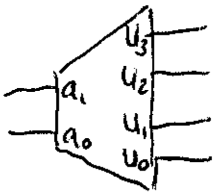
Ö5.1

Konstruera en 2-4 binär avkodare.

<u>Lösning:</u>

2-4 avkodare



sanningstabell

rad	a_1	a_0	u_3	u_2	u_1	u_0
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

avkodaren har fyra utgångar. För var och en av dessa ska vi ta fram en logisk funktion.
Eftersom uttrycken enbart innehåller <u>en</u> minterm (se sanningstabell) så kan vi direkt skriva upp dem.

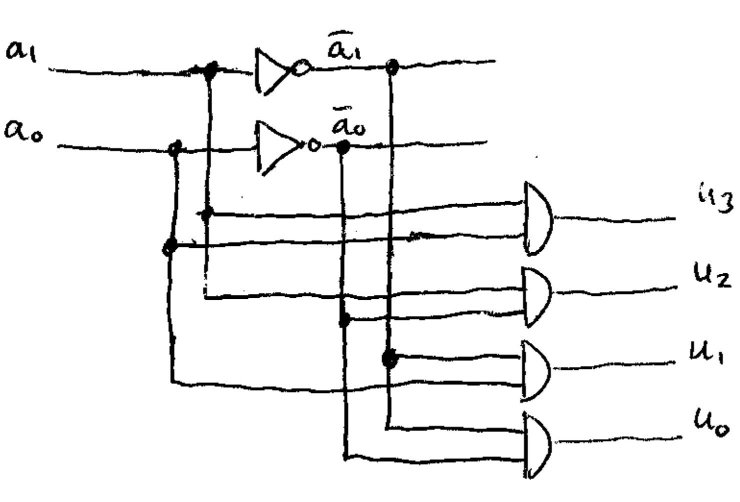
$$u_0 = \bar{a}_1\bar{a}_0$$

$$u_1 = \bar{a}_1 a_0$$

$$u_2 = a_1\bar{a}_0$$

$$u_3 = a_1 a_0$$

<u>Schema:</u>

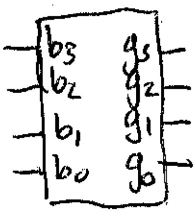


Ö5.2

Ta fram en krets som omvandlar 4 bitars binärkod till 4 bitars gray-kod.

Lösning:

Omvandlare:



$\{b_3, b_2, b_1, b_0\}$ är den binära koden
$\{g_3, g_2, g_1, g_0\}$ är gray-koden.

Sanningstabell:

rad	b_3	b_2	b_1	b_0	g_3	g_2	g_1	g_0
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	1	0
5	0	1	0	1	0	1	1	1
6	0	1	1	0	0	1	0	1
7	0	1	1	1	0	1	0	0
8	1	0	0	0	1	1	0	0
9	1	0	0	1	1	1	0	1
10	1	0	1	0	1	1	1	1
11	1	0	1	1	1	1	1	0
12	1	1	0	0	1	0	1	0
13	1	1	0	1	1	0	1	1
14	1	1	1	0	1	0	0	1
15	1	1	1	1	1	0	0	0

M.h.a K-diagram tas funktionerna g_3, g_2, g_1 och g_0 fram.

b_3b_2 \ b_1b_0	00	01	11	10
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	1	1	1	1

$g_3 = b_3$

b_3b_2 \ b_1b_0	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	0	0
10	1	1	1	1

