



Tentamen

EDA432 Digital- och datorteknik, IT DIT790 Digital- och datorteknik, GU

Måndag 18 Oktober 2010, kl. 8.30 - 12.30

Examinatorer

Rolf Snedsböl, tel 772 16 65

Kontaktperson under tentamen

Rolf Snedsböl, tel 772 16 65

Tillåtna hjälpmedel

Häften

Instruktionslista för FLEX

Instruktionslista för CPU12

I dessa får rättelser och understrykningar vara införda, inget annat.

Tabellverk och miniräknare får ej användas!

Lösningar

se kursens hemsida.

Granskning

Tid och plats anges på kursens hemsida.

Allmänt

Tentamen är uppdelad i del A och del B. På del A kan 30 poäng uppnås och på del B 20 poäng. Totalt 50 poäng på del A och del B tillsammans. För att del B av tentamen skall granskas och rättas krävs minst 20 poäng på del A.

Del A bedöms och betygssätts utifrån bifogat svarsblankett. Poängsättning på del A anges vid varje uppgift. Siffror inom parentes anger poängintervallet på uppgiften. Fel svar kan ge poängavdrag. En obesvarad uppgift ger inte poängavdrag.

De olika svarsalternativen a, b, c etc. kan innehålla

- korrekt svar
- nästan korrekt svar
- mer eller mindre fel svar
- helt fel svar
- inget korrekt svarsalternativ

Svara med endast ett kryss på varje uppgift

Poängsättning på del B anges vid varje uppgift. Siffror inom parentes anger maximal poäng på uppgiften. **För full poäng krävs att:**

- redovisningen av svar och lösningar är läslig och tydlig.
- ett Lösningsblad får endast innehålla redovisningsdelar som hör ihop med en uppgift.
- lösningen ej är onödigt komplicerad.
- du har motiverat dina val och ställningstaganden
- redovisningen av hårdvarukonstruktioner innehåller funktionsbeskrivning, lösning och realisering.
- redovisningen av mjukvarukonstruktioner i assembler är dokumenterade.

Betygsättning

För godkänt slutbetyg på kursen fordras att både tentamen och laborationer är godkända. Tentamen ger slutbetyget:

$20p \leq \text{betyg 3} < 30p \leq \text{betyg 4} < 40p \leq \text{betyg 5}$

DEL A – fyll i svarsblanketten sist i tesen och lämna in denna**Uppgift 1 Talomvandling, aritmetik, flaggor och koder.**I uppgifter 1.1 t.o.m 1.4 används 5-bitars tal där $X = (01110)_2$ och $Y = (10011)_2$ **Uppgift 1.1**

Tolka X och Y som tal *utan* tecken.
Vilket av alternativen anger dess decimala motsvarighet?

a	X=-14, Y=19
b	X=18, Y=-12
c	X=18, Y=12
d	X=14, Y=19
e	X=14, Y=12
f	X=18, Y=19

Poäng på uppgiften: [-1, 1]

Uppgift 1.2

Tolka X och Y som tal *med* tecken.
Vilket av alternativen anger dess decimala motsvarighet?

a	X=18, Y=20
b	X=14, Y=-13
c	X = 14, Y=-3
d	X = 14, Y=-12
e	X = -13, Y=-3
f	X = -13, Y=20

Poäng på uppgiften: [-1, 1]

Uppgift 1.3

Utför operationen $R = X - Y$ som binär addition av Y's 2-komplement
Vilket av alternativen anger R?
Tolka X, Y och R som tal *med* tecken.

a	R=25
b	R=26
c	R=27
d	R=-5
e	R=-4
f	R=-6

Poäng på uppgiften: [-1, 1]

Uppgift 1.4

Utför operationen $R=X - Y$ som binär addition av Y's 2-komplement.
Vad blir flaggbitarna NZVC efter räkneoperationen?

a	NZVC=0011
b	NZVC=1011
c	NZVC=1110
d	NZVC=0111
e	NZVC=1100
f	NZVC=1001

Poäng på uppgiften: [-1, 1]

Uppgift 1.5

Bitmönstret 01011101 kan representera:

	ASCIIkod för en versal (stor bokstav)	Negativt 2k-tal	Positivt tecken belopps tal	Förskjuten gray-kod	Två NBCD-siffror
a	Ja	Nej	Ja	Nej	Nej
b	Nej	Nej	Ja	Nej	Ja
c	Ja	Nej	Nej	Nej	Ja
d	Nej	Nej	Ja	Ja	Nej
e	Ja	Ja	Nej	Ja	Ja
f	Nej	Ja	Ja	Ja	Nej

