MUX/DEC Sequential logic design

Vikas Dhiman for ECE275

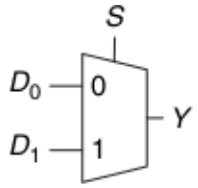
December 1, 2023

1 Objectives

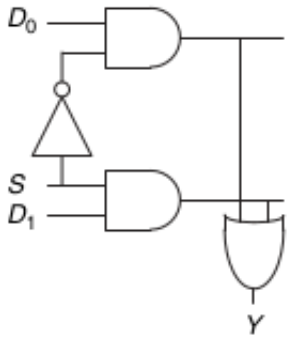
1. Design combinational circuits using multiplexers and decoders

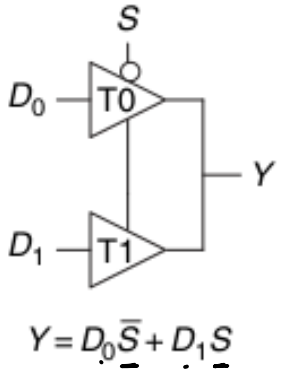
2 Design combinational circuit using multiplexers [1, Section 2.8.1]

2.1 Review: 2to1 Multiplexer (MUX)



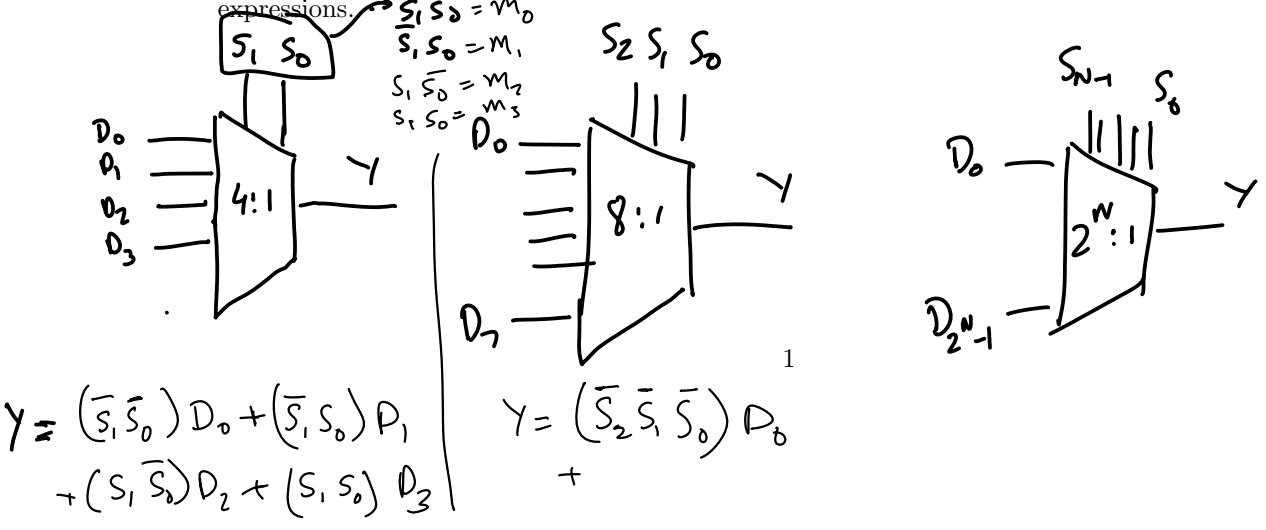
S	D_1	D_0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



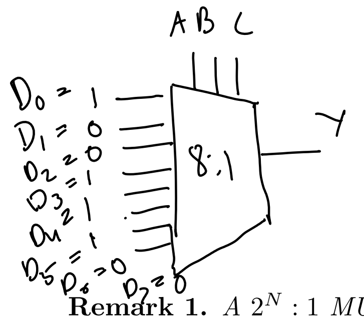


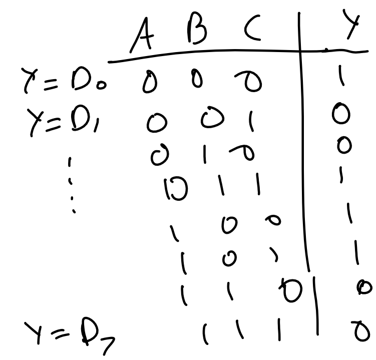
2.2 Wider multiplexers

Draw the symbol for a 4:1 MUX, an 8:1 MUX and a $2^N : 1$ MUX and write corresponding Boolean expressions.