$g_2 = \bar{b}_3 b_2 + b_3 \bar{b}_2$

b_3b_2 \ b_1b_0	00	01	11	10
00	0	0	1	1
01	1	1	0	0
11	1	1	0	0
10	0	0	1	1

$g_1 = b_2 \bar{b}_1 + \bar{b}_2 b_1$

b_3b_2 \ b_1b_0	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

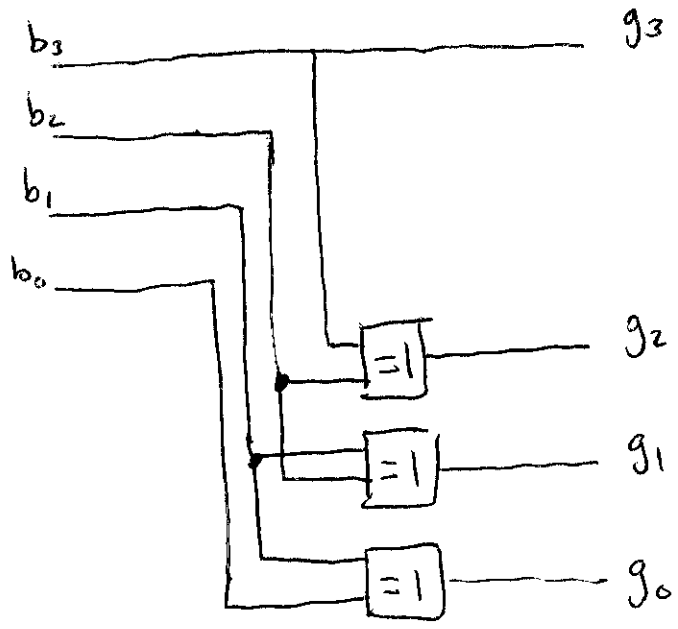
$g_0 = \bar{b}_1 b_0 + b_1 \bar{b}_0$

$g_3 = b_3$

$g_2 = b_3 \oplus b_2$

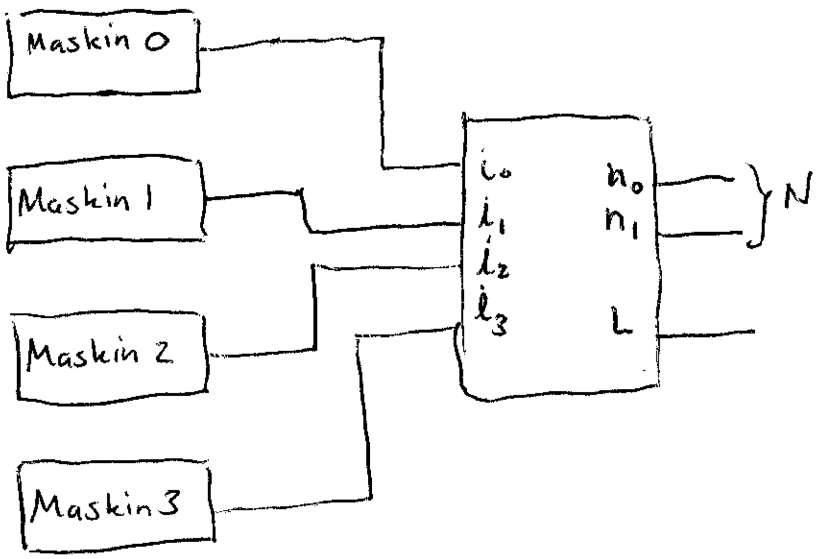
$g_1 = b_2 \oplus b_1$

$g_0 = b_1 \oplus b_0$



Ö5.3

Konstruera en krets som tar emot larm från flea maskiner. En maskin larmar genom att låta en digital signal vara 1. Kretsen ska kunna ta emot larm från fyra maskiner. Om fler än en maskin larmar samtidigt så ska kretsen prioritera. Den som är kopplad till ingång 0 ska ha högst prioritet och den som är kopplat till ingång 3 har lägst prioritet. Kretsen indikerar på utgången $N=(n_1, n_0)$ vilken maskin som larmat (0-3) och på utgången L (1 bit) om det är ett larm eller ej.



Ö5.3 forts.

<u>Lösning</u>:

Ta fram logiskt uttryck för larm signalen (L). :

Om en av ingångarna i är 1 så ska $L = 1$.

alltså: $L = i_0 + i_1 + i_2 + i_3$

Ta fram uttrycket för N. Beskriv funktionen i en sanningstabell.

i_0	i_1	i_2	i_3	n_1	n_0
0	0	0	0	–	–
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	1	0
0	1	0	0	0	1
0	1	0	1	0	1
0	1	1	0	0	1
0	1	1	1	0	1
1	0	0	0	0	0
1	0	0	1	0	0
1	0	1	0	0	0
1	0	1	1	0	0
1	1	0	0	0	0
1	1	0	1	0	0
1	1	1	0	0	0
1	1	1	1	0	0

(inget larm)

Minimera m.h.a K-diagram:

$i_0 i_1$ \ $i_2 i_3$	00	01	11	10
00	–	0	1	1
01	0	0	0	0
10	0	0	0	0
11	0	0	0	0

$n_1 = \bar{i}_0 \bar{i}_1 i_2 + \bar{i}_0 \bar{i}_1 i_3$

$i_0 i_1$ \ $i_2 i_3$	00	01	11	10
00	–	1	0	0
01	1	1	1	1
11	0	0	0	0
10	0	0	0	0

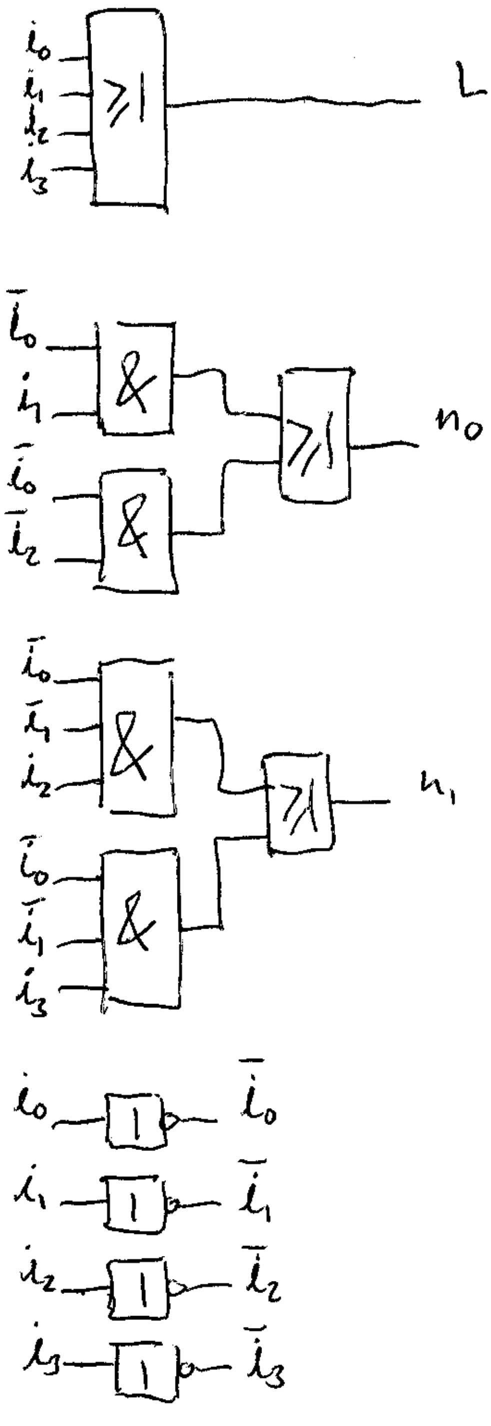
$n_0 = \bar{i}_0 i_1 + \bar{i}_0 \bar{i}_2$

Ö5.3 forts.

<u>Lösning:</u>

$$n_1 = \bar{i}_0 \bar{i}_1 i_2 + \bar{i}_0 \bar{i}_1 i_3$$
$$n_0 = \bar{i}_0 i_1 + \bar{i}_0 \bar{i}_2$$
$$L = i_0 + i_1 + i_2 + i_3$$

Schema:



Ö5.4

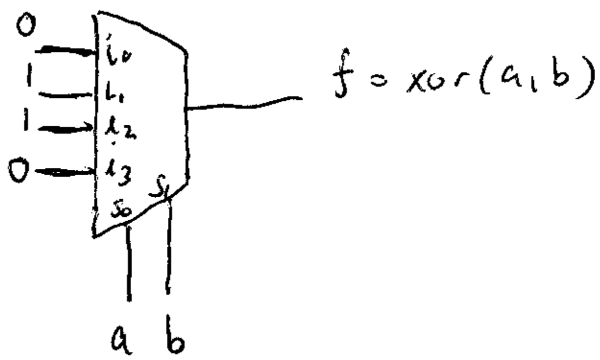
En multiplexer kan användas för att göra en logisk funktion.
Använd en 4-1 Multiplexer för att göra XOR-funktionen

Lösning:

Sanningstabell för XOR

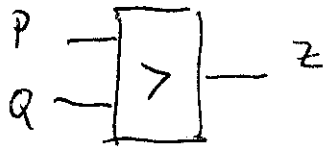
a	b	Z
0	0	0
0	1	1
1	0	1
1	1	0

Använd a och b som styrsignaler till multiplexern
Låt konstanta värden läggas på multiplexerns dataingångar så att XOR-funktionen utförs



ÖS.5

En två-bitars "jämförar"-krets tar in två stycken 2-bitars positiva tal, $P=(P_1, P_0)$ och $Q=(Q_1, Q_0)$. Konstruera en krets på minimal summa-av-produkt form som ger en logisk etta ut på utgången Z om och endast om $P>Q$



Lösning:

Sanningstabell:

P_1	P_0	Q_1	Q_0	Z
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Karnaugh-diagram:

P_1P_0 \ Q_1Q_0	00	01	11	10
00	0	0	0	0
01	1	0	0	0
11	1	1	0	1
10	1	1	0	0

$$Z = P_1\bar{Q}_1 + P_0\bar{Q}_1\bar{Q}_0 + P_0P_1\bar{Q}_0$$

Schema: