ESc201 : Introduction to Electronics

Digital Circuits

Amit Verma
Dept. of Electrical Engineering
IIT Kanpur

Design Flow (Recap)

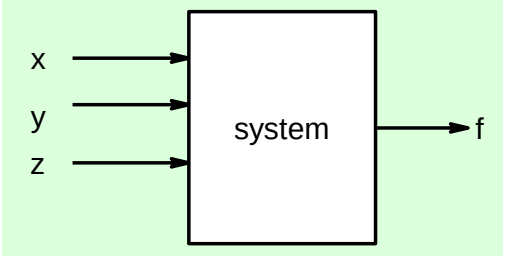
System Description

Truth Table

Boolean Expression

Minimized Boolean Expression

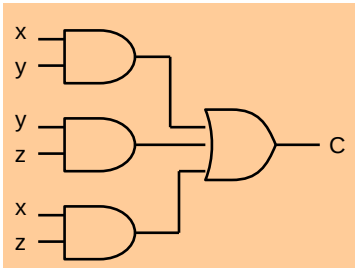
Gate Netlist



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

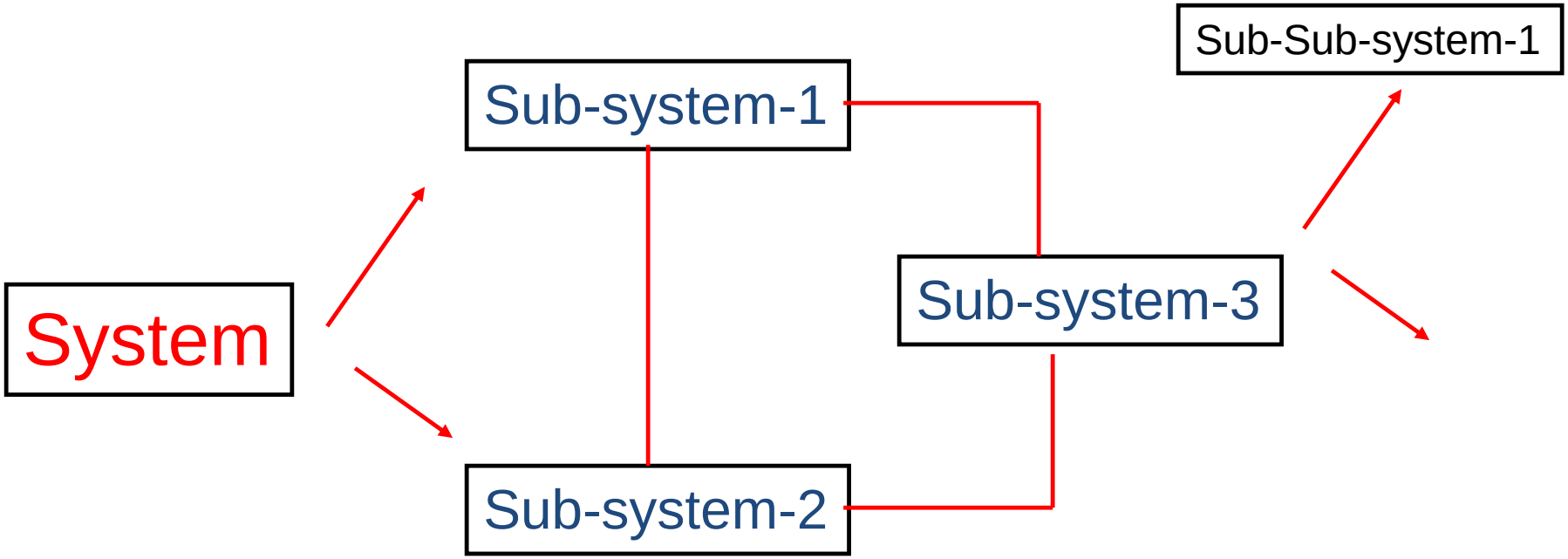
$$f = \bar{x}.\bar{y}.z + \bar{x}.y.z + x.\bar{y}.z + x.y.z$$

$$\Rightarrow f = \bar{x}.\bar{z} + x.z$$



This design approach becomes difficult to use

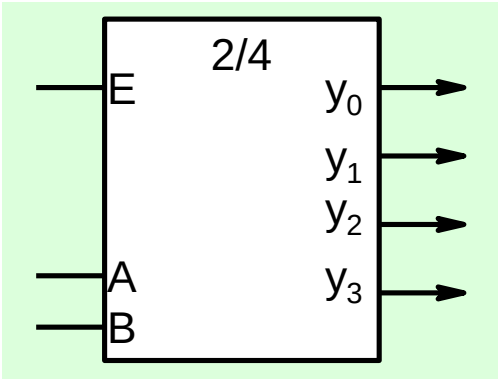
Recap: General Approach



There are certain sub-systems or blocks that are used quite often such as :

