Uppgift 1

(a) Börja med att rita upp ett karnaughdiagram för funktionen f.

$x_1x_2 \backslash x_3x_4$	00	01	11	10
00	1	0	0	1
01	0	-	-	0
11	0	1	1	0
10	-	1	-	1

Primimplikatorerna är $x_2'x_4'$, x_1x_2', x_1x_4, x_2x_4. Av dessa är enbart $x_2'x_4'$ väsentlig.

(b) $f(x_1, x_2, x_3, x_4) = x_2'x_4' \vee x_1x_4$.

(c)

$$\begin{aligned} f(x_1, x_2, x_3, x_4) &= x_2'x_4' \vee x_1x_4 = x_2'x_4' \oplus x_1x_4 = (x_2 \oplus 1)(x_4 \oplus 1) \oplus x_1x_4 \\ &= x_2x_4 \oplus x_2 \oplus x_4 \oplus 1 \oplus x_1x_4. \end{aligned}$$

Uppgift 2

(a) För s_1 gäller att $S_1(D) = D + D^2 + D^3 + \mathcal{O}(D^4)$. Alltså är $P_1(D) = S_1(D) \cdot C(D) = (1 + D + D^3 + D^4)(D + D^2 + D^3 + \mathcal{O}(D^4)) = D + \mathcal{O}(D^4) = D$. Allså är D-transformen av s_1 lika med

$$\frac{D}{1 + D + D^3 + D^4}.$$

(b) På motsvarande sätt gäller för s_2 att $S_2(D) = D^3 + \mathcal{O}(D^4)$. Alltså är $P_2(D) = S_2(D) \cdot C(D) = (1 + D + D^3 + D^4)(D^3 + \mathcal{O}(D^4)) = D^3 + \mathcal{O}(D^4) = D^3$. Allså är D-transformen av s_2 lika med

$$\frac{D^3}{1 + D + D^3 + D^4}.$$

Det följer att D-transformen av $s_1 + s_2$ är lika med

$$\frac{D + D^3}{1 + D + D^3 + D^4}.$$

(c) Svaret i (b) kan förkortas enligt

$$\frac{D + D^3}{1 + D + D^3 + D^4} = \frac{D(1 + D^2)}{(1 + D^3)(1 + D)} = \frac{D(1 + D)^2}{(1 + D)^2(1 + D + D^2)} = \frac{D}{1 + D + D^2}.$$

Kopplingspolynomet för den kortaste LFSR som kan generera $s_1 + s_2$ är alltså lika med $1 + D + D^2$.

(d) Utför polynomdivisionen $\frac{1}{1 \oplus D \oplus D^3 \oplus D^4}$.

$$
\begin{array}{r|l}
 & 1 \oplus D \oplus D^2 \\
\hline
1 \oplus D \oplus D^3 \oplus D^4 & 1 \\
 & 1 \oplus D \oplus D^3 \oplus D^4 \\
\hline
 & D \oplus D^3 \oplus D^4 \\
 & D \oplus D^2 \oplus D^4 \oplus D^5 \\
\hline
 & D^2 \oplus D^3 \oplus D^5 \\
 & D^2 \oplus D^3 \oplus D^5 \oplus D^6 \\
\hline
 & \oplus D^6
\end{array}
$$

Alltså är perioden av $C(D)$ lika med 6.

(e) Perioden av en sekvens som genereras av en LFSR med kopplingspolynom $C(D)$ delar perioden av $C(D)$.

Uppgift 3

(a) Det finns två sätt som utsignalen kan bli 1. Dels om minst en av $X(t)$, $X(t-1)$ och $X(t-2)$ är lika med 0, dels om minst två av värdena är lika med 2. 1:or och 3:or kan inte göra att utsignalen blir lika med 0 och kan därför klumpas ihop. Låt våra tillstånd vara baserade på de två föregående värdena $X(t-2)$ och $X(t-1)$ och låt $X(t)$ ses som vår insignal. Beteckna insignalerna 00 som 0, 01 eller 11 som 1 och 10 som 2. För tillstånden låter vi det första värdet beteckna insignalen vid $t-2$ och det andra värdet beteckna insignalen vid $t-1$. Detta ger upphov till tabellen nedan.

Tillstånd	$i=0$	$i=1$	$i=2$
s_{00}	$s_{00}/1$	$s_{01}/1$	$s_{02}/1$
s_{01}	$s_{10}/1$	$s_{11}/1$	$s_{12}/1$
s_{02}	$s_{20}/1$	$s_{21}/1$	$s_{22}/1$
s_{10}	$s_{00}/1$	$s_{01}/1$	$s_{02}/1$
s_{11}	$s_{10}/1$	$s_{11}/0$	$s_{12}/0$
s_{12}	$s_{20}/1$	$s_{21}/0$	$s_{22}/1$
s_{20}	$s_{00}/1$	$s_{01}/1$	$s_{02}/1$
s_{21}	$s_{10}/1$	$s_{11}/0$	$s_{12}/1$
s_{22}	$s_{20}/1$	$s_{21}/1$	$s_{22}/1$

(b) Vi applicerar RF-algoritmen på problemet.

$$P_1 : \{s_{00}, s_{01}, s_{02}, s_{10}, s_{20}, s_{22}\}, \{s_{11}\}, \{s_{12}, s_{21}\}$$
$$P_2 : \{s_{00}, s_{20}, s_{10}\}, \{s_{01}\}, \{s_{02}, s_{22}\}, \{s_{11}\}, \{s_{12}\}, \{s_{21}\}$$
$$P_3 : \{s_{00}, s_{20}, s_{10}\}, \{s_{01}\}, \{s_{02}, s_{22}\}, \{s_{11}\}, \{s_{12}\}, \{s_{21}\}$$

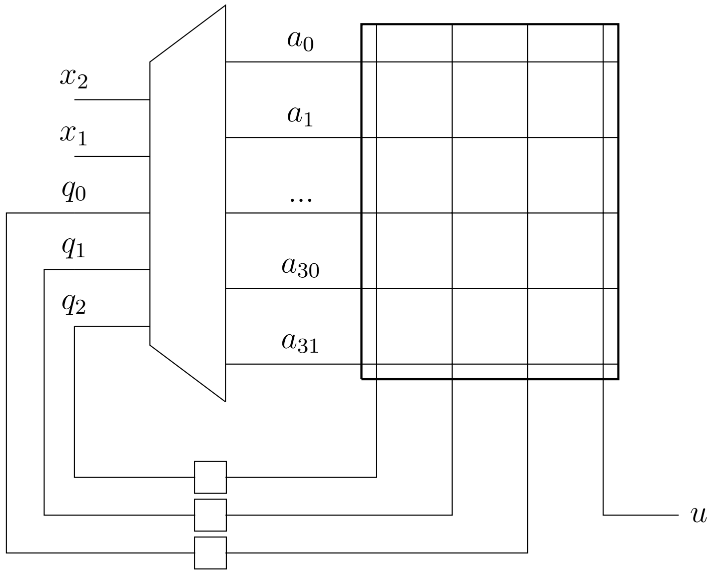
På tabellform får vi nu följande minimerad tillståndsmaskin nedan.

Tillstånd	$i=0$	$i=1$	$i=2$
$s_{00,10,20}$	$s_{00,10,20}/1$	$s_{01}/1$	$s_{02,22}/1$
s_{01}	$s_{00,10,20}/1$	$s_{11}/1$	$s_{12}/1$
$s_{02,22}$	$s_{00,10,20}/1$	$s_{21}/1$	$s_{02,22}/1$
s_{11}	$s_{00,10,20}/1$	$s_{11}/0$	$s_{12}/0$
s_{12}	$s_{00,10,20}/1$	$s_{21}/0$	$s_{22}/1$
s_{21}	$s_{00,10,20}/1$	$s_{11}/0$	$s_{12}/1$

(c) Används en naturlig binärkodning så fås följande tabell. Notera att vi nu byter ut den förkortade formen av insignalen, för att lättare kunna översätta tabellen till tabellen för ROM:et.

$q_0q_1q_2$	$x_1x_2 = 00$	$x_1x_2 = 01$	$x_1x_2 = 10$	$x_1x_2 = 11$
000	000/1	001/1	010/1	001/1
001	000/1	011/1	100/1	011/1
010	000/1	101/1	010/1	101/1
011	000/1	011/0	100/0	011/0
100	000/1	101/0	010/1	101/0
101	000/1	011/0	100/1	011/0

Sekvensnätet för realiseringen ser ut enligt följande.



Tabellen för ROM:et ser ut enligt följande.

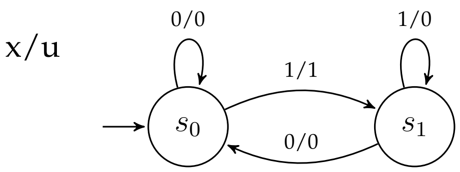
Rad	Ord	Rad	Ord	Rad	Ord	Rad	Ord
00000	0001	01000	0001	10000	0001	11000	$----$
00001	0011	01001	1011	10001	1010	11001	$----$
00010	0101	01010	0101	10010	0101	11010	$----$
00011	0011	01011	1011	10011	1010	11011	$----$
00100	0001	01100	0001	10100	0001	11100	$----$
00101	0111	01101	0110	10101	0110	11101	$----$
00110	1001	01110	1000	10110	1001	11110	$----$
00111	0111	01111	0110	10111	0110	11111	$----$

