DIGITAL SYSTEMS AND MICROPROCESSORS (ELE2002M)

LECTURE 6 - COMBINATIONAL LOGIC CONT….

Instructor:

Assoc. Prof. Dr. Edmond Nurellari

What we have done so far?

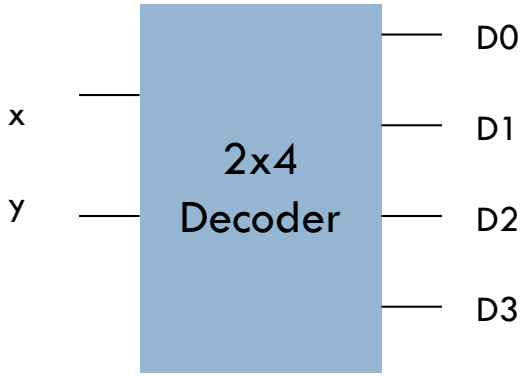
- Half and Full Adders
- Half and Full Subtractor
- Adder-Subtractor Unit
- Binary Multiplier
- Comparator
- Decoders

Today's Class

- Decoders
- Encoders
- Multiplexers
- Other Reduction Techniques

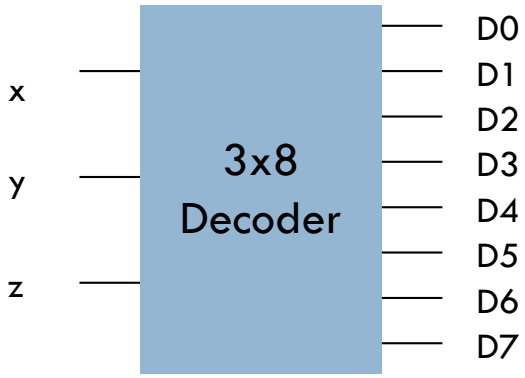
Decoder

- A n-bit binary code is capable of representing 2^n distinct elements of coded information.
- A decoder is a combinational circuit that converts binary information from n input lines to a *maximum* of 2^n outputs.
- A decoder presented here is called n-to-m-line decoder. Where m <= 2^n



Inputs			Outputs			
x	y		D0	D1	D2	D3
0	0		1	0	0	0
0	1		0	1	0	0
1	0		0	0	1	0
1	1		0	0	0	1

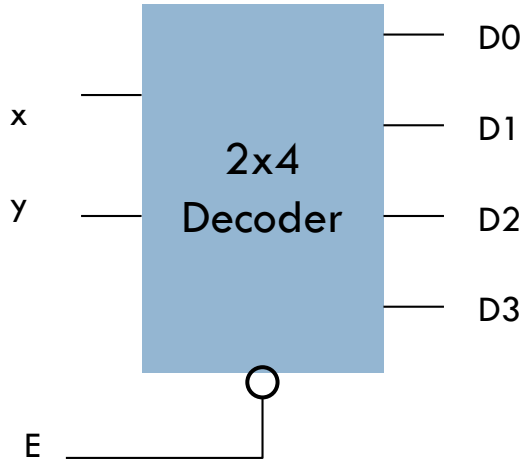
3x8 Decoder



Inputs				Outputs							
x	y	z		D0	D1	D2	D3	D4	D5	D6	D7
0	0	0		1	0	0	0	0	0	0	0
0	0	1		0	1	0	0	0	0	0	0
0	1	0		0	0	1	0	0	0	0	0
0	1	1		0	0	0	1	0	0	0	0
1	0	0		0	0	0	0	1	0	0	0
1	0	1		0	0	0	0	0	1	0	0
1	1	0		0	0	0	0	0	0	1	0
1	1	1		0	0	0	0	0	0	0	1

Complemented 2x4 Line Decoder

E	x	y		D0	D1	D2	D3
1	X	X		1	1	1	1
0	0	0		0	1	1	1
0	0	1		1	0	1	1
0	1	0		1	1	0	1
0	1	1		1	1	1	0



Logic Implementation using Decoders

- Implement a Full adder using a 3x8 Decoder.

