

Tentamen i  
Digitala system - EITA15 15hp  
varav denna tentamen 4,5hp

Institutionen för elektro- och informationsteknik  
Campus Helsingborg, LTH

2019-01-18      13.00 - 18.00 (förlängd 19.00)

Uppgifterna i tentamen ger totalt 60 poäng. Uppgifterna är inte ordnade på något speciellt sätt. Läs därför igenom alla uppgifter innan du börjar lösa dem. Några uppgifter är uppdelade i deluppgifter. Av totalt 60 möjliga poäng fordras minst 30 för godkänt.

Betygsgränser:

- 30p - 39p ger betyg 3
- 40p - 49p ger betyg 4
- 50p - 60p ger betyg 5

Inga hjälpmedel är tillåtna

**Observera!**

- För att rättning av lösning skall komma i fråga fordras att den är läslig samt klart och tydligt uppställd.
- Glöm inte att skriva personlig identifierare på varje blad.
- Alla lösa blad ska vara samlade i omslaget.
- Lösningarna ska vara numrerade och ordnade i nummerföljd.
- Påbörja ny uppgift på nytt papper.

Lycka till!



1. (a) Omvandla  $82_{10}$  till basen 2 (2 p)
- (b) Omvandla  $82_{10}$  till basen 16 (2 p)
- (c) Vilket är talområdet för ett fyra-bitars tal med teckenbit respektive utan teckenbit? (2 p)
- (d) Negativa tal representeras ofta med 2-komplement. Skriv 2-komplementrepresentationen av talet  $82_{10}$ , där  $82_{10}$  är representerat som ett 8-bitars positivt tal (2 p)
- (e) Utför subtraktionen av  $42_{10} - 75_{10}$  binärt och med hjälp av tvåkomplement. Visa tydligt hur du räknar och ge svaret i basen 10. (2 p)

2. Lite olika D-vippor skrivna i VHDL. Beskriv beteendet för de olika vipporna med avseende på clock och reset (asynkront, synkront, aktiv hög, aktiv låg, positiv flank samt negativ flank).

```
A:
—Syntax D-vippa:
reg:process(clock)
begin
  if (clock'event and clock='1') then
    Q <= D;
  end if;
end process;
```

```
B:
—Syntax D-vippa:
reg:process(clock, reset)
begin
  if (reset = '1') then
    Q <= '0';
  elsif rising_edge(clk) then
    Q <= D;
  end if;
end process;
```

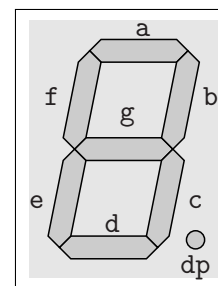
```
C:
—Syntax D-vippa:
reg:process(clock)
begin
  if rising_edge(clock) then
    if reset = '0' then
      Q <= '0';
    else
      Q <= D;
    end if;
  end if;
end process;
```

```
D:
—Syntax D-vippa:
reg:process(clock, reset)
begin
  if (reset = '0') then
    Q <= '0';
  elsif (clock'event and clock='0') then
    Q <= D;
  end if;
end process;
```

A: och B:ger vardera 2p medan C: och D: ger 3p.

**Förutsättningar för uppgift 3 och 4.**

Studenten Ture är lite sent ute med att skicka ut julkort på det traditionella sättet och vill därför göra något annorlunda. Han vill skicka en krets som innehåller en 7-segmentsdisplay där texten GOD JUL skrollas fram. Displayen framgår av figur 1 som han klistrar fast på kortet. Till detta ska han bygga dels en drivkrets som översätter från ett tre-bitars ord till bokstäverna han behöver och dels en krets som genererar sekvensen av de tre-bitars ord, så rätt text kommer fram på displayen. Displaykretsen har åtta ben som motsvarar de åtta segmenten a till f (se figur 1), där en etta gör att motsvarande segment tänds.



Figur 1: 7-segmentsdisplay.

3. Ture frågar sin kompis Lisa hur han ska göra och hon hjälper honom med hur han ska mappa tre-bitars ord och bokstav för att enkelt kunna realisera resten. I nedanstående tabell är mappningen som Lisa kom fram till (000 används inte och 110 ska ge ett blankt tecken, dvs inga segment tända).

Bitar	Bokstav	Bitar	Bokstav
000	—	100	G
001	O	101	J
010	D	110	␣
011	U	111	L

Tabell 1: Tabell över mappning mellan tre-bitars ord och bokstäver

Realisera ett kombinatoriskt nät som omvandlar från tre-bitars ord till displaysignaler. (10 p)

4. För att generera en sekvens på displayen som ska visa GOD JUL␣(␣=blanktecken), behövs någon form av sekvensnät som presenterar bokstäverna i föregående uppgift i tur och ordning. Konstruera en lämplig sekvensmaskin som skriver ut  $[GODJUL\_]^\infty$ . Notera att det inte är mellanrum mellan GOD och JUL. Denna krets kommer att klockas med ca. 1Hz (lagom takt för att hinna läsa) och utsignalen driva en display enligt tidigare uppgift. (10 p)

5. (a) Antag ett datorsystem med ett primärminne som består av 64 bytes. Hur många bitar behövs för att specificera primärminnets adressrymd om minnet är byte-adressbart? (2 p)
- (b) Om  $a=0011\ 0010$  och  $b=1011\ 0010$ . vad blir  $c= a \& b$ ; i programspråket C? (2 p)
- (c) Förklara begreppet pipelining? (2 p)
- (d) Vilka parametrar behövs för att beskriva exekveringstid i antalet klockcykler? (4 p)
- 
6. (a) Vilken uppgift har en schedulern i ett operativsystem? (2 p)
- (b) Vad menas med polling? (2 p)
- (c) Hyr många byte behövs för att lagra ett heltalsvärde mellan 0 - 65535? (2 p)
- (d) Om ordet (16 bitar) är givet till  $_{MSB} 1111\ 0001\ 1010\ 1100\ _{LSB}$ . Hur lagras det i minnet på adress  $n$  och  $n+1$  i formatet Big-endian? (4 p)