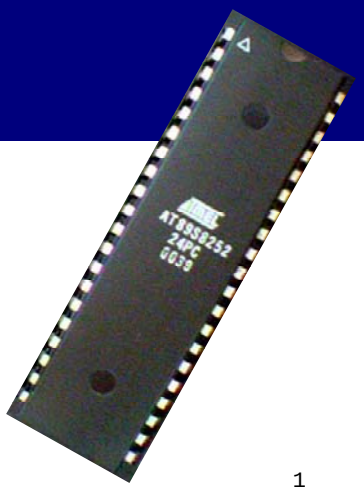
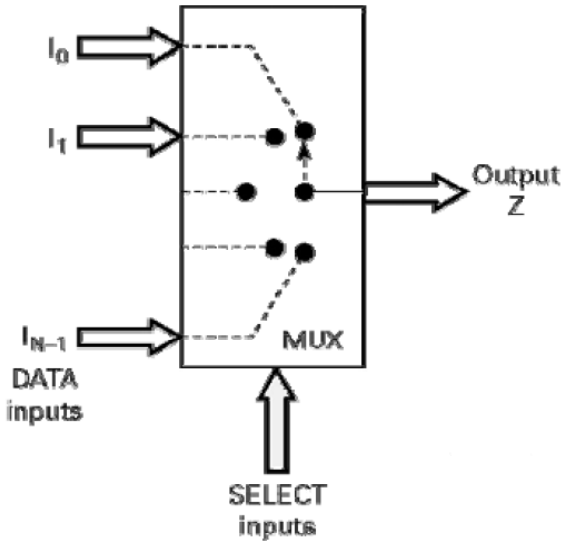
Digital Integrated Circuits & Microcontrollers

Chapter 3. Logic gates



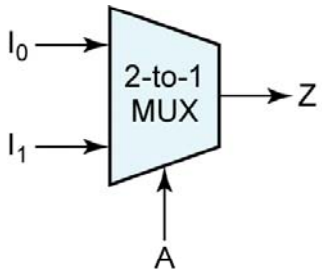
Multiplexers (Data Selectors)

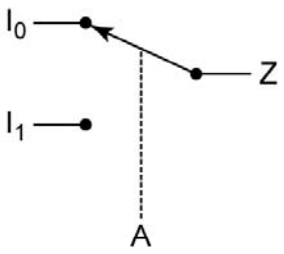
- A multiplexer has
  - N select inputs
  - 2^N data inputs
  - H output
- The selected data input is selected (routed) to the output.