Uppgift 4

Vi behöver kunna räkna till 14, och till vårt förfogande har vi en mod-16-räknare. Efter att vi har räknat 14 lådor vill vi börja om från 0 igen. Ett sätt att åstadkomma detta är att, precis om i Lab3, konstruera en mod-14-räknare. Då laddade vi in ett startvärde $16 - 14 = 2 \rightarrow [P_0, P_1, P_2, P_3] = [0, 1, 0, 0]$. Detta leder till följande insignaler till räknaren:

$$\left[\overline{\mathrm{SR}}, \mathrm{CEP}, \mathrm{CET}, \overline{\mathrm{PE}}, P_0, P_1, P_2, P_3\right] = [1, 1, \mathrm{CET}, \mathrm{TC}', 0, 1, 0, 0] \tag{1}$$

Eftersom en låda kan stå framför sensorn x i många klockcykler måste vi skapa en tillståndsmaskin som ser till att vi bara räknar en låda en gång. Insignal till vår maskin är x och utsignalen är kallar vi u. I sin tur kopplas u till porten CET på räknaren.

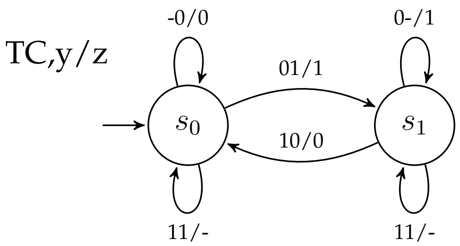


Tillstånd	**Input**, x	
	0	1
s_0	$s_0/0$	$s_1/1$
s_1	$s_0/0$	$s_1/0$

Tillståndskoda tillstånden enligt $s_0 = 0$ och $s_1 = 1$, så får vi realiseringen nedan, där q är tillståndet.

$$\mathrm{u} = q'x \tag{2}$$
$$q^+ = x \tag{3}$$

Nu återstår att konstruera kretsen för att starta/stoppa rullbandet. För det använder vi TC och startknappen y som insignaler.



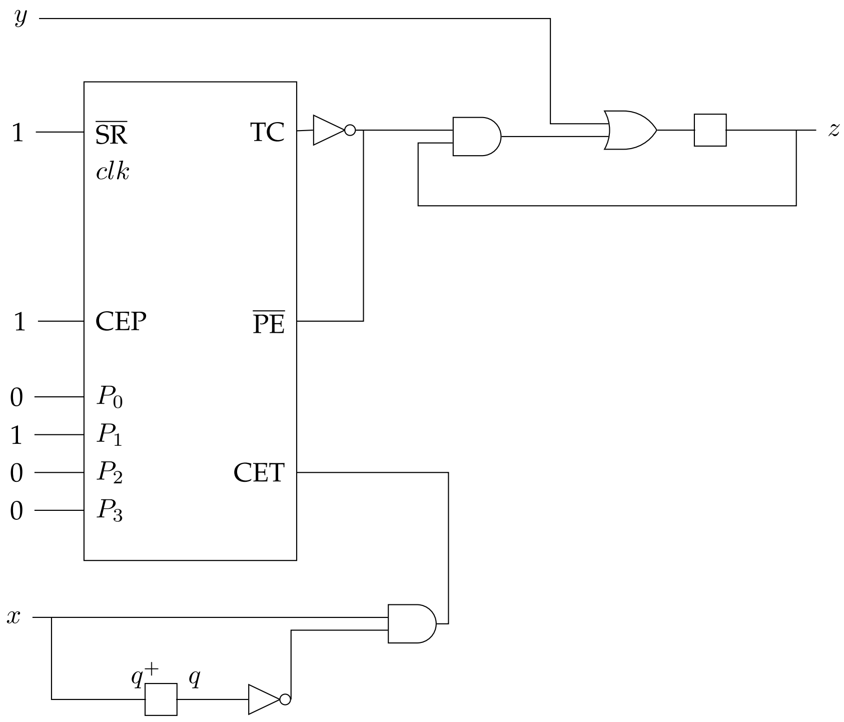
Tillstånd	**Input**, (TC, y)			
	00	01	10	11
s_0	$s_0/0$	$s_1/1$	$s_0/0$	-
s_1	$s_1/1$	$s_1/1$	$s_0/0$	-

Detta leder till Karnaughdiagram med realisering

$$z = y \vee q\mathrm{TC}' \tag{4}$$

z \ y ╲ qTC	00	01	11	10
0	0	0	0	1
1	1	-	-	1

Sätter vi nu ihop alla delar får vi följande krets



Uppgift 5

(a) Faktorisera först talet som $247 = 13 \cdot 19$. Nu fås antalet inverterbara tal i $\mathbb{Z}_{247}$ med hjälp av Eulers Φ-funktion som $\Phi(13 \cdot 19) = 12 \cdot 18 = 216$.