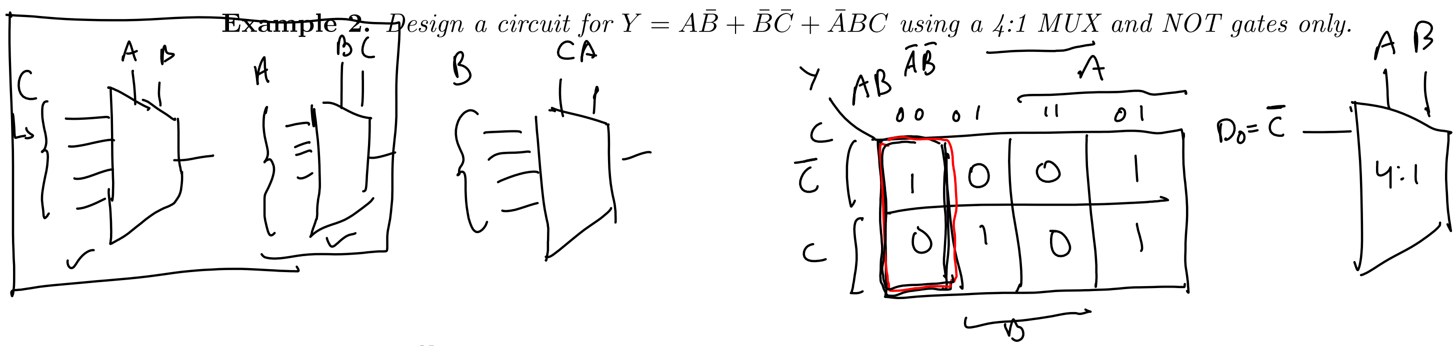
Example 1. *Design a circuit for* $Y = A\bar{B} + \bar{B}\bar{C} + \bar{A}BC$ *using a 8:1 MUX.*





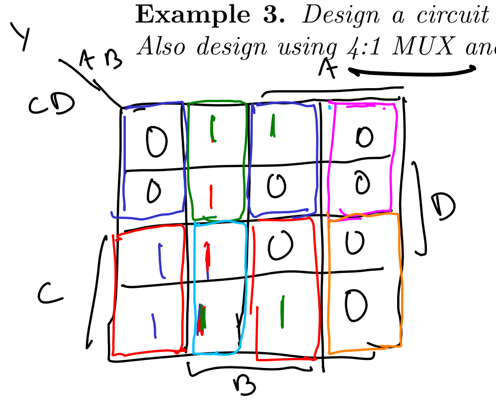
Remark 1. *A* $2^N : 1$ *MUX can be used to program any N-input logic function.*

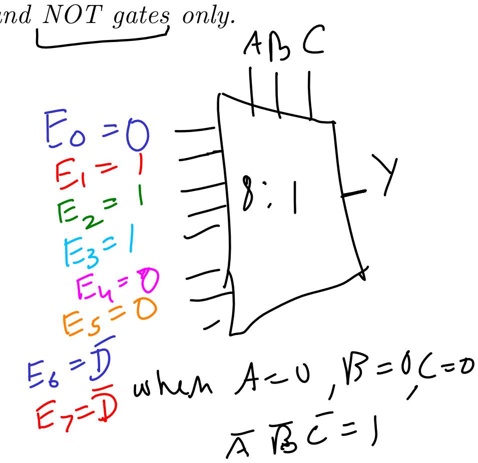
Example 2. *Design a circuit for* $Y = A\bar{B} + \bar{B}\bar{C} + \bar{A}BC$ *using a 4:1 MUX and NOT gates only.*



Remark 2. *A* $2^{N-1} : 1$ *MUX can be used to program any N-input logic function, if we use literals on the input side.*

