

Tentamen Digitalkonstruktion I, 3p

Datum: 2001-06-01

Tid: 5 timmar

Hjälpmedel: Miniräknare

Kursansvarig: Bengt Oelmann, tel: 060-148792, 060-171311, 070-2600387, e-post:
bengt@ite.mh.se

max. antal poäng: 100

antal sidor: 3

Anvisningar för inlämnade lösningar:

- Resonemang och motiveringar får ej vara så knapphändiga att de blir svåra att följa.
- Införda beteckningar skall definieras.
- Tankegången bakom uppställda ekvationer skall förklaras.
- Uträkningarna skall vara tillräckligt fullständiga för att visa hur slutresultatet erhållits.
- Approximationer ska motiveras och underkastas efterkontroll.
- Ange svaren med lämpligt antal gällande siffror
- Varje problemlösning skall avslutas med ett klart formulerat svar.

UPPGIFTER

1. Ange tre fördelar med att använda VHDL för digital konstruktion jämfört med schemainmatning. (6 p)
2. Skriv VHDL-raden för följande booleska uttryck: $a = \overline{\overline{x \oplus y} \cdot (w + \bar{z})}$. (6 p)
3. Skriv VHDL koden som beskriver en negativt flank-triggad D-vippa med asynkron reset. Skriv både entitetsdeklarationen och arkitektur. (8 p)
4. Följande VHDL kod var tänkt att beräkna udda paritet för insignalen a_bus som är av datatypen *bit_vector*. Tyvärr så fungerar inte den här koden som tänkt. Beskriv vad felet är och kom med ett förslag på en fungerande VHDL-kod. (14 p)

```
architecture wont_work of my_parity is
BEGIN
  parity: process (a_bus, x)
  begin
    x <= '0';
    for i range 'a_bus loop
      x <= a_bus(i) xor x;
    end loop;
  end process parity;
end wont_work;
```

5. Till vilket värde initieras signalen s till vid uppstart av simulering ? (6 p)

```
type sigtype is (z_state, x_state, y_state, reset);

signal s: sigtype;
```

6. Hur många vippor ger följande VHDL kod upphov till vid syntes ? (8 p)

```
entity uppg6 of rtl is
  port( a: in std_logic_vector(4 downto 0);
        b: buffer std_logic_vector(4 downto 0);
        q: out std_logic_vector(4 downto 0));
end;

architecture rtl of uppg6 is
  signal a, b: std_logic_vector(4 downto 0);
begin
  process(clk)
  begin
    if clk='1' and clk'event then
      q <= a + b;
      b <= a;
    end if;
  end process;
end;
```

7. Skriv en VHDL modell av en 64 bitars ALU som har funktionen enligt tabellen. ALU:n kan utföra sju operationer på två operander med 64 bitars ordbredd vardera. Signalgränssnittet står beskriven nedan. ALU:n måste vara fullständigt kombinatorisk. (16 p)

S	F
000	00 ... 0
001	$B - A - 1 + \text{CIN}$
010	$A - B - 1 + \text{CIN}$
011	$A + B + \text{CIN}$
100	$A \text{ xor } B$
101	$A \text{ and } B$
110	$A \text{ or } B$

```

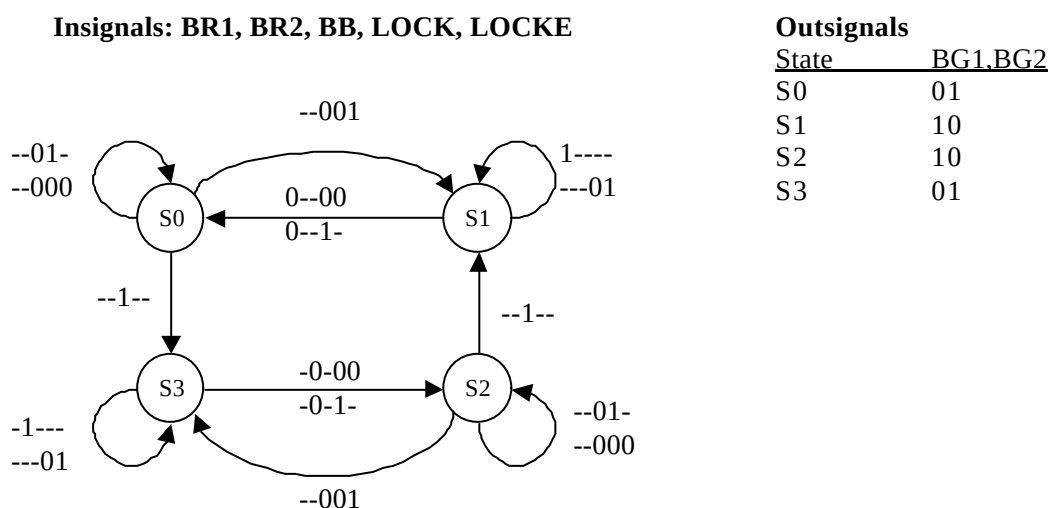
component ALU
port (
  A, B : in  std_logic_vector(63 downto 0);
  S     : in  std_logic_vector(2 downto 0);
  CIN   : in  std_logic;
  F     : out std_logic_vector(63 downto 0);
  COUT  : out std_logic);
end component;

```

8. Skriv en syntetiserbar VHDL beskrivning av en 16 bitars räknare med synkron laddningsfunktion (ett räknarvärde c ska kunna laddas till räknaren då insignalen ld är hög) samt asynkron reset (rst). Insigalner: ld , $c[15:0]$, rst , clk . Utsigalner $q[15:0]$ (räknarens värde). (16 p)

9. Skriv en VHDL beskrivning (entitet och arkitektur) för nedanstående tillståndsgraf (en bussfördelningskontroller för Motorola 68040). Beskrivningen ska vara skriven för syntes, d.v.s. inga latchar eller vippor ska genereras i den kombinatoriska delen av kretsen. Gränssnitt: insigalner: $BR1$, $BR2$, $LOCK$, $LOCKE$, BB , clk utsigalner: $BG1$, $BG2$. (20 p)

Tillståndsgraf:



- är don't care

insigalnerna är angivna i vektorform [BR1,BR2,BB,LOCK,LOCKE]

utsigalnerna är angivna i vektorform [BG1,BG2]