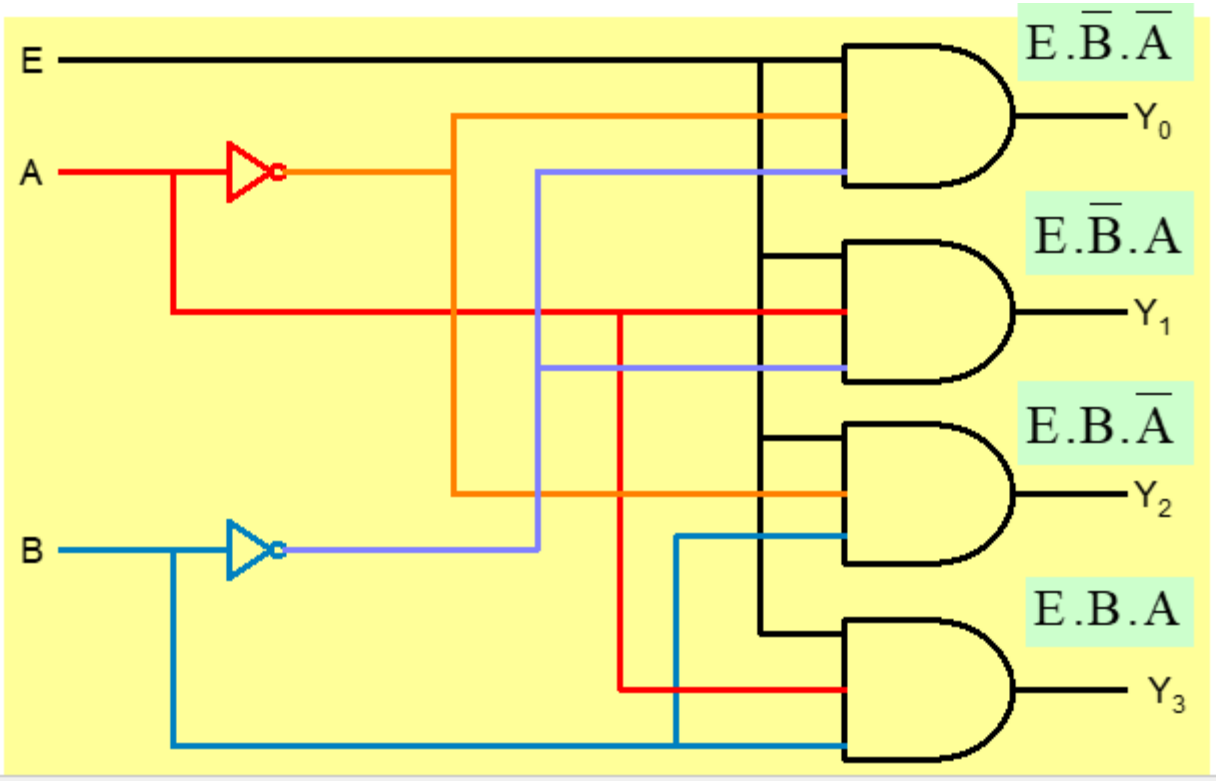
1. **Decoders, Encoders**
2. **Multiplexers**
3. **Adder/Subtractors, Multipliers**
4. **Comparators**
5. **Parity Generators**
6. **……..**

Decoder (recap)



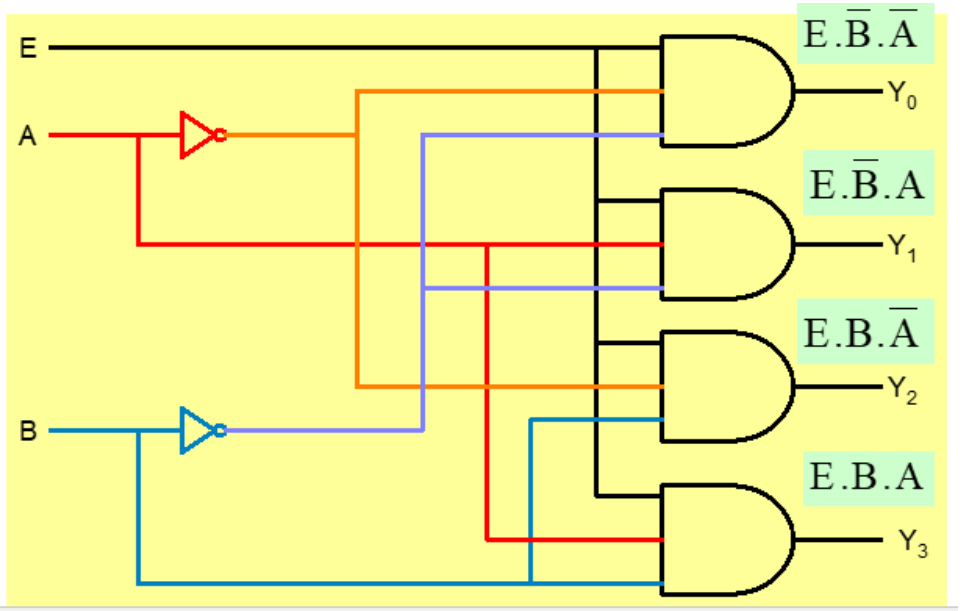
E	B	A	Y_0	Y_1	Y_2	Y_3
0	x	x	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

$$Y_0 = E.\overline{B}.\overline{A}\ ; Y_1 = E.\overline{B}.A ; Y_2 = E.B.\overline{A}\ ; Y_3 = E.B.A$$