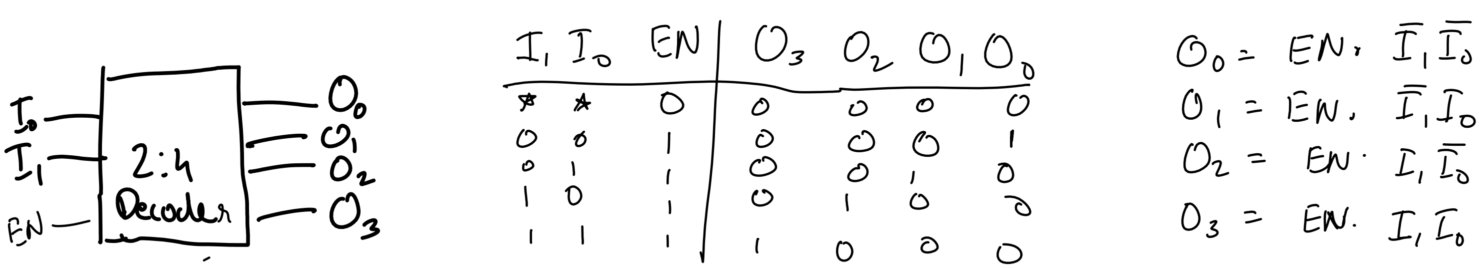
Example 3. *Design a circuit for* $Y = \bar{A}C + \bar{A}B + B\bar{D}$ *using a 8:1 MUX and NOT gates only. Also design using 4:1 MUX and other gates.*



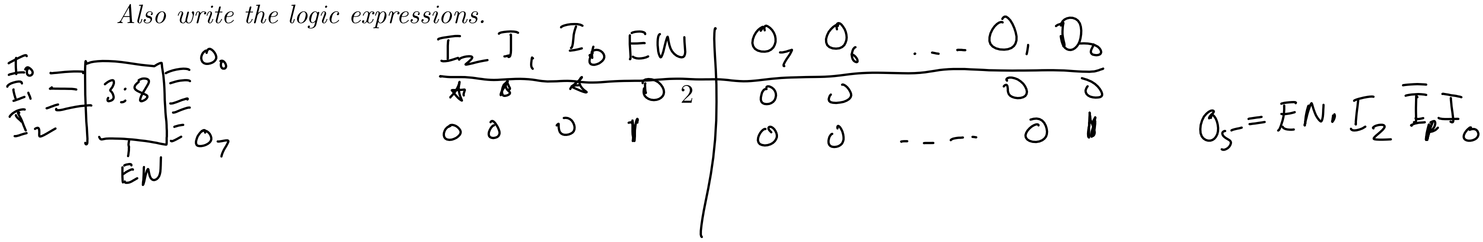


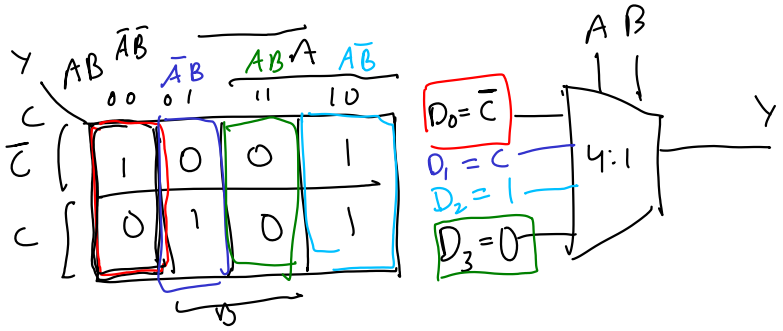
3 Encoders and Decoders

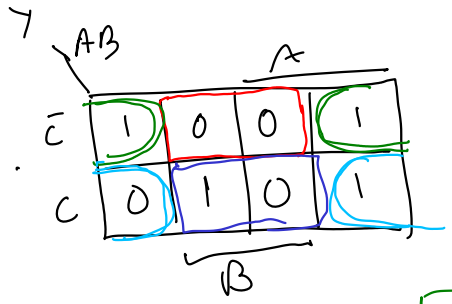
Example 4. *Draw the symbol and the truth table for 2:4 decoder. Also write the logic expressions.*



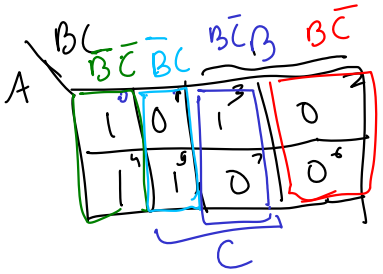
Example 5. *Draw the symbol and the truth table for 3:8 decoder, 4:16 decoder and* $N : 2^N$ *decoder. Also write the logic expressions.*





