

Enkel kretskonstruksjon- Multiplikator

INF2270 – Oblig 1

Petteala@student.matnat.uio.no

Sannhetstabell

Da vi skulle ta løs på denne oppgaven startet vi med å sette opp Sannhetstabell.

				128	64	32	16	8	4	2	1
A3	A2	A1	A0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	1	0	1	0
0	0	1	0	0	0	0	1	0	1	0	0
0	0	1	1	0	0	0	1	1	1	1	0
0	1	0	0	0	0	1	0	1	0	0	0
0	1	0	1	0	0	1	1	0	0	1	0
0	1	1	0	0	0	1	1	1	1	0	0
0	1	1	1	0	1	0	0	0	1	1	0
1	0	0	0	0	1	0	1	0	0	0	0
1	0	0	1	0	1	0	1	1	0	1	0
1	0	1	0	0	1	1	0	0	1	0	0
1	0	1	1	0	1	1	0	1	1	1	0
1	1	0	0	0	1	1	1	1	0	0	0
1	1	0	1	1	0	0	0	0	0	1	0
1	1	1	0	1	0	0	0	1	1	0	0
1	1	1	1	1	0	0	1	0	1	1	0

Sannhetstabell for ønsket virkemåte, hvor inngangssignalene hver representerer en bit i det som skal ganges med den binære representasjon av tallet 10. utgangssignalene representerer en bit hver av svaret. Siden vi multipliserer to 4bits tall får vi 8bit i svaret.

Karnaugh Diagrammer og Logikk

Vi satte så opp Karnaughdiagrammer for hver av utgangssignalene med hensyn på inngangssignaler slik at det vil bli lett å se det boolske uttrykket. Her valgte vi konsekvent at A3A2 som vertikalt på diagrammet og A1A0 som horisontalt. Her kunne man om man ønsker en mer kompakt løsning bruke XOR for 1erne som står på skrå ovenfor hverandre.

