

Tentamen i Digital o Datorteknik för E, GU, IT, Z. 2006-10-23

Kortform av lösningar till tentan. För full poäng krävs fullständiga lösningar enligt typtentan

1a) $R=X-Y$ utförs som $R=X+Y_{1k}+1$; $Y_{1komp} = 001110$.

	011111
X	01101
+Y _{1komp}	+ 00110
=R	= 10100

1b) $N=1$; $Z=0$; $V=1$; $C_5=0 \Rightarrow C=1$

1c) $X=13$ $Y=25$; $R=20$ (Kontroll: $13-25=20$???); verkar rimligt ty $C=1$
 $C=1$ anger att resultatet är fel vid tal utan tecken

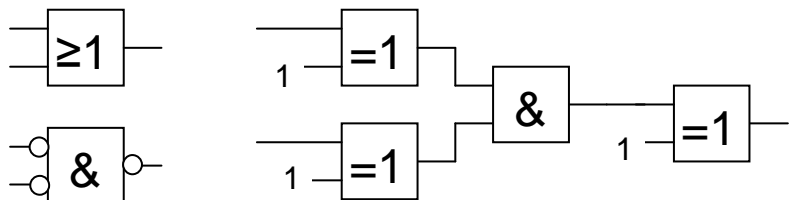
1d) $X=13$; $Y=-7$; $R=-12$ (Kontroll: $13-(-7) = -12$???); verkar rimligt ty $V=1$
 V anger fel vid tal med tecken.

1e) 1) Nej-Ja; 2) Ja – Nej; 3) Ja – Ja; 4) Ja – Ja; 5) Nej – Ja; 6) Ja - Ja

1f) $2^6=64$ vilket ger talområdet $[-32_{10}, 31_{10}]$ för tvåkomplementstal. Vi kan därför *inte* representera 32_{10} som ett 6-bitars tvåkomplementstal.

Uppg 2

2a) Vi vet att en OR-grind är samma sak som en NAND-grind med inverterare på ingångarna (MORGAN)



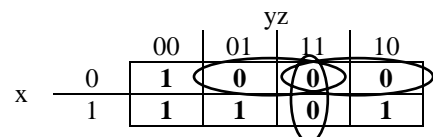
2b) Se blåa boken del 1 exempel 5.14

2c)

xyz	F
000	1
001	0
010	0
011	0
100	1
101	1
110	1
111	0

Konjunktiv normal form: $f = (x+y+z')(x+y'+z)(x+y'+z')(x'+y'+z')$

Konjunktiv minimal form:
 $f = (x+z')(x+y')(y'+z')$

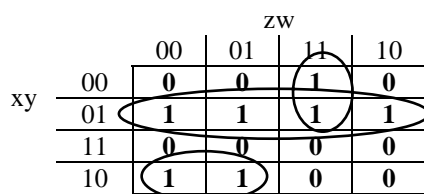


2d) Se blåa boken del 1 sidan 4-3.

Uppg 3 a)

Minimerat blir $f=(x'y)+(xy'z')+(x'zw)$

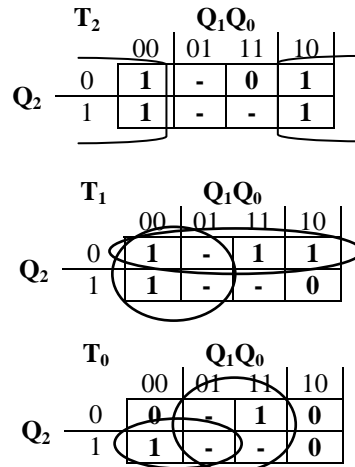
Rita nätet med NAND/NAND-logik



xyzw	F
0000	0
0001	0
0010	0
0011	1
0100	1
0101	1
0110	1
0111	1
1000	1
1001	1
1010	0
1011	0
1100	0
1101	0
1110	0
1111	0

Uppg 3b)

Detta Tillst	Nästa tillst			
$q_2^+ q_1^+ q_0^+$	$q_2^+ q_1^+ q_0^+$	T_2	T_1	T_0
000	110	1	1	0
001	---	-	-	-
010	100	1	1	0
011	000	0	1	1
100	011	1	1	1
101	---	-	-	-
110	010	1	0	0
111	---	-	-	-



Rita figur med följande insignaler till vipporna

$$T_{2_1} = Q_0' \quad | \quad T_{2_0} = Q_2' + Q_1' \quad | \quad T_2 = Q_{0_1} + Q_2 Q_1'$$

Uppg 4

4a) Se sid 189; JSR ADR

4b)

- 0) Förbered för läsning av adressoperand i minnet, Öka PC med ett, Minska stackpekaren
- 1) Läs adressoperanden från minnet till register T
- 2) Förbered för skrivning till stack
- 3) Spara återhopsadress på stacken och flytta adressoperanden till R
- 4) Ge PC det nya värdet (Adressoperanden)

Instruktionen är JSR Adr

4c)

State nr	RTN-beskrivning	Styrsignaler (=1)
0	2B → R	OE _B , f ₃ , f ₁ , f ₀ , LD _R ,
1	R → T	OE _R , LD _T
2	A-T → R	OE _A , f ₃ , f ₂ , LD _R , Cin,
3	2R → R,	OE _R , f ₃ , f ₁ , f ₀ , LD _R ,
4	R → A	OE _R , LD _A ,

Uppg 5

5a)

- INC \$A7 Öka lägsta o Kolla Carry
- BCC SLUT Hoppa om noll
- INC \$A6 annars öka mellersta
- BCC SLUT Hoppa om noll
- INC \$A5 Annars Öka högsta
- SLUT NOP

5b) Hopp sker för Q [0,134]

5c) Hopp sker för Q [128,134]

Upp 6b

```
Start      LDX  #SegCode  Pekare till tabell
           LDAB Inport  Läs inporten
           ANDB #$0F   Maskera b7-b4
           TSTB        Någon fördröjning??
           BEQ  SLUT   Hoppa om NEJ

           LDAA B,X    Hämta kod
           STAA Utport  . och visa

Loop       JSR  DELAY1s  .. och vänta
* (Förutsätter att DELAY inte ändrar Register A)

           DECB        Mera delay?
           BNE  Loop    Hoppa om JA

SLUT       STAB Utport  Slut och SLÄCK
           NOP - eller JMP Start
```