...Poäng på uppgiften: [-1, 1]

Uppgift 2 Kombinatorik, switchnätalgebra

Uppgift 2.1

Följande funktion är given, $f(x, y, z) = xy + \bar{x}z + yz$

Ange vilket av följande alternativ som utgör funktionen på konjunktiv normal form.

a	$f(x, y, z) = (\bar{x} + \bar{y} + \bar{z}) \cdot (\bar{x} + \bar{y} + z) \cdot (x + \bar{y} + \bar{z}) \cdot (x + y + \bar{z})$
b	$f(x, y, z) = (\bar{x} + \bar{y}) \cdot (x + \bar{z})$
c	$f(x, y, z) = (x + y + z) \cdot (x + y + \bar{z}) \cdot (\bar{x} + y + z) \cdot (\bar{x} + \bar{y} + z)$
d	$f(x, y, z) = (x + y + z) \cdot (x + \bar{y} + z) \cdot (\bar{x} + y + z) \cdot (\bar{x} + y + \bar{z})$
e	$f(x, y, z) = \bar{x}\bar{y}z + \bar{z} \cdot (x + y) + x\bar{y}\bar{z}$
f	$f(x, y, z) = (\bar{x} + \bar{y} + \bar{z}) \cdot (\bar{x} + \bar{y} + z) \cdot (x + \bar{y} + z) \cdot (x + y + z)$
g	$f(x, y, z) = \bar{x}\bar{y}\bar{z} + \bar{x}\bar{y}z + x\bar{y}z + xy\bar{z}$
h	$f(x, y, z) = (\bar{x} + \bar{z}) \cdot (x + y)$
i	$f(x, y, z) = \bar{x}yz + \bar{x}y\bar{z} + xy\bar{z} + x\bar{y}\bar{z}$

Poäng på uppgiften: [-1, 2]

Uppgift 2.2

Följande Karnaughdiagram för en boolesk funktion är givet.

Vilket av följande alternativ utgör funktionens disjunktiva minimala form?

		yz			
		00	01	11	10
	0	0	1	1	0
x	1	1	1	0	1

a	$f(x, y, z) = (x + z) \cdot (\bar{x} + y)$
b	$f(x, y, z) = \bar{x}\bar{y}z + \bar{x}yz + x\bar{y}\bar{z} + x\bar{y}z$
c	$f(x, y, z) = x\bar{y} + \bar{x}y$
d	$f(x, y, z) = (x \oplus z) + x\bar{y}$
e	$f(x, y, z) = (x \oplus z) + \bar{y}z$
f	$f(x, y, z) = (y + z) \cdot (\bar{y} + \bar{z})$
g	$f(x, y, z) = \bar{x}z + x\bar{z} + x\bar{y}$
h	$f(x, y, z) = \bar{x}z + x\bar{z} + \bar{y}z$
i	$f(x, y, z) = \bar{x}\bar{y}\bar{z} + \bar{x}y\bar{z} + xyz + x\bar{y}\bar{z}$

Poäng på uppgiften: [-1, 1]

Uppgift 2.3

Följande Karnaughdiagram är givet:

Ange vilket av följande alternativ som svarar mot den minimala booleska funktionen (konjunktiv form).

		zw			
		00	01	11	10
	00	0	1	1	1
	01	0	0	-	0
xy	11	0	1	1	0
	10	0	-	0	1

a	$f(x, y, z, w) = (\bar{z} + \bar{w}) \cdot (\bar{x} + y) \cdot (x + \bar{y} + w) \cdot (y + \bar{w})$
b	$f(x, y, z, w) = \bar{y}\bar{z} + zw + x$
c	$f(x, y, z, w) = zw + \bar{x}\bar{y}z + x\bar{y}z$
d	$f(x, y, z, w) = zw + xyz + x\bar{y}z$
e	$f(x, y, z, w) = (x + \bar{y}) \cdot (z + w) \cdot (\bar{y} + \bar{z} + w) \cdot (\bar{x} + y + \bar{w})$
f	$f(x, y, z, w) = (\bar{x} + \bar{y} + \bar{z}) \cdot (\bar{x} + \bar{y} + z) \cdot (x + \bar{y} + z) \cdot (x + y + z)$
g	$f(x, y, z, w) = (x + \bar{y}) \cdot (z + w) \cdot (\bar{y} + w) \cdot (\bar{x} + y + \bar{w})$
h	$f(x, y, z, w) = yz + \bar{z}w$

Poäng på uppgiften: [-1, 2]

Uppgift 2.4

Ett kombinatoriskt nät med nedanstående funktionstabell skall konstrueras.

Vilket av följande Karnaugh-diagram skall användas?

x	y	z	w	f
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

a)

		zw			
		00	01	11	10
xy	00	0	0	1	1
	01	0	0	0	0
	11	1	1	1	1
	10	1	1	0	0

b)

		zw			
		00	01	11	10
xy	00	0	0	1	1
	01	0	0	0	0
	11	0	0	1	1
	10	1	1	1	1

c)

		zw			
		00	01	11	10
xy	00	1	1	0	0
	01	1	1	1	1
	11	0	0	0	0
	10	1	1	0	0

d)

		zw			
		00	01	11	10
xy	00	0	0	1	1
	01	0	0	0	0
	11	0	0	0	1
	10	0	0	1	1

e)

		zw			
		00	01	11	10
xy	00	1	1	0	0
	01	1	1	1	1
	11	1	1	0	0
	10	1	1	0	0

f)

		zw			
		00	01	11	10
xy	00	0	0	1	1
	01	0	0	0	0
	11	1	1	0	0
	10	0	0	1	1

Poäng på uppgiften: [-1, 1]

Uppgift 3 Sekvensnät

Uppgift 3.1

Ange funktionstabellen för en JK vippa.

a)	b)	c)	d)	e)	f)																																																												
<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>QQ⁺</th><th>JK</th></tr> </thead> <tbody> <tr><td>00</td><td>0-</td></tr> <tr><td>01</td><td>1-</td></tr> <tr><td>10</td><td>-1</td></tr> <tr><td>11</td><td>-0</td></tr> </tbody> </table>	QQ ⁺	JK	00	0-	01	1-	10	-1	11	-0	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>QQ⁺</th><th>JK</th></tr> </thead> <tbody> <tr><td>00</td><td>-1</td></tr> <tr><td>01</td><td>-1</td></tr> <tr><td>10</td><td>1-</td></tr> <tr><td>11</td><td>1-</td></tr> </tbody> </table>	QQ ⁺	JK	00	-1	01	-1	10	1-	11	1-	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>QQ⁺</th><th>JK</th></tr> </thead> <tbody> <tr><td>00</td><td>0-</td></tr> <tr><td>01</td><td>-1</td></tr> <tr><td>10</td><td>1-</td></tr> <tr><td>11</td><td>-0</td></tr> </tbody> </table>	QQ ⁺	JK	00	0-	01	-1	10	1-	11	-0	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>JK</th><th>Q⁺</th></tr> </thead> <tbody> <tr><td>00</td><td>Q</td></tr> <tr><td>01</td><td>0</td></tr> <tr><td>10</td><td>1</td></tr> <tr><td>11</td><td>*</td></tr> </tbody> </table>	JK	Q ⁺	00	Q	01	0	10	1	11	*	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>JK</th><th>Q⁺</th></tr> </thead> <tbody> <tr><td>00</td><td>Q</td></tr> <tr><td>01</td><td>1</td></tr> <tr><td>10</td><td>0</td></tr> <tr><td>11</td><td>Q'</td></tr> </tbody> </table>	JK	Q ⁺	00	Q	01	1	10	0	11	Q'	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>JK</th><th>Q⁺</th></tr> </thead> <tbody> <tr><td>00</td><td>0</td></tr> <tr><td>01</td><td>1</td></tr> <tr><td>10</td><td>1</td></tr> <tr><td>11</td><td>Q</td></tr> </tbody> </table>	JK	Q ⁺	00	0	01	1	10	1	11	Q
QQ ⁺	JK																																																																
00	0-																																																																
01	1-																																																																
10	-1																																																																
11	-0																																																																
QQ ⁺	JK																																																																
00	-1																																																																
01	-1																																																																
10	1-																																																																
11	1-																																																																
QQ ⁺	JK																																																																
00	0-																																																																
01	-1																																																																
10	1-																																																																
11	-0																																																																
JK	Q ⁺																																																																
00	Q																																																																
01	0																																																																
10	1																																																																
11	*																																																																
JK	Q ⁺																																																																
00	Q																																																																
01	1																																																																
10	0																																																																
11	Q'																																																																
JK	Q ⁺																																																																
00	0																																																																
01	1																																																																
10	1																																																																
11	Q																																																																

