

Semestralklausur

Einführung in Computer Microsystems

07. Juli 2008

Dr.-Ing. Wolfgang Heenes



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Name (Nachname, Vorname)	
Matrikelnummer	
Unterschrift	
Prüfung Bitte ankreuzen	<input type="checkbox"/> Bachelor of Science – Informatik <input type="checkbox"/> Diplom – Informatik <input type="checkbox"/> Bachelor of Computational Engineering – Informatik <input type="checkbox"/> Bachelor of Education – Informatik <input type="checkbox"/> Bachelor Informationssystemtechnik – Informatik <input type="checkbox"/> Sonstiger Abschluss:
Anzahl abgegebene Zusatzblätter:	

Aufgabe	Punkte	Erreicht
1	14	
2	12	
3	8	
4	7	
5	11	
6	13	
7	7	
8	18	
Summe	90	
Note		

Hinweise:

Überprüfen Sie zunächst, ob Ihre Klausur die Seiten 1 bis 9 besitzt.

Füllen Sie zuerst das Deckblatt aus und halten Sie Studenausweis und Lichtbildausweis bereit.

Sie sollten die Lösungen möglichst direkt in die Klausur eintragen. Reicht der vorhandene Platz nicht aus, so können Sie zusätzliche Blätter verwenden, die Sie zuerst mit Nachname, Vorname und Matrikelnummer kennzeichnen.

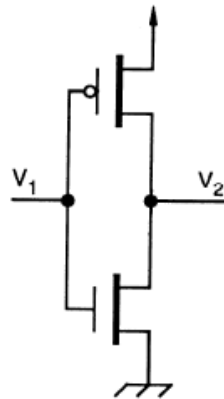
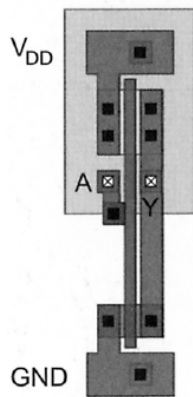
Nicht mit Bleistift schreiben und keine roten oder grünen Stifte verwenden.

Erlaubte Hilfsmittel: Formelzettel, Taschenrechner

Viel Erfolg!

Aufgabe 1: Entwurfsmethoden und Entwurfsebenen - 14 Punkte

1. Definieren Sie kurz den Begriff **Microsystem**.
2. Welcher Zusammenhang besteht zwischen dem Leistungsumsatz (P), der Spannung (U) und der Taktfrequenz (f) bei CMOS-Schaltungen?
3. Die in den folgenden drei Abbildungen dargestellten Schaltungen haben alle die gleiche Funktion. Welche Entwurfsebenen zeigen die Darstellungen und was ist die realisierte Funktion?



```
module calc_y(  
  input x,  
  output y  
);  
  assign y = ~x;  
endmodule
```

4. Nennen Sie drei grundlegende Entwurfsmethoden und erläutern Sie deren Merkmale.
5. In der CMS Vorlesung bedeutet die Abkürzung **HDL**?

Aufgabe 2: Auf-/Abwärtszähler - 12 Punkte

Gegeben ist folgender Aufwärtszähler:

```
module UpCounter(  
    input CLK,  
    input RESET,  
    output reg [3:0] COUNT  
);  
always @(posedge CLK)  
    if (RESET) COUNT <= 0;  
    else COUNT <= COUNT + 1;  
endmodule
```

1. Erweitern Sie den gegebenen Zähler zu einem ladbaren Auf-/Abwärtszähler mit programmierbarer Schrittweite. Verwenden Sie hierzu die zusätzlichen Eingänge STEP, VALUE, LOAD und DOWN, deren Funktionen im Folgenden beschrieben wird.
 - Eine „1“ auf dem Eingang STEP zeigt an, dass mit der nächsten positiven Taktflanke ein Wert vom Eingang VALUE als neue *Schrittweite* des Zählers geladen werden soll.
 - Eine „1“ auf dem Eingang LOAD zeigt an, dass mit der nächsten positiven Taktflanke ein Wert vom Eingang VALUE als neuer *Zählerstand* geladen werden soll.
 - Bei einer „0“ auf LOAD und STEP soll der Zähler bei jeder positiven Taktflanke um die programmierte Schrittweite erhöht (DOWN := „0“) bzw. erniedrigt (DOWN := „1“) werden.
 - Während eine neue Schrittweite geladen wird, soll der Zähler nicht zählen.

2. Der Chip aus 1) soll in der Fertigung billiger werden. Welcher Pin kann eingespart werden? Begründen Sie ihre Entscheidung.