Y1	00.	01.	11.	10.
00.	0	1	1	0
01.	0	1	1	0
11.	0	1	1	0
10.	0	1	1	0

Her fikk vi at $Y1 = A0$,

Y2	00.	01.	11.	10.
00.	0	0	1	1
01.	0	0	1	1
11.	0	0	1	1
10.	0	0	1	1

Som ga $Y2 = A1$

Y3	00.	01.	11.	10.
00.	0	1	1	0
01.	1	0	0	1
11.	1	0	0	1
10.	0	1	1	0

Dette ga at $Y3 = A2A0' + A2'A0$

Y4	00.	01.	11.	10.
00.	0	0	1	1
01.	0	1	0	1
11.	1	0	1	0
10.	1	1	0	0

$Y4 = A3'A2'A1 + A3'A1A0' + A3A1'A0' + A3A2'A1' +$

$A3'A2A1'A0 + A3A2A1A0$

Y5	00.	01.	11.	10.
00.	0	0	0	0
01.	1	1	0	1
11.	1	0	0	0
10.	0	0	1	1

$Y5 = A2A1'A0' + A3'A2A1' + A3A2'A1 + A3'A2A0'$

Y6	00.	01.	11.	10.
00.	0	0	0	0
01.	0	0	1	0
11.	1	0	0	0
10.	1	1	1	1

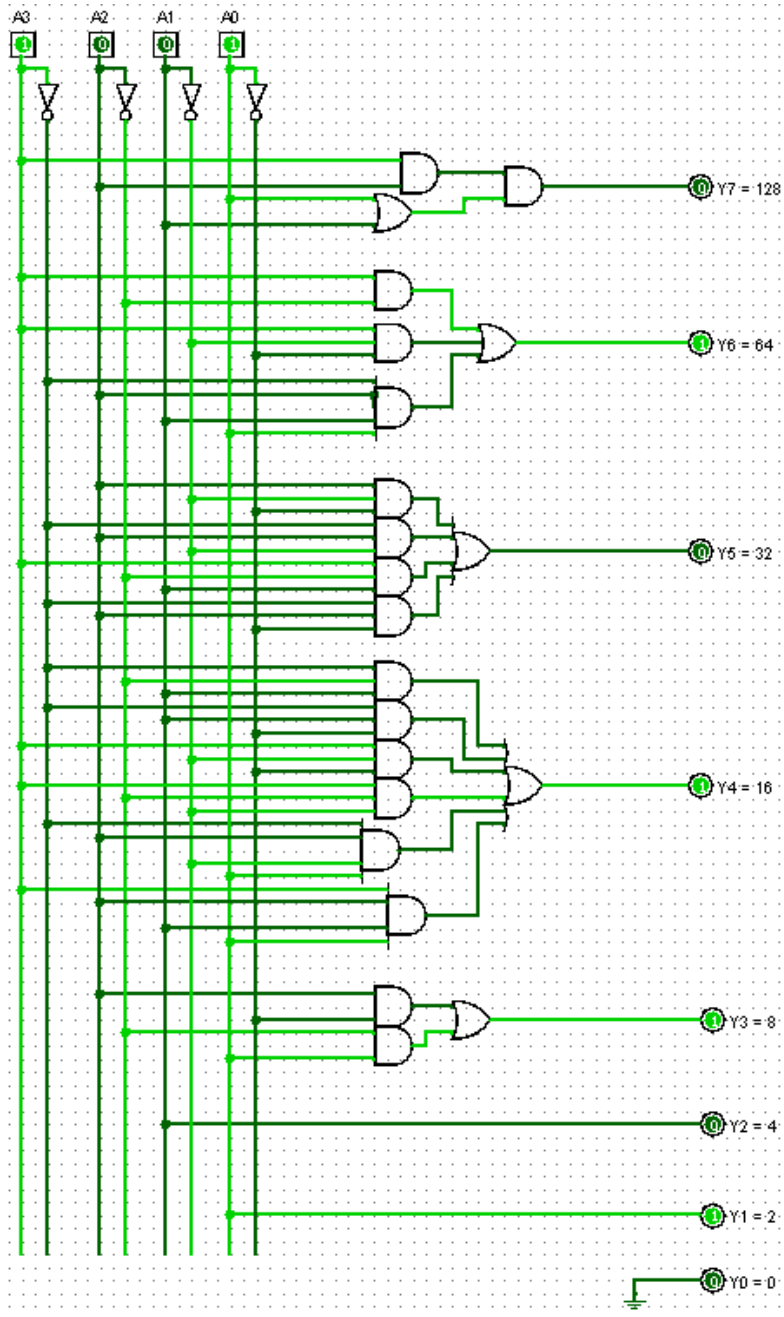
$Y6 = A3A2' + A3A1'A0' + A3'A2A1A0$

Y7	00.	01.	11.	10.
00.	0	0	0	0
01.	0	0	0	0
11.	0	1	1	1
10.	0	0	0	0

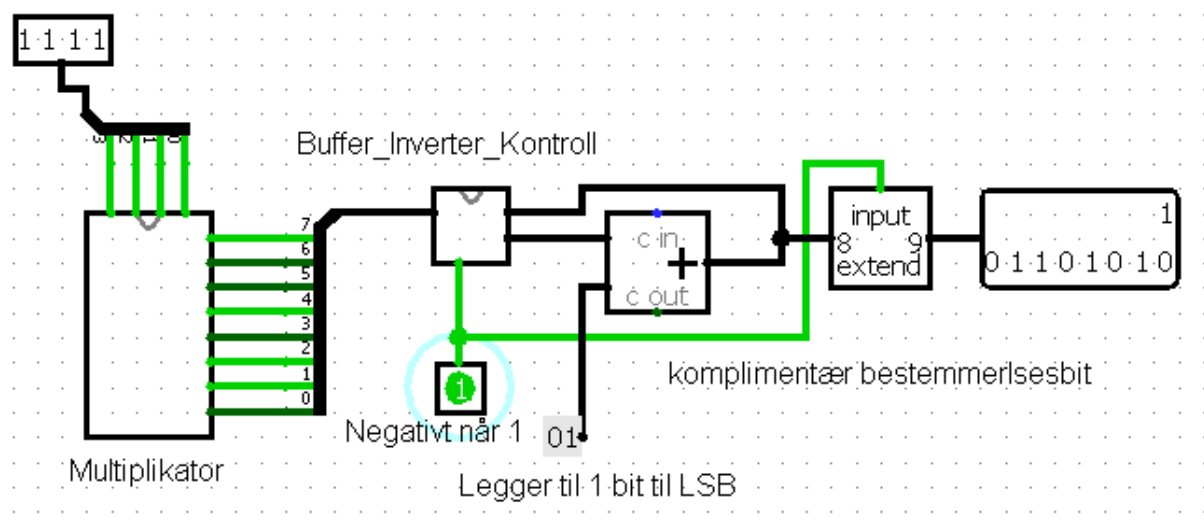
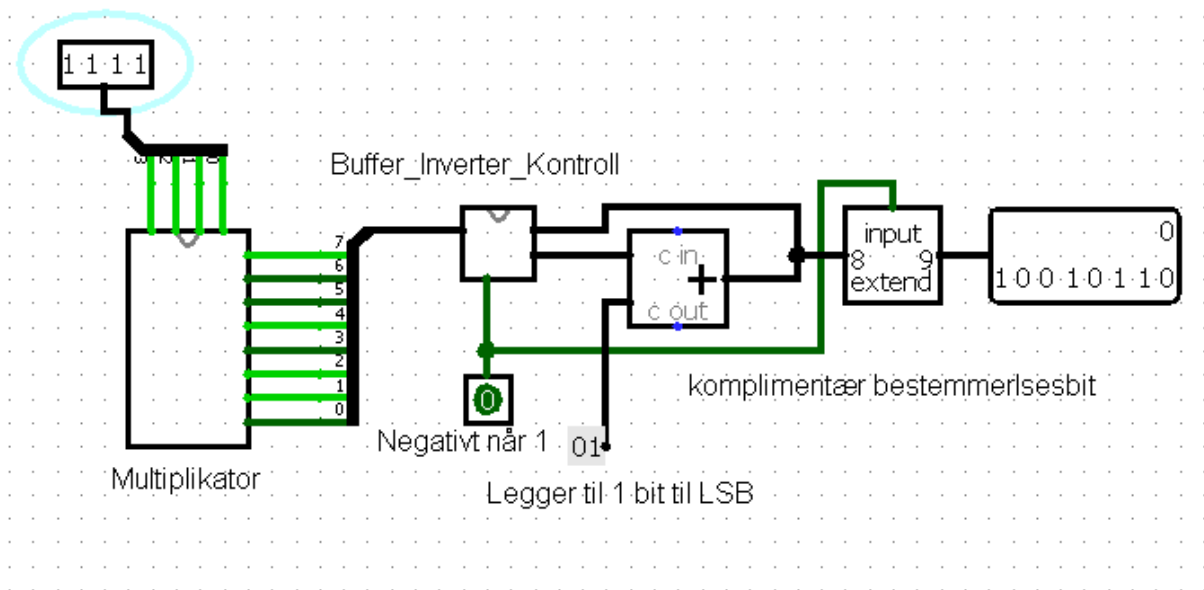
$Y7 = A3A2(A1 + A0)$

Logisk Multiplikator Krets

Vi brukte så de logiske / bolske uttrykkene for å sette opp logikken for hver utgang, med inngang og porter gitt av de bolske uttrykkene. Her valgte vi et «grid» oppsett for å få det oversiktlig. Om man ønsket en mer kompakt design kunne man satt logikken til hver utgang inn i subcircuits. Eller så kunne man brukt signal linjene til hver output om hverandre for å kutte ned på antall porter.



Denne har jeg så satt inn som en subcircuit i en større krets. Etterfulgt av en bufferkontrollkrets med bestående av to buffere, hvor kontrollsignalet på den ene er invertert. Slik at virkemåten blir som en motsatt MUX for utsignalet. Når kontrollsignalet er høyt vill også utsignalet blir invertert og få lagt til 1 på «Least Significant Bit». Så bruker jeg en Bitextender for å legge til en komplementær bestemmelsesbit lik kontrollsignalet, foran «Most Significant Bit».



Demonstrasjon av virkemåte. $15 \times 10 = 150$: $(1111)_2 \times (1010)_2 \rightarrow (10010110)_2$

-150 : $-(10010110)_2 \Rightarrow (101101001)_2 + (000000001)_2 = (101101010)_2$