- Implement a 4x16 Decoder using two 3x8 Decoders

Encoder

- An encoder is a digital circuit that performs inverse operation of a decoder.
- An encoder has 2^n (or fewer) input lines and n output lines.
- An example of encoder is the **Octal-to-Binary encoder** given below.

Inputs									Outputs		
D0	D1	D2	D3	D4	D5	D6	D7		x	y	z
1	0	0	0	0	0	0	0		0	0	0
0	1	0	0	0	0	0	0		0	0	1
0	0	1	0	0	0	0	0		0	1	0
0	0	0	1	0	0	0	0		0	1	1
0	0	0	0	1	0	0	0		1	0	0
0	0	0	0	0	1	0	0		1	0	1
0	0	0	0	0	0	1	0		1	1	0
0	0	0	0	0	0	0	1		1	1	1

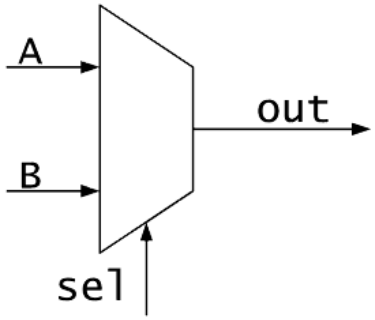
Priority Encoder

- Includes a priority function
- If two or more inputs are '1' at the same time, the input having the highest priority will take precedence.

D0	D1	D2	D3		x	y	V
0	0	0	0		X	X	0
1	0	0	0		0	0	1
X	1	0	0		0	1	1
X	X	1	0		1	0	1
X	X	X	1		1	1	1

Multiplexer

- The Multiplexer
  - Selects one of 2^n inputs and copies it to a single output
  - The selected line is determined from the bit combination (address) on the n selection line



2-to-1 line MUX

2:1 Multiplexer

sel	A	B	out
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

sel	A	B	out
0	0	?	0
0	1	?	1
1	?	0	0
1	?	1	1

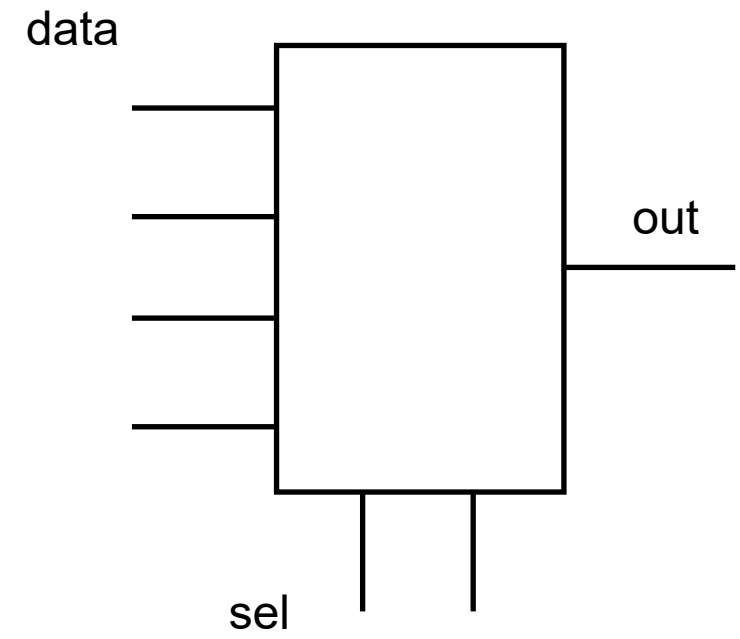
if A is selected, don't care about b.

sel \ AB	00	01	11	10
0			1	1
1		1	1	

K map for 2:1 Multiplexer

sel \ AB	00	01	11	10
0			1	1
1		1	1	

output = $\overline{sel}$.A + sel.B



Principal can be extended to

4:1 – 2 select lines and 4 data lines

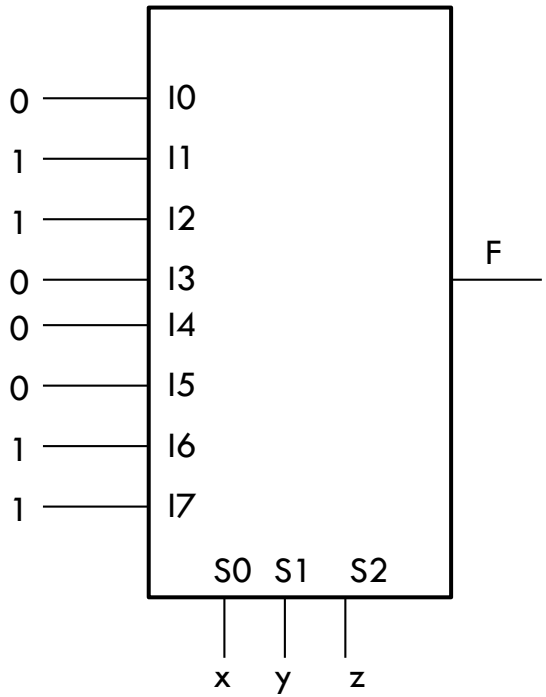
8:1 – 3 select lines and 8 data lines

and so on…

Implementing Logic using MUX

- Implement a Boolean function $F = \sum (1,2,6,7)$ using a MUX

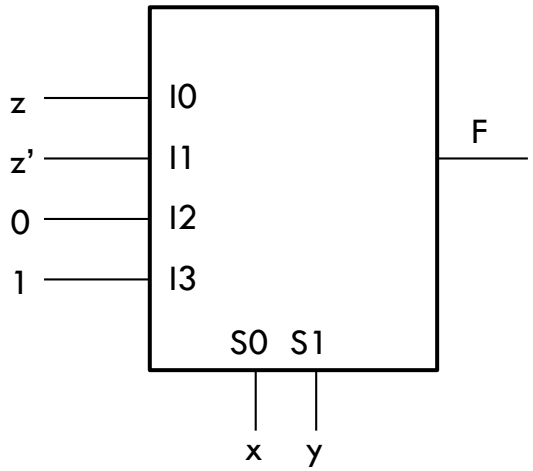
Using 8x1 MUX



More Optimum Solution

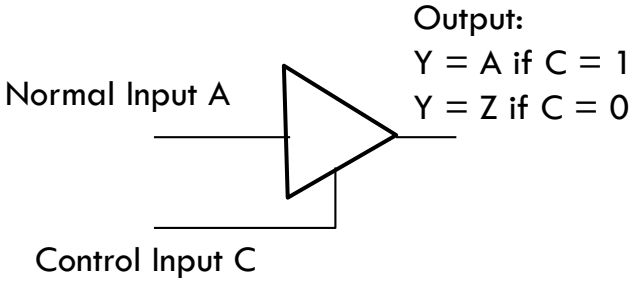
x	y	z	F	
0	0	0	0	F = z
0	0	1	1	
0	1	0	1	F = z'
0	1	1	0	
1	0	0	0	F = 0
1	0	1	0	
1	1	0	1	F = 1
1	1	1	1	

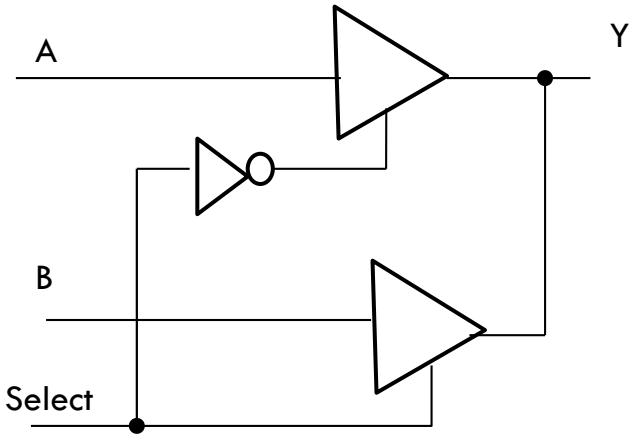
Using 4x1 MUX



MUX using 3 State Gates

- A MUX can be constructed with 3-state gates.
- 3-states are 1,0 and Z (high impedance)
- 3 state gates may perform any conventional loc (such as AND or NAND).
- However, the are most commonly used for buffer gates





2x1 Line MUX

Next Lecture

- Sequential Logic Design