.Poäng på uppgiften: [-1, 1]

Uppgift 3.2

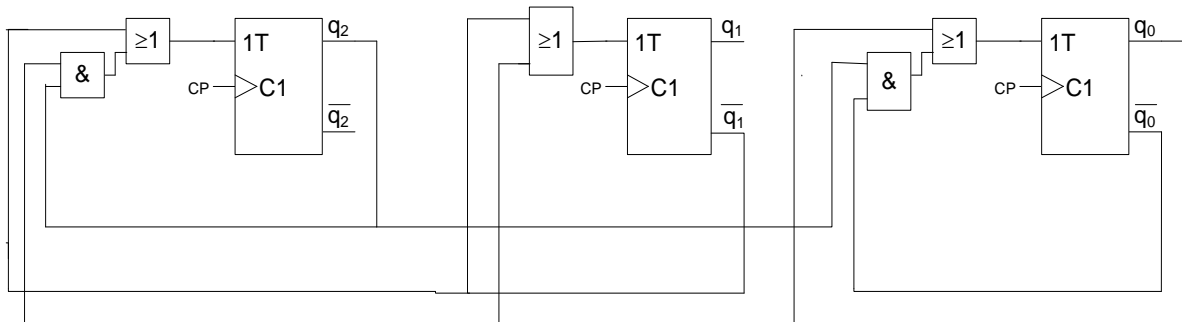
Ange excitationstabellen för en SR vippa.

a)	b)	c)	d)	e)	f)																																																												
<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>SR</th><th>Q⁺</th></tr> </thead> <tbody> <tr><td>00</td><td>1</td></tr> <tr><td>01</td><td>0</td></tr> <tr><td>10</td><td>1</td></tr> <tr><td>11</td><td>*</td></tr> </tbody> </table>	SR	Q ⁺	00	1	01	0	10	1	11	*	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>SR</th><th>Q⁺</th></tr> </thead> <tbody> <tr><td>00</td><td>Q</td></tr> <tr><td>01</td><td>0</td></tr> <tr><td>10</td><td>1</td></tr> <tr><td>11</td><td>*</td></tr> </tbody> </table>	SR	Q ⁺	00	Q	01	0	10	1	11	*	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>SR</th><th>Q⁺</th></tr> </thead> <tbody> <tr><td>00</td><td>Q</td></tr> <tr><td>01</td><td>0</td></tr> <tr><td>10</td><td>1</td></tr> <tr><td>11</td><td>Q'</td></tr> </tbody> </table>	SR	Q ⁺	00	Q	01	0	10	1	11	Q'	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>QQ⁺</th><th>SR</th></tr> </thead> <tbody> <tr><td>00</td><td>-0</td></tr> <tr><td>01</td><td>10</td></tr> <tr><td>10</td><td>01</td></tr> <tr><td>11</td><td>0-</td></tr> </tbody> </table>	QQ ⁺	SR	00	-0	01	10	10	01	11	0-	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>QQ⁺</th><th>SR</th></tr> </thead> <tbody> <tr><td>00</td><td>-0</td></tr> <tr><td>01</td><td>10</td></tr> <tr><td>10</td><td>01</td></tr> <tr><td>11</td><td>11</td></tr> </tbody> </table>	QQ ⁺	SR	00	-0	01	10	10	01	11	11	<table border="1" style="border-collapse: collapse;"> <thead> <tr><th>QQ⁺</th><th>SR</th></tr> </thead> <tbody> <tr><td>00</td><td>-0</td></tr> <tr><td>01</td><td>1-</td></tr> <tr><td>10</td><td>-1</td></tr> <tr><td>11</td><td>0-</td></tr> </tbody> </table>	QQ ⁺	SR	00	-0	01	1-	10	-1	11	0-
SR	Q ⁺																																																																
00	1																																																																
01	0																																																																
10	1																																																																
11	*																																																																
SR	Q ⁺																																																																
00	Q																																																																
01	0																																																																
10	1																																																																
11	*																																																																
SR	Q ⁺																																																																
00	Q																																																																
01	0																																																																
10	1																																																																
11	Q'																																																																
QQ ⁺	SR																																																																
00	-0																																																																
01	10																																																																
10	01																																																																
11	0-																																																																
QQ ⁺	SR																																																																
00	-0																																																																
01	10																																																																
10	01																																																																
11	11																																																																
QQ ⁺	SR																																																																
00	-0																																																																
01	1-																																																																
10	-1																																																																
11	0-																																																																

.Poäng på uppgiften: [-1, 1]

Uppgift 3.3

Analysera räknaren nedan. Vilken tabell motsvarar räknaren?



a)

Q	Q ⁺
0	6
1	6
2	-
3	0
4	3
5	2
6	7
7	0

b)

Q	Q ⁺
0	1
1	2
2	3
3	4
4	5
5	6
6	7
7	0

c)

Q	Q ⁺
0	6
1	6
2	2
3	0
4	3
5	2
6	7
7	0

$Q = q_2q_1q_0$

d)

Q	Q ⁺
0	6
1	6
2	7
3	0
4	3
5	5
6	7
7	0

e)

Q	Q ⁺
0	3
1	3
2	3
3	6
4	6
5	6
6	2
7	1

f)

Q	Q ⁺
0	6
1	6
2	5
3	0
4	3
5	4
6	7
7	0

Poäng på uppgiften: [-1, 4]

Uppgift 4 FLEX styrenhet

Uppgift 4.1

I tabellen intill visas styrsignalerna för EXECUTE-sekvensen för en **instruktion** för FLEX-processorn. NF i tabellens sista rad anger att nästa tillstånd (state) skall vara det första i FETCH-sekvensen. Vilken instruktion är det?

S	Styrsignalerna (= 1)
5	OE _X , LD _{MA} ,
6	MR, LD _T
7	OE _B , f ₃ , f ₂ , g ₀ , LD _{CC} , LD _R
8	OE _R , LD _B , NF

a	CMPB Adr	b	SUBB Adr	c	SUBB ,X
d	SUBB #Data	e	CMPB ,X	f	SBCB ,X

Poäng på uppgiften: [-1, 1]

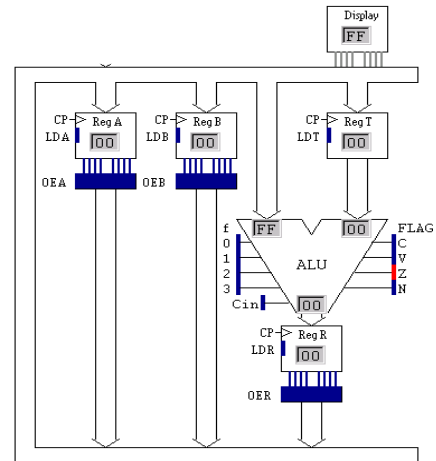
Uppgift 4.2

Ange vilken tabell som beskriver utförandet av operationen enligt nedanstående RTN-beskrivning:

RTN-beskrivning: $3A + 2B - 4 \rightarrow B$

Förutsätt att register A och B i datavägen till höger innehåller de data som skall beräknas. Register A får inte ändras. Använd så få tillstånd som möjligt.

Vilket svarsalternativ väljer du?



a

S	RTN-beskrivning
1	A→T
2	B+T→R
3	2R→R
4	R+T→R
5	R-1→R
6	R-1→R
7	R-1→R
8	A+T→R
9	R→B

b

S	RTN-beskrivning
1	FF→R
2	2R→R
3	R→T
4	B-T→R
5	R+A→R
6	2R→R
7	R→T
8	A+T→R
9	R→B

c

S	RTN-beskrivning
1	A-1→R, B→T
2	R+T→R
3	2R→R
4	R→T
5	A+T→R
6	R→B

d

S	RTN-beskrivning
1	B-1→R
2	R-1→R
3	A→T
4	R+T→R
5	2R→R
6	R+T→R
7	R→B

e

S	RTN-beskrivning
1	FF→R
2	2R→R
3	R→T
4	B+T→R
5	R+A→R
6	2R→R
7	R→T
8	A+T→R
9	R→B

f

S	RTN-beskrivning
1	A→T
2	B+T-1→R
3	R+T-1→R
4	R+A→R
5	A+T→R
6	R→B

Poäng på uppgiften: [-1, 3]