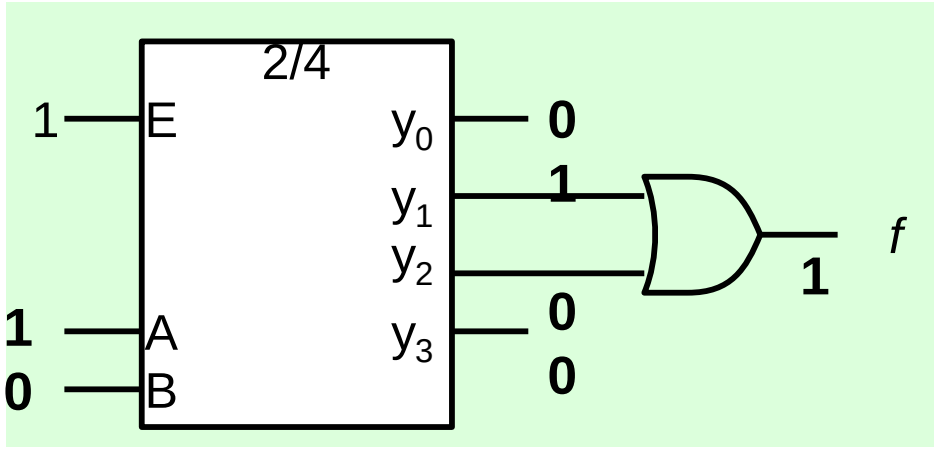
A n to 2^n decoder is a minterm generator (recap)

x	y	min term
0	0	$\bar{x}.\bar{y}$ m0
0	1	x . y m1
1	0	x . y m2
1	1	x . y m3

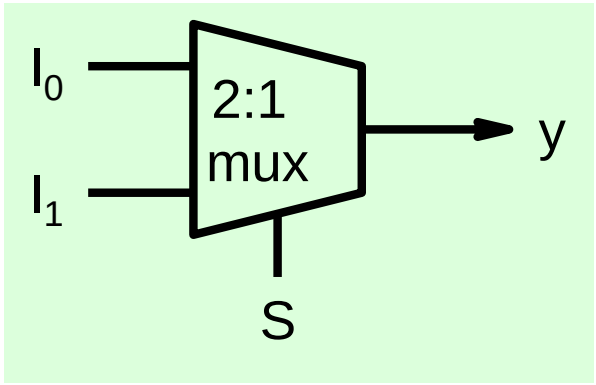


It can be used to implement any combinational circuit

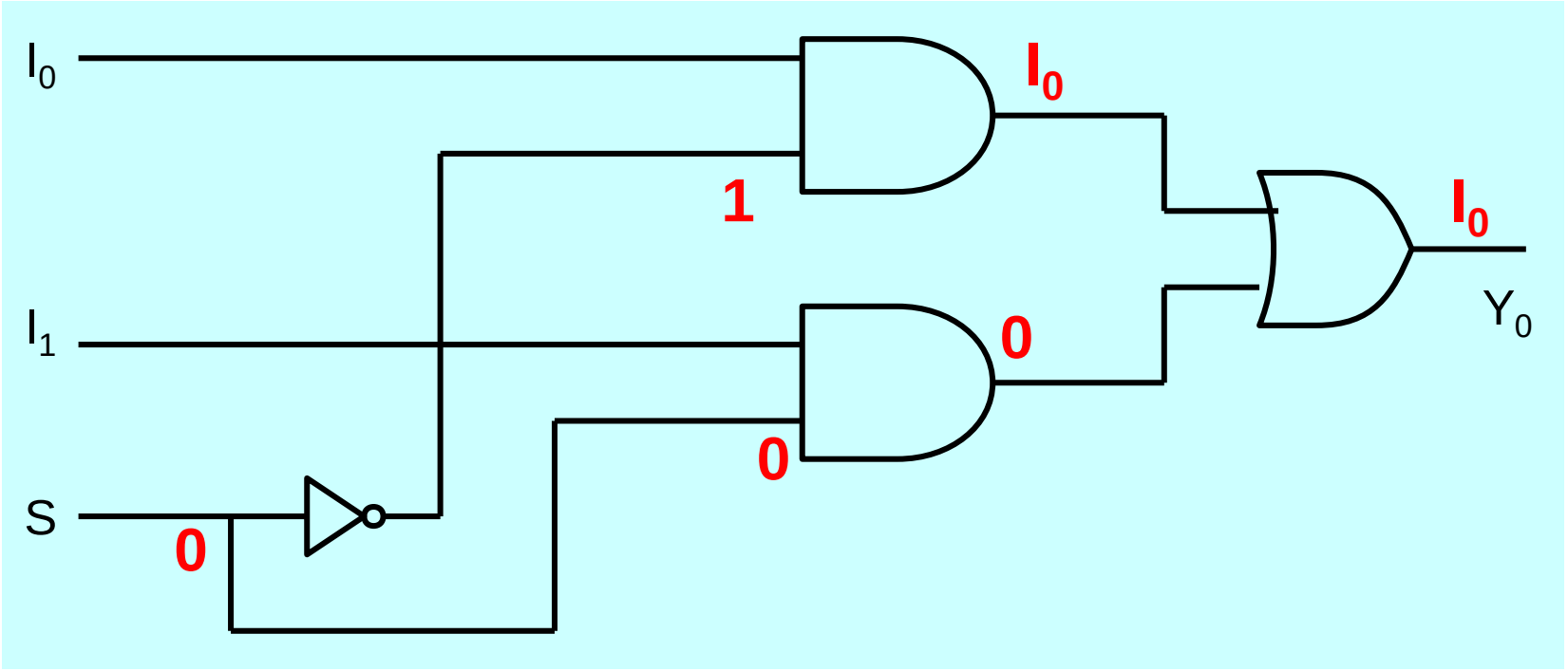
B	A	f_1
0	0	0
0	1	1
1	0	1
1	1	0

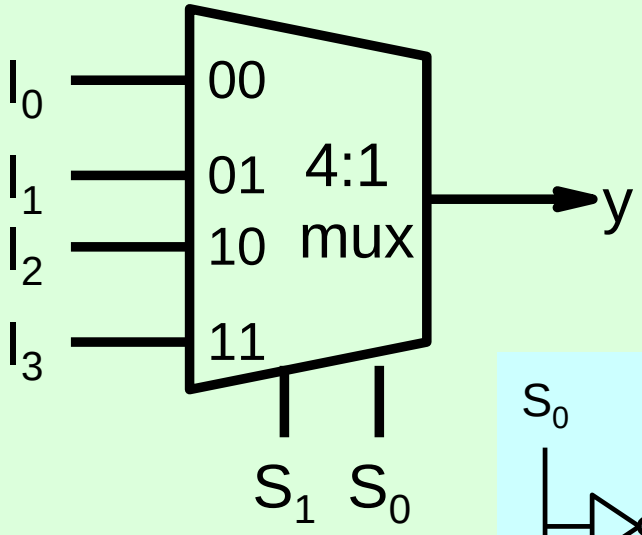


Multiplexers

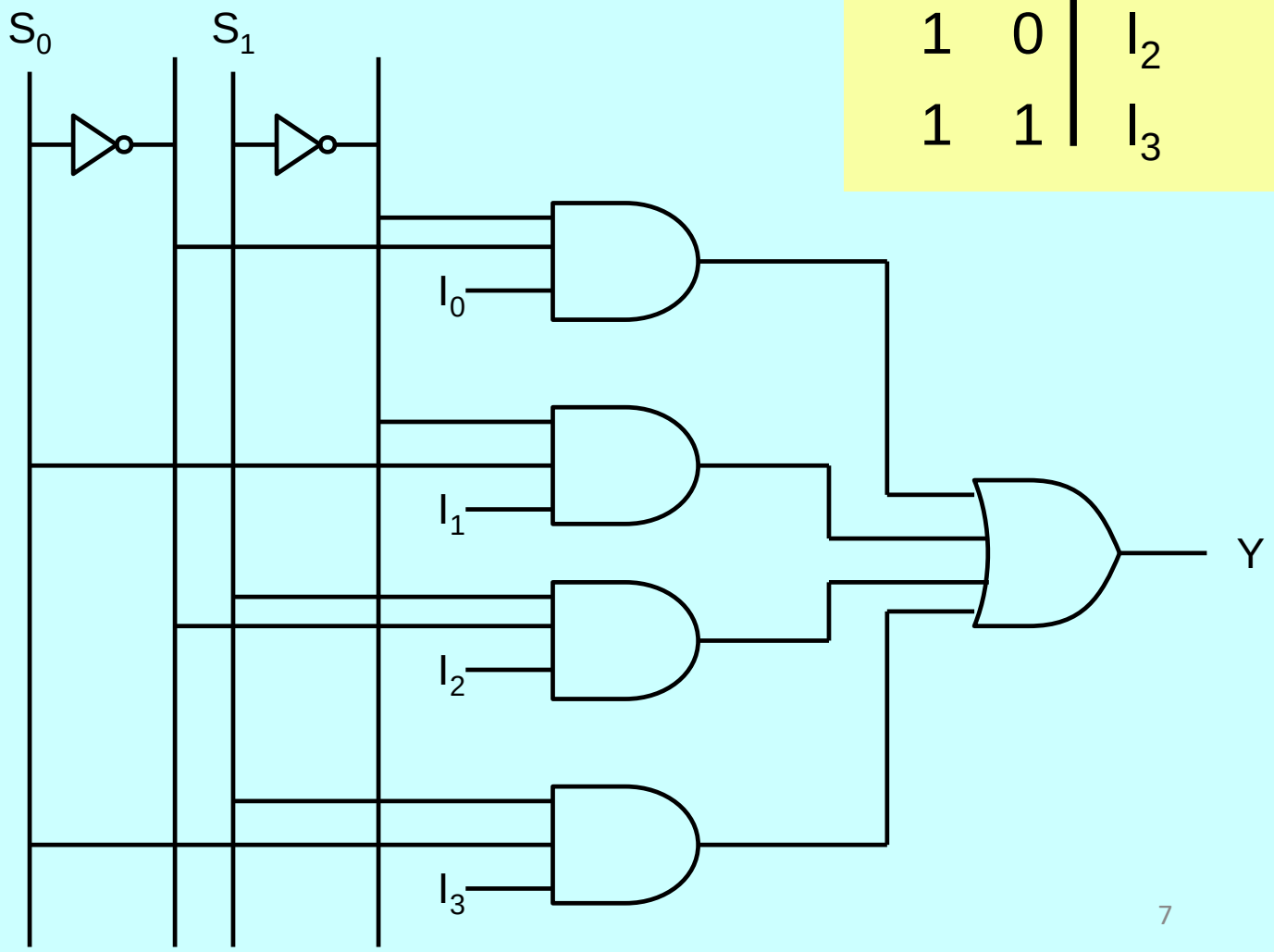


S	*y*
0	I_0
1	I_1



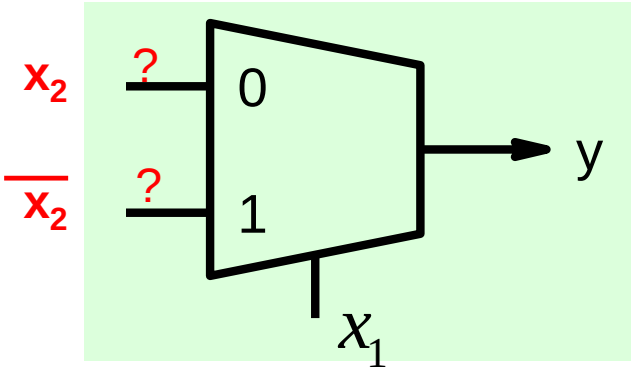


S_1	S_0	y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3



Implementing Boolean expressions using Multiplexers

$$y = x_1\overline{x_2} + \overline{x_1}x_2$$



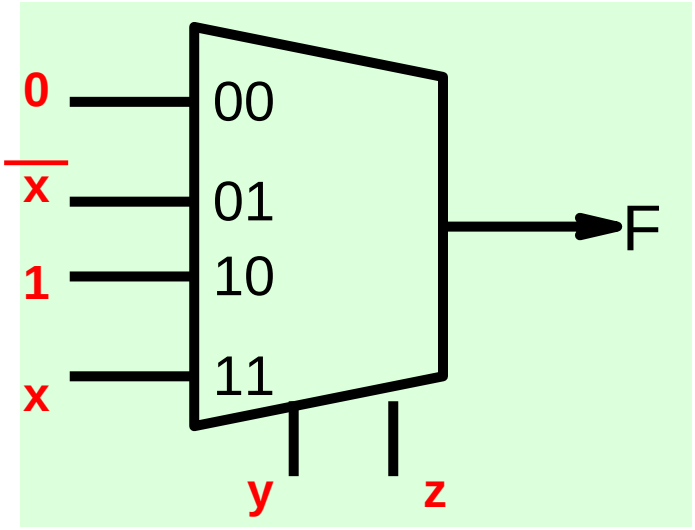
x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