(b) Börja med att köra Euklides algoritm framlänges

$$\begin{aligned} 247 &= 35 \cdot 7 + 2, \\ 7 &= 3 \cdot 2 + 1. \end{aligned}$$

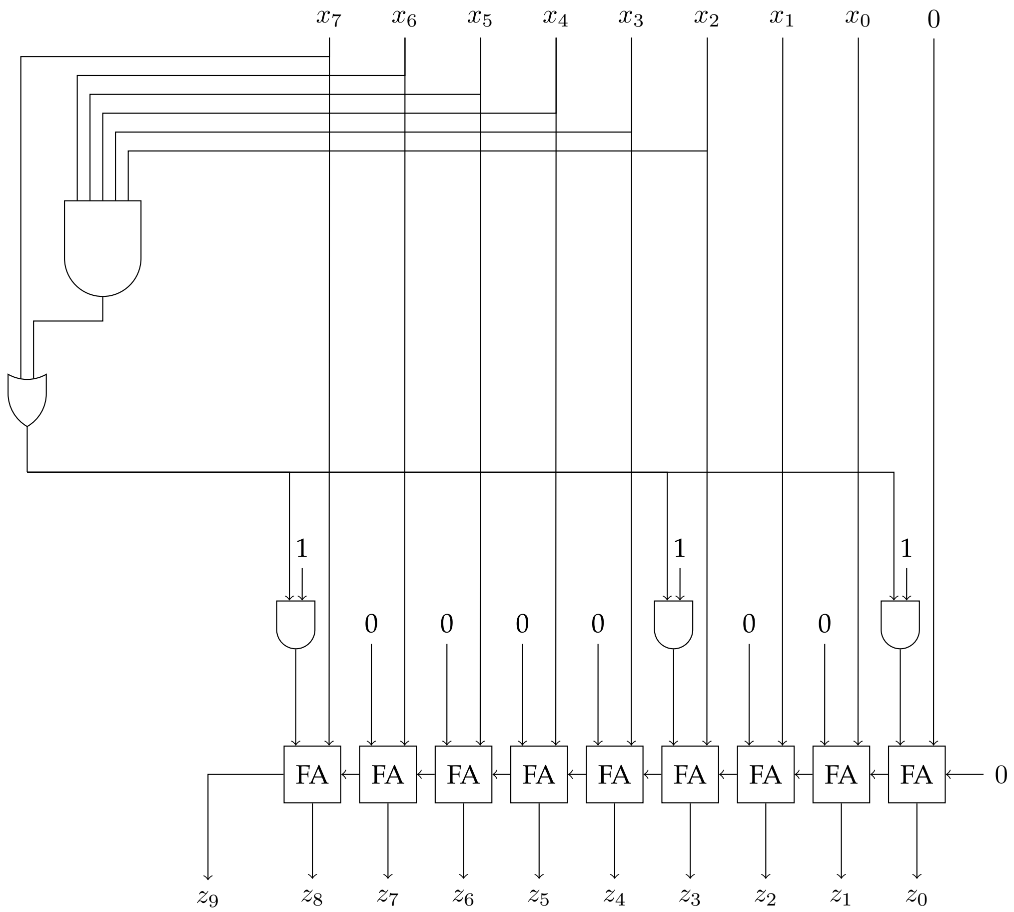
Kör sen Euklides algoritm baklänges och för att få fram att

$$1 = 7 - 3 \cdot 2 = 7 - 3(247 - 35 \cdot 7) = 106 \cdot 7 - 3 \cdot 247.$$

Således gäller att $106 \cdot 7 = 1 + 3 \cdot 247$. Alltså är $x = 7^{-1} = 106$ i $\mathbb{Z}_{247}$.

(c) För att dubblera åttabitarstalet $x = x_7 x_6 \cdots x_0$ skiftar vi alla bitar åt vänster, fyller på med 0:a och bildar niobitarstalet $x' = x_7 x_6 \cdots x_0 0$[1]. Om vårt resultat är större än eller lika med 247 ska vi subtrahera 247 från x'. Detta ska ske då x är lika med $124 = 01111100$ eller större. Detta sker då $x_7 = 1$ eller då $x_6 = x_5 = \cdots = x_2 = 1$. Att subtrahera med 247 motsvarar för niobitarstal att addera med $512 - 247 = 265 = 100001001$. En realisering av denna lösning finns i figuren nedan. De 8 minst signifikanta bitarna $z_7 z_6 \cdots z_0$ är lika med $2x \mod (247)$.

[1] En korrekt, men inte lika snygg, lösning är att addera x med sig själv genom att använda 8 heladderare.



x_7 x_6 x_5 x_4 x_3 x_2 x_1 x_0 0
1
1
1
0 0 0 0 0 0
FA FA FA FA FA FA FA FA FA
0
z_9 z_8 z_7 z_6 z_5 z_4 z_3 z_2 z_1 z_0