Uppgift 4.3

En instruktion för FLEX-processorn är **JSR n,X**. Se instruktionslistan för FLEX. Ange RTN-beskrivningen för utförandefasen för denna instruktion. (S anger aktuellt State)

a	b	c
S RTN-beskrivning	S RTN-beskrivning	S RTN-beskrivning
5 PC→MA, PC+1→PC	5 PC→MA, PC+1→PC	5 PC→MA, PC+1→PC, S-1→S
6 M→T, S-1→S	6 M→T,	6 M→T
7 X + T →R	7 S→MA, S-1→S	7 S→MA
8 R→S	8 X + T →R	8 PC→S
9 PC→MA	9 R→S	9 X + T →R
10 S→PC	10 M→PC	10 R→PC

d	e	f
S RTN-beskrivning	S RTN-beskrivning	S RTN-beskrivning
5 PC→MA, PC+1→PC	5 PC→MA, PC+1→PC	5 PC→MA, PC+1→PC
6 M→T, S-1→S	6 M→T	6 M→T
7 S→MA	7 S→MA, S-1→S	7 S→MA, S-1→S
8 PC→M	8 PC→M	8 PC→S
9 X + T →R	9 X + T →R	9 X + T →R
10 R→PC	10 R→PC	10 R→PC

Poäng på uppgiften: [-1, 3]

Uppgift 5 FLEX programmering

Uppgift 5.1

Vad blir maskinkoden för instruktionerna BMI och BRA?

BMI är placerad på (har sin OP-kod på) adress AA₁₆ och BRA på adress E9₁₆.
Loop är placerad på adress 23₁₆ och End på adress F7₁₆.

Loop	-
	-
BMI	End
	-
	-
BRA	Loop
	-
	-
End	-

a	b	c
BMI: 5B 23 BRA: 5A 93	BMI: 5A 29 BRA: 5B 99	BMI: 5B 27 BRA: 5A 97
d	e	f
BMI: 5B 23 BRA: 5A 67	BMI: 5B 27 BRA: 5A 99	BMI: 5B 29 BRA: 5A 99

Poäng på uppgiften: [-1, 2]

Uppgift 5.2

Ange maskinkoden för FLEX-processorn för instruktionssekvensen till höger.

	ORG	\$E4
Rut	EQU	\$17
Var	EQU	-10
	STAA	9,X
	STAA	A,X
	STAA	Var,X
	LDAB	Data
	JSR	Rut
	LDAA	#Adr
	RTS	
Adr	FCB	16,32,64
Data	RMB	1

a		b		c		d		e		f	
Adr	Maskin-kod	Adr	Maskin-kod	Adr	Maskin-kod	Adr	Maskin-kod	Adr	Maskin-kod	Adr	Maskin-kod
E4	8F	E4	17	E4	17	E4	8F	E4	8F	E4	8F
E5	09	E5	F6	E5	F6	E5	09	E5	09	E5	09
E6	91	E6	8F	E6	8F	E6	8F	E6	91	E6	91
E7	8F	E7	09	E7	09	E7	0A	E7	8F	E7	8F
E8	10	E8	8F	E8	91	E8	8F	E8	F6	E8	F6
E9	10	E9	0A	E9	8F	E9	10	E9	0C	E9	10
EA	F3	EA	8F	EA	F6	EA	F3	EA	F3	EA	F3
EB	69	EB	F6	EB	10	EB	69	EB	69	EB	69
EC	17	EC	10	EC	F5	EC	17	EC	17	EC	17
ED	0B	ED	F6	ED	69	ED	0B	ED	0F	ED	0B
EE	F0	EE	69	EE	17	EE	F0	EE	F0	EE	F0
EF	6A	EF	17	EF	0B	EF	6A	EF	6A	EF	6A
F0	10	F0	0B	F0	F2	F0	16	F0	10	F0	10
F1	20	F1	F3	F1	6A	F1	32	F1	20	F1	20
F2	40	F2	6A	F2	16	F2	64	F2	40	F2	40
F3	??	F3	16	F3	32	F3	01	F3	??	F3	??
F4		F4	32	F4	64	F4		F4		F4	
F5		F5	64	F5	01	F5		F5		F5	
		F6	??								

Poäng på uppgiften: [-1, 4]

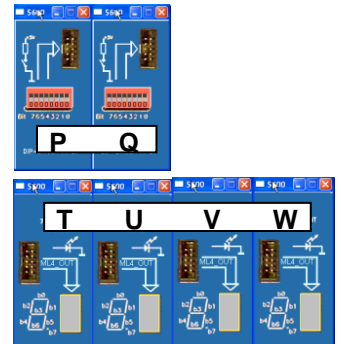
DEL B – Svara på separata ark. Blanda inte uppgifter på samma ark.

Uppgift 6

Vid simulatorpassen och i labbet använde du stömbrytarna (ML4 INPUT) och sifferindikatorn (ML4 OUTPUT).

Du skall skriva ett program för CPU12 som hela tiden

- läser ett 16 bitars tvåkomplementstal från två inportar (P och Q)
- Skriver beloppet av talet till fyra sifferindikatorer (T, U, V och W)



Alltså: Om det lästa talet är positivt skall det direkt skrivas till sifferindikatorerna. Om det lästa talet är negativt skall detta negeras (tvåkomplementeras) innan det skrivs ut till sifferindikatorerna. (Exempel: Om det lästa talet är $F68A_{16}$ så skall 0976_{16} skrivas till sifferindikatorerna ty tvåkomplementet av $F68A$ är 976)

Du har tillgång till en tabell med segmentkoder och följande definitioner:

DipSwP	EQU	\$600	Adress för strömbrytare P
DipSwQ	EQU	\$601	Adress för strömbrytare Q
SifT	EQU	\$400	Adress för Sifferindikator T
SifU	EQU	\$401	Adress för Sifferindikator U
SifV	EQU	\$402	Adress för Sifferindikator V
SifW	EQU	\$403	Adress för Sifferindikator W
SegCode	FCB	xx,yy,zz,etc	Tabell med segmentkoder för [0,F]

Skriv kommentarer i ditt program

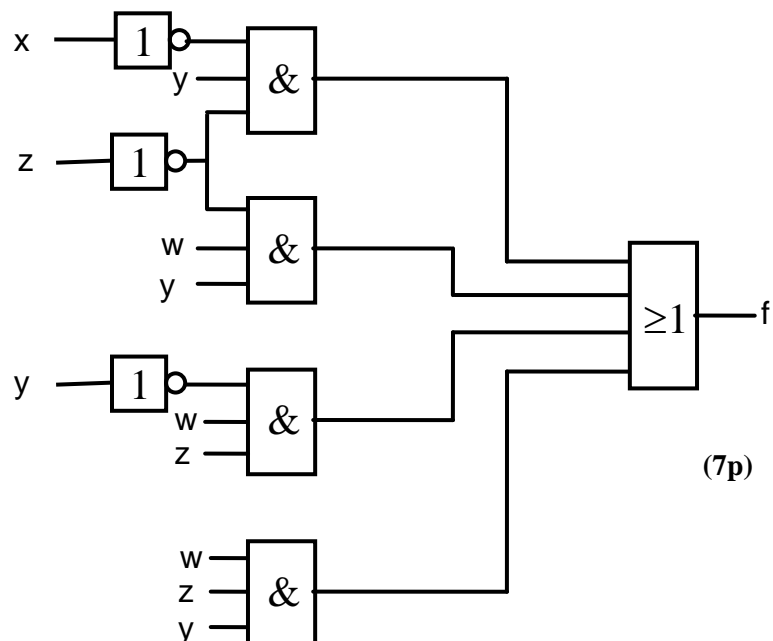
(8p)

Uppgift 7

Om det är möjligt, minimera följande grindnät och realicera en lösning på NAND/NAND-form (Disjunktiv form)

Visa varje steg i din lösning

- 1) Analys
- 2) Funktionstabell
- 3) Minimering
- 4) Realicering



(7p)

Uppgift 8

Vi önskar att bestycka en CPU12-liknande processor med en 16kByte ROM-modul och en 4kByte RWM-modul. RWM-modulen skall placeras i adressrummet med start på adress 0. ROM-modulen skall placeras i adressrummet på de högsta adresserna (slutadress FFFF). Adressbussen är 16 bitar bred och databussen 8 bitar bred. Processorn har en E-klocka och R/W'-signal på samma sätt som CPU12.

- a)** Konstruera adressavkodningslogiken och använd fullständig adressavkodning.
- b)** Hur kan du utnyttja R/W'-signalen för att skilja mellan in- och utportar. Visa med ett exempel.

(5p)

Anonym
kod:

Svarsblankett för del A

Uppg 1	uppgiften besvaras inte	inget rätt svars- alternativ	a	b	c	d	e	f	g	h	i
1.1											
1.2											
1.3											
1.4											
1.5											

poäng	

Uppg 2	uppgiften besvaras inte	inget rätt svars- alternativ	a	b	c	d	e	f	g	h	i
2.1											
2.2											
2.3											
2.4											

poäng	

Uppg 3	uppgiften besvaras inte	inget rätt svars- alternativ	a	b	c	d	e	f	g	h	i
3.1											
3.2											
3.3											

poäng	

Uppg 4	uppgiften besvaras inte	inget rätt svars- alternativ	a	b	c	d	e	f	g	h	i
4.1											
4.2											
4.3											

poäng	

Uppg 5	uppgiften besvaras inte	inget rätt svars- alternativ	a	b	c	d	e	f	g	h	i
5.1											
5.2											

poäng	

Poäng totalt	

Anonym
kod:

Svarsblankett för del A

Uppg 1	uppgiften besvaras inte	inget rätt svarsalternativ	a	b	c	d	e	f	g	h	i
1.1						X					
1.2				X							
1.3						X					
1.4				X							
1.5		X				X					

poäng	

Uppg 2	uppgiften besvaras inte	inget rätt svarsalternativ	a	b	c	d	e	f	g	h	i
2.1						X					
2.2									X	X	
2.3									X		
2.4						X					

poäng	

Uppg 3	uppgiften besvaras inte	inget rätt svarsalternativ	a	b	c	d	e	f	g	h	i
3.1		X									
3.2		X									
3.3					X						

poäng	

Uppg 4	uppgiften besvaras inte	inget rätt svarsalternativ	a	b	c	d	e	f	g	h	i
4.1					X						
4.2						X					
4.3						X					

poäng	

Uppg 5	uppgiften besvaras inte	inget rätt svarsalternativ	a	b	c	d	e	f	g	h	i
5.1		X									
5.2							X				

poäng	

Poäng totalt	

Uppgift 6

```

        ORG    $1000
        LDX    #SegCode
Loop
        LDD    DipSwP      Läs 16 bitar (P och Q)
        BPL    Positiv
        COMA
        COMB              2-komplementera
        ADDD    #1
Positiv
        PSHB
        ANDB    #$0F      Visa Low Nibble
        LDAB    B,X
        STAB    SifW
        PULB

        ANDB    #$F0
        LSRB
        LSRB              Skifta fram o
        LSRB
        LSRB              .. och visa Nibble
        LDAB    B,X
        STAB    SifV

        PSHA
        ANDA    #$0F      Visa Nibble
        MOVB    A,X,SifU
        PULB

        ANDB    #$F0
        LSRB
        LSRB              Skifta fram o
        LSRB
        LSRB              .. och visa High Nibble
        MOVB    B,X,SifT

        BRA    Loop

```

Uppgift 7

1) Analys ger: $f = x'yz' + yz'w + y'zw + yzw$

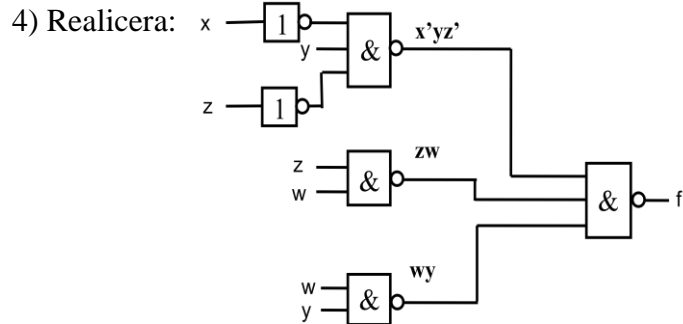
2) Funktionstabell:

x	y	z	w	f
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

3) Minimera:

	zw			
	00	01	11	10
xy	0	0	1	0
01	1	1	1	0
11	0	1	1	0
10	0	0	1	0

Ger $f = wy + zw + x'yz'$



Uppgift 8

ROM: 16kbyte $\Rightarrow 2^4 \cdot 2^{10}$ byte \Rightarrow 14 Adressbiter \Rightarrow [A13,A0] direkt till ROM-kapsel.

RWM: 4kbyte $\Rightarrow 2^2 \cdot 2^{10}$ byte \Rightarrow 12 Adressbiter \Rightarrow [A11,A0] direkt till RWM-kapsel.

	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀
ROM Start: C000	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROM Stop: FFFF	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
RWM Start: 0000 -	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RWM Stop: 0FFF	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1

ROM: Rita en NAND-grind enligt $CS_ROM' = (A_{15} \cdot A_{14} \cdot E \cdot R/W)'$

RWM: Rita en NAND-grind enligt $CS_RWM' = (A_{15}' \cdot A_{14}' \cdot A_{13}' \cdot A_{12}' \cdot E)$

Vi gör STAA till en utport och då är R/W låg. När vi gör LDAA från en inport är R/W hög. Detta kan utnyttjas när vi bildar CS-logik för I/O-portar. Ex

$$CS_InPort' = \{f(Adr_x) \cdot E \cdot R/W\}'$$

$$CS_UtPort' = \{f(Adr_x) \cdot E \cdot R/W'\}'$$