
SystemVerilog Initialisierungsphase (nicht klausurrelevant)



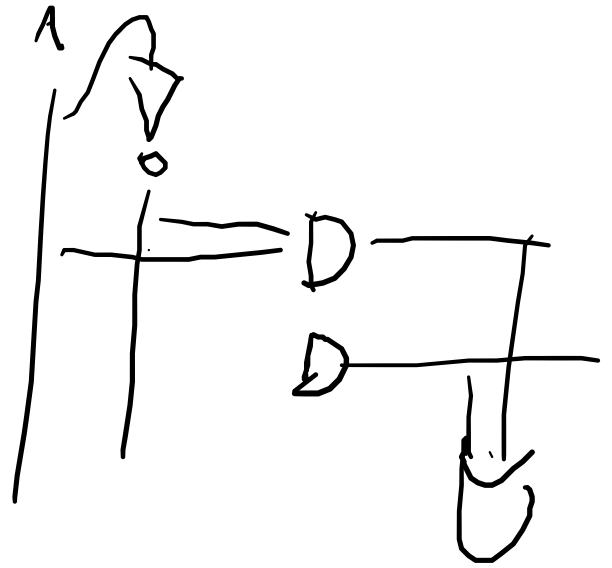
TECHNISCHE
UNIVERSITÄT
DARMSTADT

init_events.sv

```
1 logic a = 1'b0;
2 always #1 a=!a;
3 always @a $display("change a at %t", $time); // 0ns, 1ns, 2ns
4 always @(posedge a) $display("posedge a at %t", $time); // 1ns,
5 always @(negedge a) $display("negedge a at %t", $time); // [0ns,] 2ns
6
7 logic b = 1'bx;
8 always #1 b=!b;
9 always @b $display("change b at %t", $time);
10 always @(posedge b) $display("posedge b at %t", $time);
11 always @(negedge b) $display("negedge b at %t", $time);
12
13 logic c = 1'bx;
14 initial #1 c = 1'b0;
15 always @c $display("change c at %t", $time); // 1ns
16 always @(posedge c) $display("posedge c at %t", $time);
17 always @(negedge c) $display("negedge c at %t", $time); // 1ns
```



A	B	\overline{AB}
0	0	1
0	1	1
1	0	1
1	1	0



A D0

$\overline{A} = D0 \sim$

NOT

$$\overline{AB} = \overline{A} + \overline{B}$$

$\overline{A} \overline{B} = D0 \sim D0$

AND

$A \vee B = D0 \sim D0$

OR

Weitere Fragen



TECHNISCHE
UNIVERSITÄT
DARMSTADT