Aufgabe 3: Simulation von Verilog HDL - 8 Punkte

1. Definieren Sie den Begriff Parallelität.
2. Definieren Sie den Begriff Nebenläufigkeit.

3. Gegeben ist folgendes Verilog Modul

```
module two_blocks;
initial
begin
  $display ("Ja");
  $display ("Ja");
end
initial
begin
  $display ("Nein");
  $display ("Nein");
end
endmodule
```

Geben Sie die möglichen Simulationsergebnisse an?

4. Nennen Sie zwei Arten der Simulation.

Aufgabe 4: FPGA-Technologie - 7 Punkte

1. Welches sind die logischen Grundelemente eines FPGAs?
2. Welche andere Technologie haben Sie in der Vorlesung/Übung kennengelernt und welche Vorteile (mindestens zwei Vorteile aufzählen) ergibt ergeben sich im Vergleich zum FPGA?

Aufgabe 5: Speicher, Taktteiler und Verifikation - 11 Punkte

Geben Sie bei den Aufgaben 2-5 den Lösungsweg (Formel) an.

1. Erläutern Sie die Vorteile (mindestens zwei) und einen Nachteil des SRAM-Speichers.
2. Gegeben ist ein Speicher mit 16 Bit breitem Adressbus. Der Datenbus ist 32 Bit breit. Geben Sie die Gesamtkapazität in Kilo-Byte an.
3. Gegeben ist ein 16 Bit Zähler in Verilog HDL. Der Zähler wird mit 25 MHz betrieben. Mit welcher Frequenz ändert sich das MSB.
4. Der Student F möchte den korrekten Entwurf seiner ALU durch das Testen aller möglichen Eingangssignale nachweisen. Seine ALU hat zwei 32 Bit Dateneingänge und zur Steuerung unterschiedlicher Operationen einen weiteren 8 Bit breiten Eingang. Wieviele zu testende Fälle können auftreten?
5. F möchte nun auf seiner neuen SUN Workstation alle Fälle testen. Die Workstation ist sehr leistungsfähig und schafft es in 1 ns (Nano-Sekunde) 10 Fälle zu testen. Wieviele Tage braucht F. (bzw. seine Workstation) bis er alle Fälle getestet hat?

-
2. Konstruieren Sie ein Verilog-Modul, welches dieselbe Schaltung einschließlich aller Ein- und Ausgänge mit identischer Funktionalität implementiert. Die mit FDC bezeichneten Komponenten sind vorderflankengesteuerte D-Flip-Flops mit asynchronem positivem Reset.

Aufgabe 7: Logik oder Latch? - 7 Punkte

Geben Sie für alle folgenden reg-Variablen an, ob sie bei der Synthese in Latches, Flip-Flops oder kombinatorische Logik übersetzt werden. Begründen Sie Ihre Antworten mit den Kriterien für potenzielle Register.

1. Listing 1:

```
module a (input CLOCK, I, output reg O);
  reg T;
  always @(CLOCK, I)
    if (CLOCK) begin
      T = I;
      O = T;
    end
endmodule
```

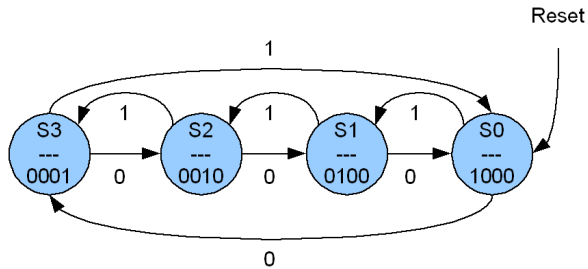
2. Listing 2:

```
module b (
  input wire A,
  input wire [2:0] I,
  output reg [7:0] OCTAL);

  always @(posedge A)
    case (I)
      4'h0: OCTAL = 8'b00000001;
      4'h1: OCTAL = 8'b00000010;
      4'h2: OCTAL = 8'b00000100;
      4'h3: OCTAL = 8'b00001000;
      4'h4: OCTAL = 8'b00010000;
      4'h5: OCTAL = 8'b00100000;
      4'h6: OCTAL = 8'b01000000;
      4'h7: OCTAL = 8'b10000000;
    endcase
endmodule
```

Aufgabe 8: Automatenbeschreibungen - 18 Punkte

Gegeben ist folgender Zustandsgraph. Die Ausgabe ist in den Zuständen angegeben.



1. Beschreiben Sie den Automaten in Verilog HDL. Als Zustandskodierung verwenden Sie eine One-Hot Kodierung. Zu Beschreibung verwenden Sie bitte die symbolischen Bezeichner der Zustände (S0 - S3). Der Reset soll asynchron implementiert werden und auf die negative Flanke reagieren.

2. Wieviele Flip-Flops benötigt man, um einen Zustandsautomaten mit 257 Zuständen in One-Hot Kodierung zu realisieren?