$y = x_2$ when $x_1 = 0$

$y = \overline{x_2}$ when $x_1 = 1$

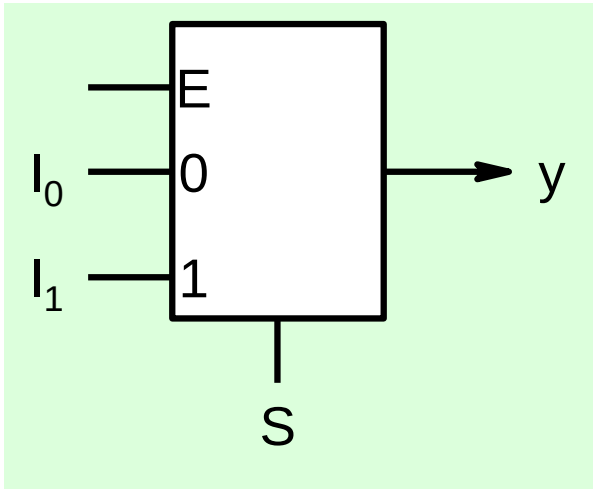
$$F(x, y, z) = \sum(1, 2, 6, 7)$$

A 3 variable function can be implemented with a 4:1 mux with 2 select lines



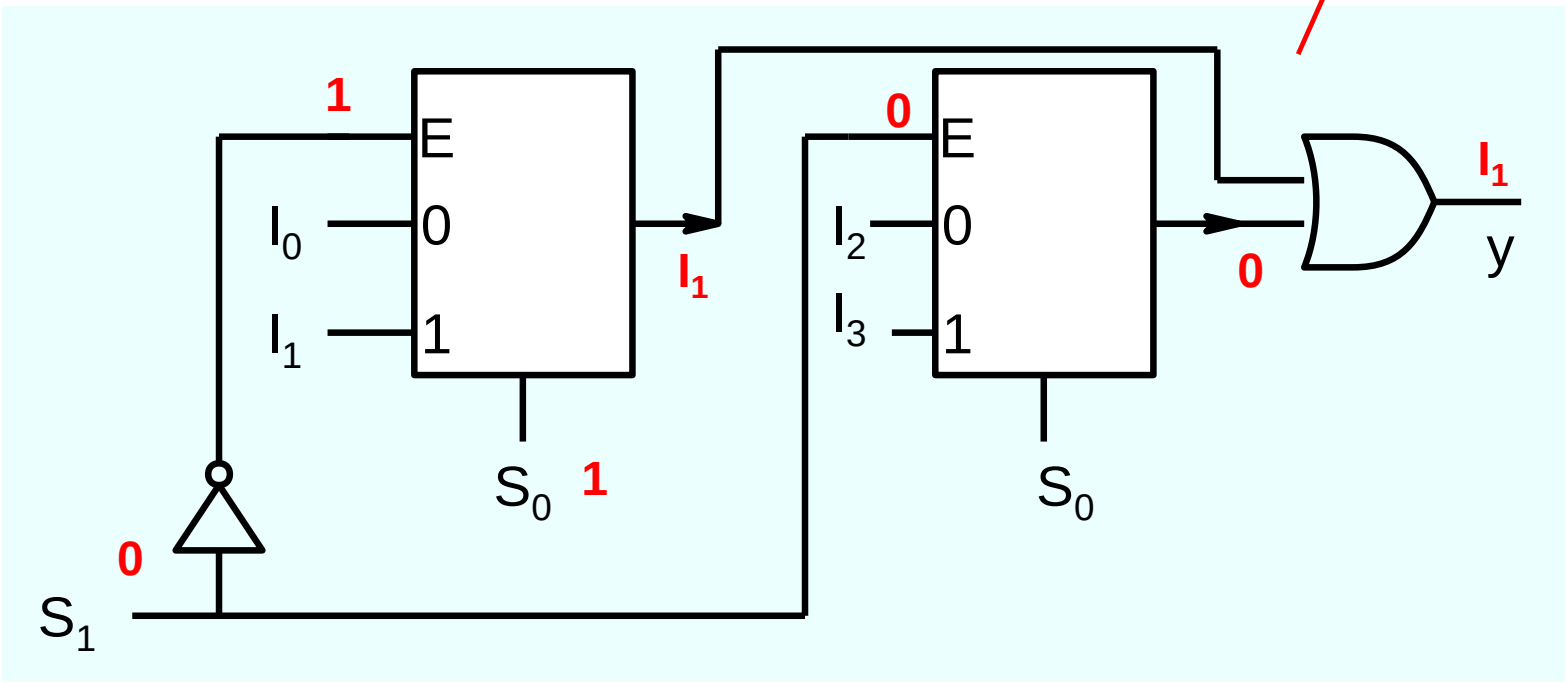
x	y	z	F	
0	0	0	0	F = 0 when yz = 00
1	0	0	0	
0	0	1	1	F = $\overline{x}$ when yz =01
1	0	1	0	
0	1	0	1	F = 1 when yz = 10
1	1	0	1	
0	1	1	0	F = x when yz = 11
1	1	1	1	

Mux. expansion

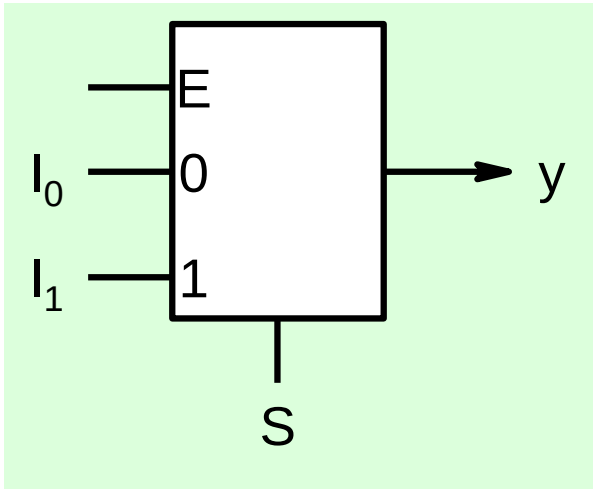


E	S	y
0	x	0
1	0	I_0
1	1	I_1

S_1	S_0	y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

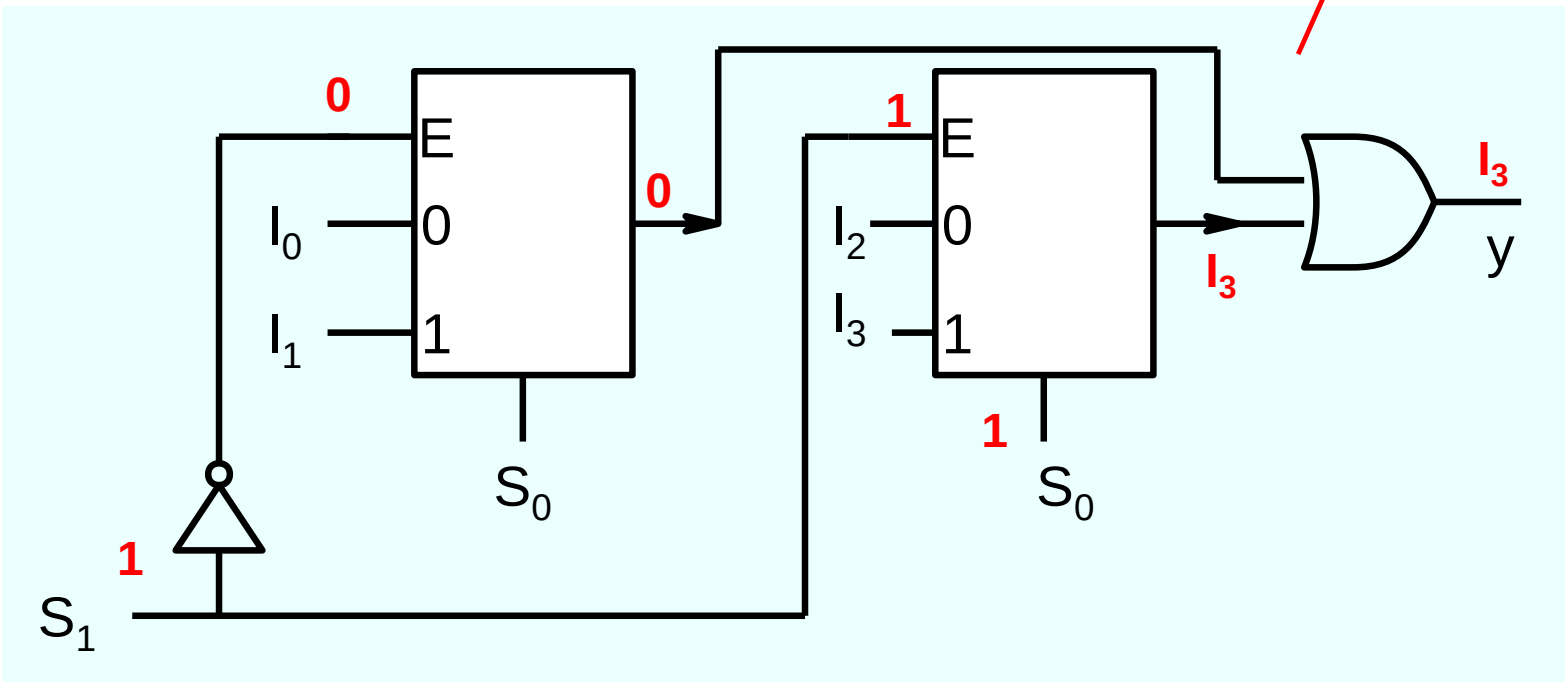


Mux. expansion

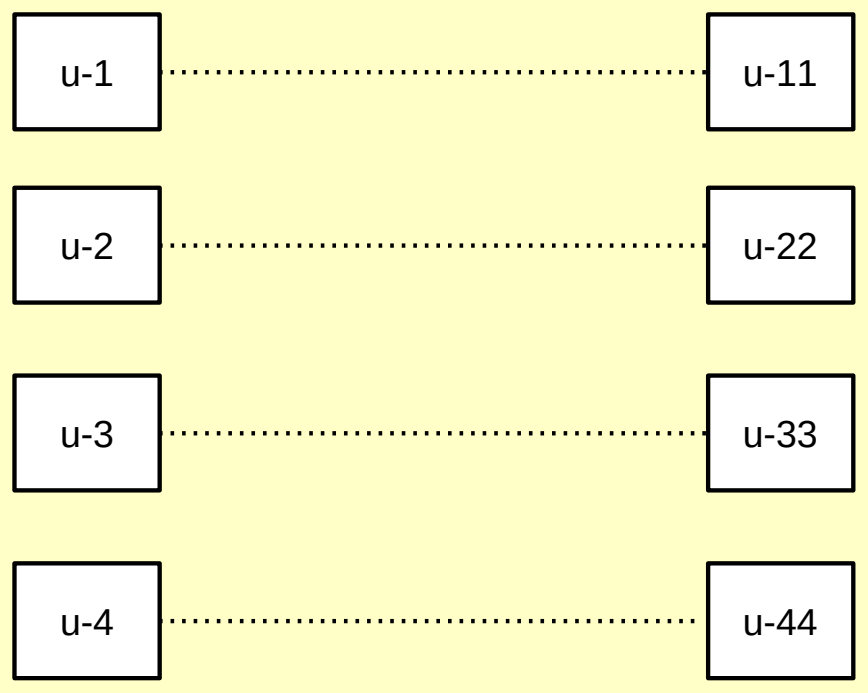


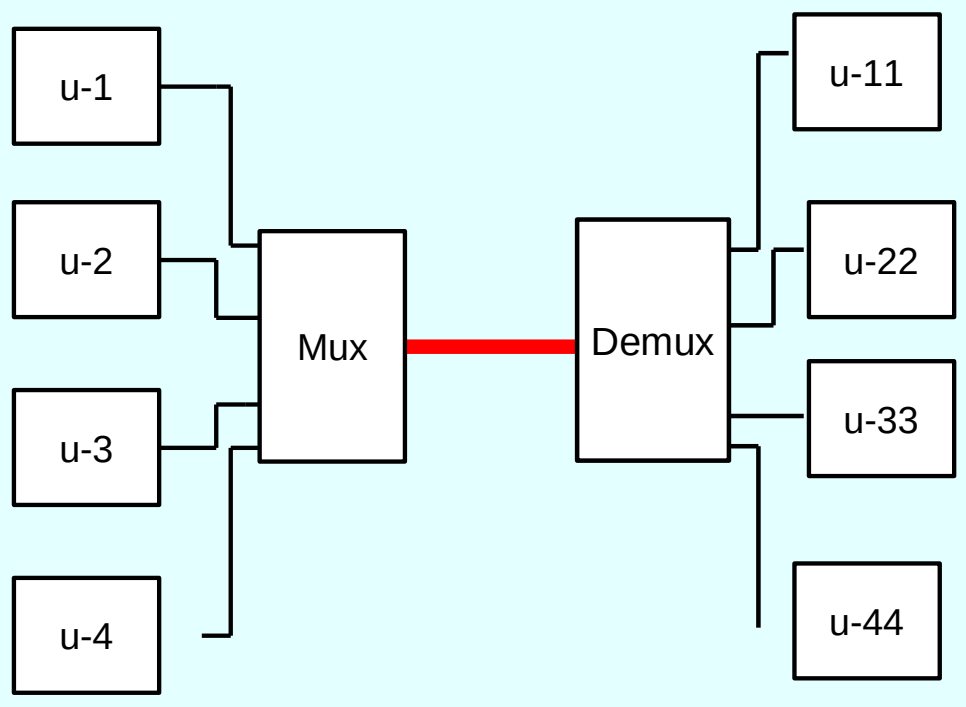
E	S	y
0	x	0
1	0	I_0
1	1	I_1

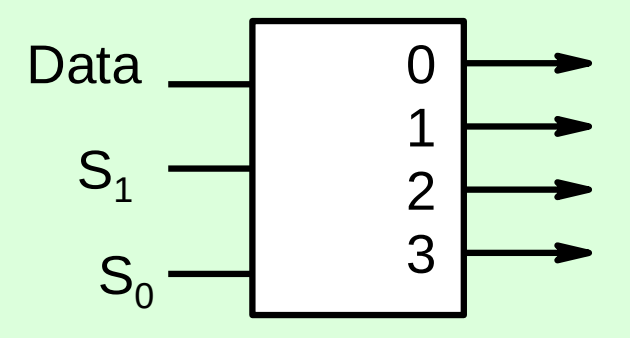
S_1	S_0	y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3