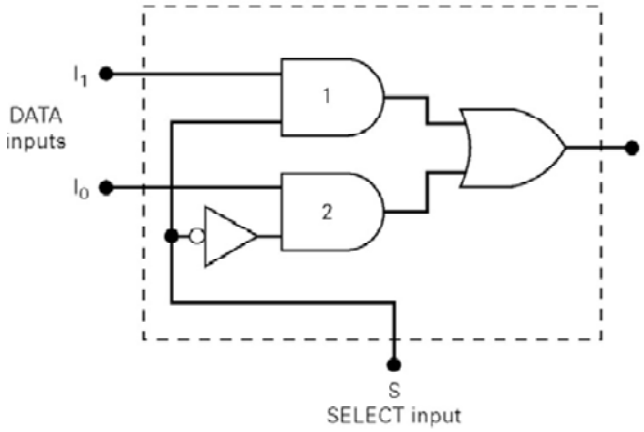


Multiplexers

- 2-input multiplexer





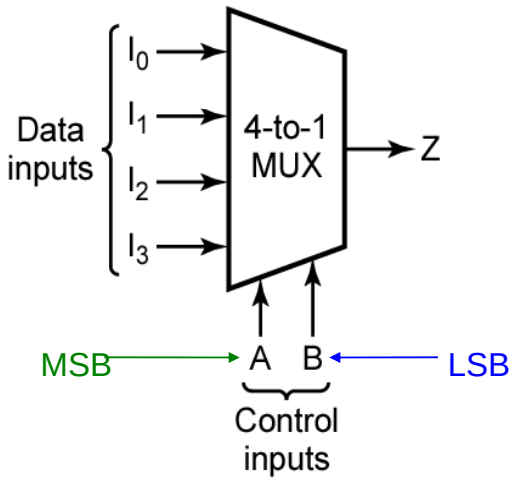


$Z = I_0 \cdot \bar{S} + I_1 \cdot S$

S	Output
0	$Z = I_0$
1	$Z = I_1$

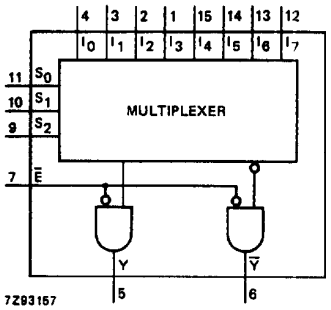
Multiplexers

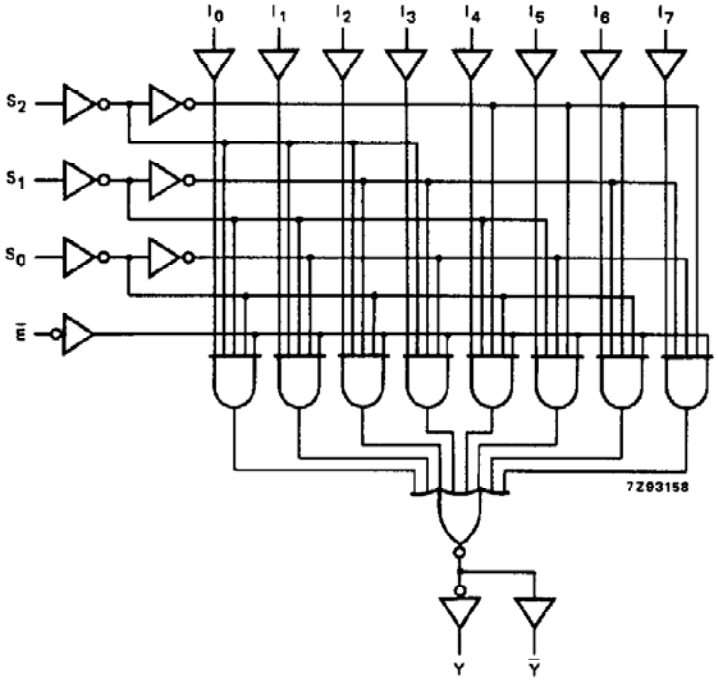
- 4-input multiplexer



A	B	Z
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

8 bit Multiplexer (74HC151)

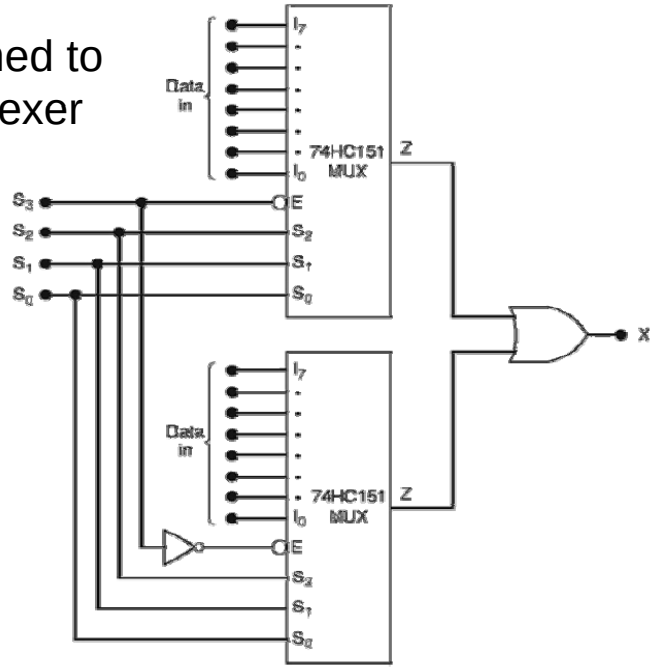




$\overline{E}$	S_2	S_1	S_0	Y
1	X	X	X	0
0	0	0	0	I_0
0	0	0	1	I_1
0	0	1	0	I_2
0	0	1	1	I_3
0	1	0	0	I_4
0	1	0	1	I_5
0	1	1	0	I_6
0	1	1	1	I_7

Multiplexers

- two 74HC151s combined to form a 16-input multiplexer

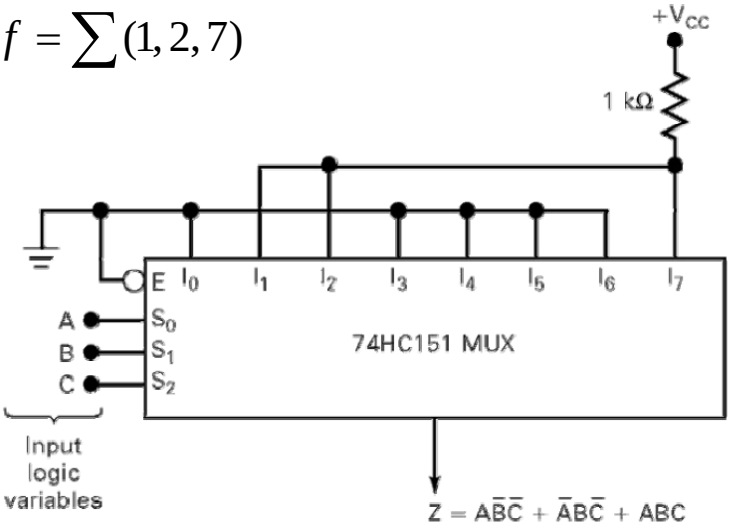


- Exercise:
- Design a 16-to-1 multiplexer using 4-to-1 multiplexers only

Multiplexers used to Implement Logic Function

$$f = A\bar{B}\bar{C} + \bar{A}B\bar{C} + ABC$$

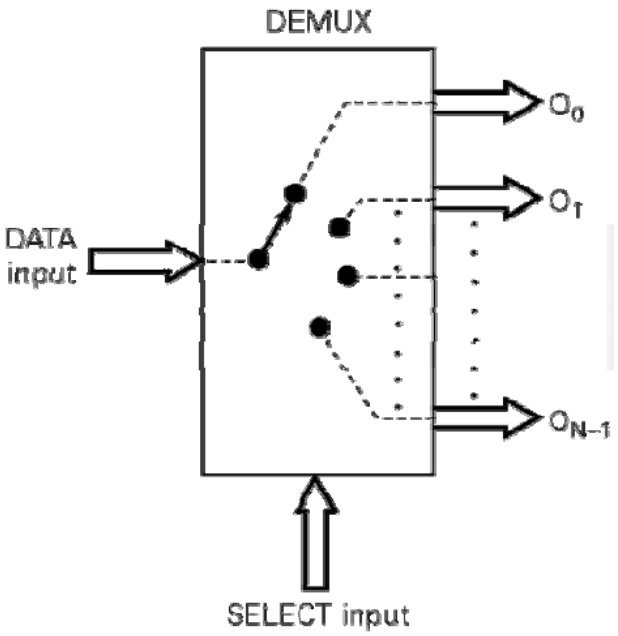
$$f = \sum(1, 2, 7)$$



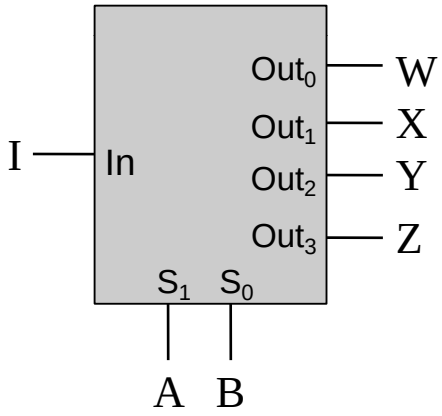
C	B	A	Z
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Demultiplexers (Data Distributors)

- A demultiplexer has
  - N control inputs
  - 1 data input
  - 2^N outputs
- A demultiplexer routes (or connects) the data input to the selected output.



Demultiplexers

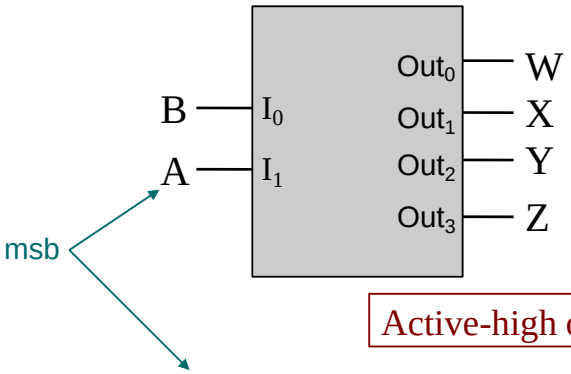


A	B	W	X	Y	Z
0	0	I	0	0	0
0	1	0	I	0	0
1	0	0	0	I	0
1	1	0	0	0	I

Decoders

- A decoder has
  - N inputs
  - 2^N outputs
- A decoder selects one of 2^N outputs by decoding the binary value on the N inputs.