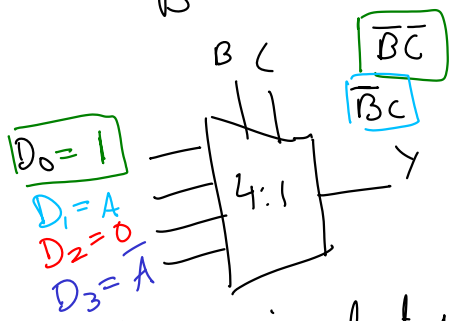


$$Y = \bar{A}\bar{B}\bar{C} + \bar{A}BC + AB\cdot 0 + A\cdot\bar{B}\cdot 1$$



$$Y = A\bar{B} + \bar{B}\bar{C} + \bar{A}BC$$

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



$Y = D_0$ is selected

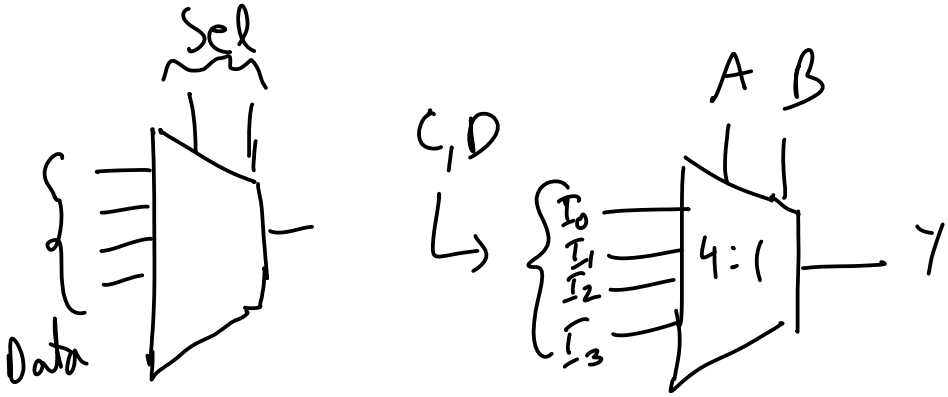
when $B = 0$, $C = 0$

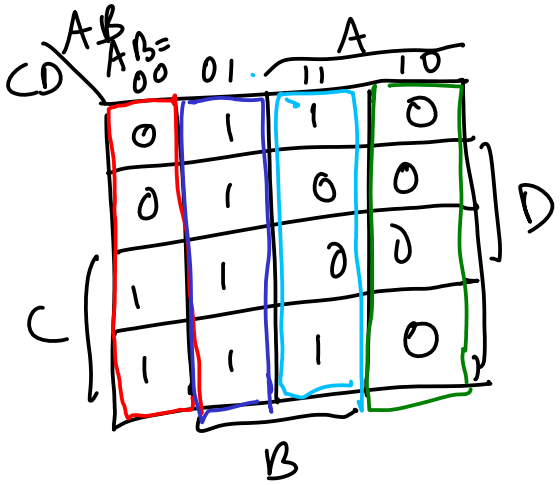
when $\bar{B}\bar{C} = 1$

$B = 0$ $C = 1$

$\bar{B}C = 1$

$$Y = \bar{A}C + \bar{A}B + B\bar{D}$$ using 4:1 MUX + gates





if $AB = 00$ the $Y = I_0$

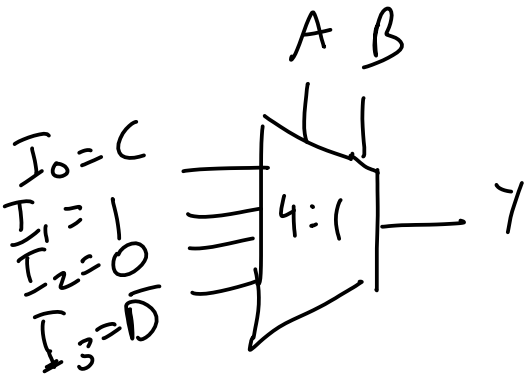
$I_0 = C$

if $AB = 00$ then $Y = I_0 = C$

$I_1 = 1$

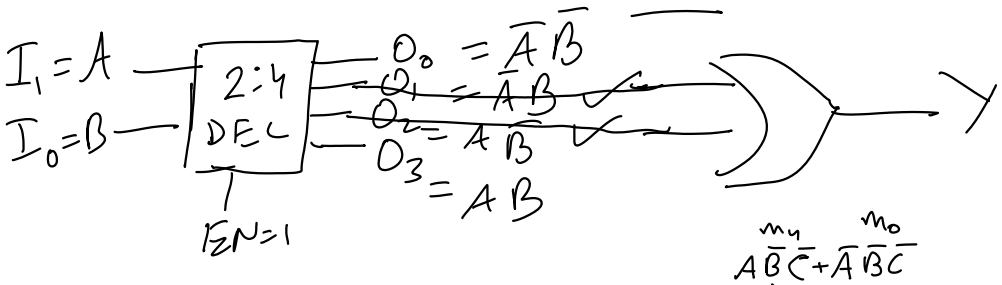
$I_2 = 0$

$I_3 = \bar{D}$


A
B
$I_0 = C$
$I_1 = 1$
$I_2 = 0$
$I_3 = \overline{D}$
4:1
Y

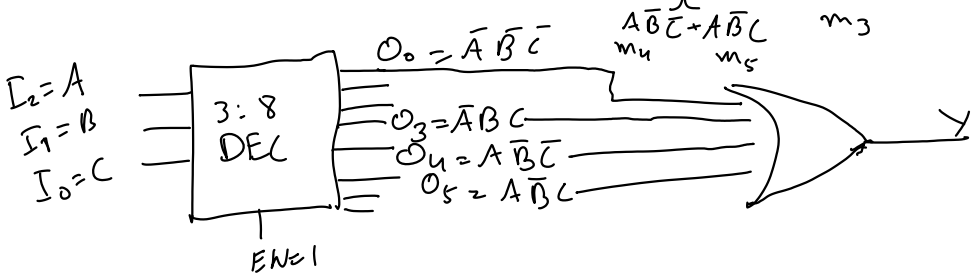
$Y = A\bar{B} + \bar{A}B$ $\qquad$ $Y = A \oplus B$

Example 6. *Design a circuit for a XOR gate using a 2:4 decoder and an OR gate.*



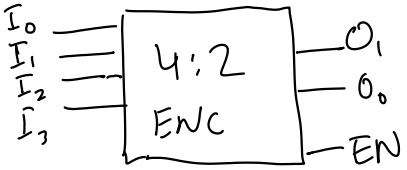
$A\bar{B}\bar{C} + \bar{A}B\bar{C}$ (m_4, m_0) 4-input

Example 7. *Design a circuit for $Y = A\bar{B} + \bar{B}\bar{C} + \bar{A}BC$ using a 3:8 decoder and an OR gate.*



3.1 (Priority) Encoders

Example 8. *Draw symbol and truth table for a 4:2 priority encoder.*



← Most significant bits

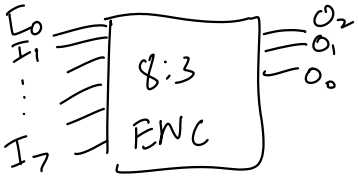
← higher priority

I_3	I_2	I_1	I_0	O_1	O_0	EN
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	*	0	1	1
0	1	*	*	1	0	1
1	*	*	*	1	1	1

2/3[4->[8-15[

$O_1 = I_3 + I_2$

Example 9. *Draw symbol and truth table for a 8:3 priority encoder.*



I_7	...	I_2	I_1	I_0	O_2	O_1	O_0	EN
0		0	0	0	0	0	0	0
0		0	0	1	0	0	1	1
0		0	1	*	0	1	0	1
		1	*	*	0	1	1	1
1		*	*	*	1	0	0	1

8 rows

O_1 $I_3\bar{I}_2$ $I_1\bar{I}_0$

I_3			
0	1	1	1
0	1	1	1
0	1	1	1
0	1	1	1

I_2

References

[1] Sarah L Harris and David Harris. *Digital design and computer architecture.* Morgan Kaufmann, 2022.

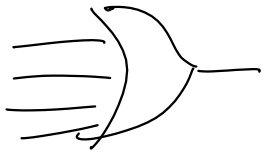
FAN-IN and FAN-OUT

FAN-IN is the maximum number of inputs that you can connect to a gate

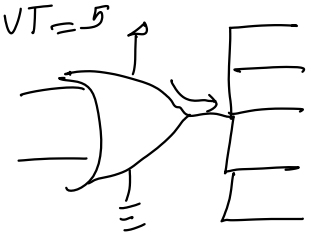
FAN-OUT is the max. number of gates that you can connect

to the output of a gate

FAN-IN = 4



FAN-OUT = 5



V_{IH}