DeMultiplexer

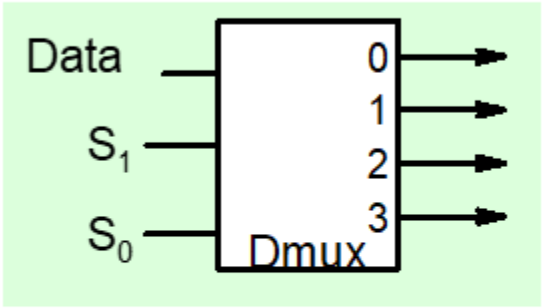


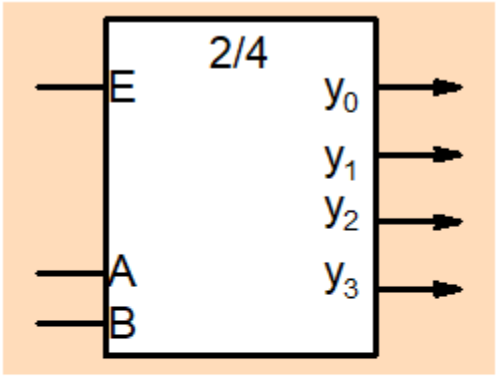




S_1	S_0	y_0	y_1	y_2	y_3
0	0	D	0	0	0
0	1	0	D	0	0
1	0	0	0	D	0
1	1	0	0	0	D

Demultiplexer is very much like a decoder





S_1	S_0	y_0	y_1	y_2	y_3
0	0	D	0	0	0
0	1	0	D	0	0
1	0	0	0	D	0
1	1	0	0	0	D

E	B	A	Y_0	Y_1	Y_2	Y_3
0	x	x	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1