Decoders



W = A'.B'

X = A.B'

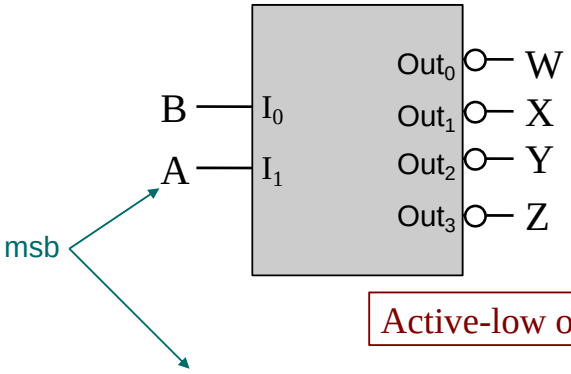
Y = A'.B

Z = A.B

Active-high outputs

A	B	W	X	Y	Z
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

Decoders



W = (A'.B')'

X = (A.B')'

Y = (A'.B)'

Z = (A.B)'

Active-low outputs

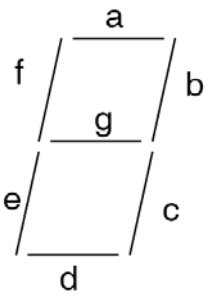
A	B	W	X	Y	Z
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

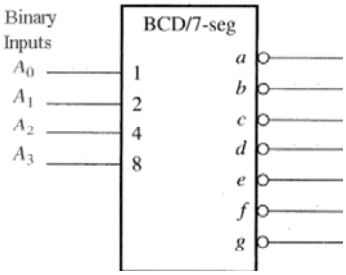
BCD (Binary Coded Decimal)

Decimal	Binary	HEX	BCD
0	00000	0	0000 0000
1	00001	1	0000 0001
2	00010	2	0000 0010
3	00011	3	0000 0011
4	00100	4	0000 0100
5	00101	5	0000 0101
6	00110	6	0000 0110
7	00111	7	0000 0111
8	01000	8	0000 1000
9	01001	9	0000 1001
10	01010	A	0001 0000
11	01011	B	0001 0001
12	01100	C	0001 0010
13	01101	D	0001 0011
14	01110	E	0001 0100
15	01111	F	0001 0101

BCD-to-7 Segment Display Decoder

Decimal digit	INPUTS D C B A	SEGMENT OUTPUTS a b c d e f g
0	0 0 0 0	1 1 1 1 1 1 0
1	0 0 0 1	0 1 1 0 0 0 0
2	0 0 1 0	1 1 0 1 1 0 1
3	0 0 1 1	1 1 1 1 0 0 1
4	0 1 0 0	0 1 1 0 0 1 1
5	0 1 0 1	1 0 1 1 0 1 1
6	0 1 1 0	1 0 1 1 1 1 1
7	0 1 1 1	1 1 1 0 0 0 0
8	1 0 0 0	1 1 1 1 1 1 1
9	1 0 0 1	1 1 1 1 0 1 1
10	1 0 1 0	X X X X X X X
11	1 0 1 1	X X X X X X X
12	1 1 0 0	X X X X X X X
13	1 1 0 1	X X X X X X X
14	1 1 1 0	X X X X X X X
15	1 1 1 1	X X X X X X X

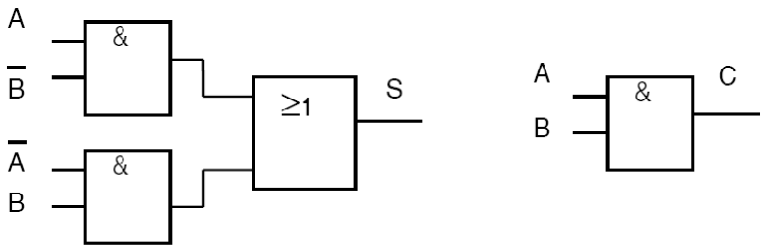




Half Adder

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

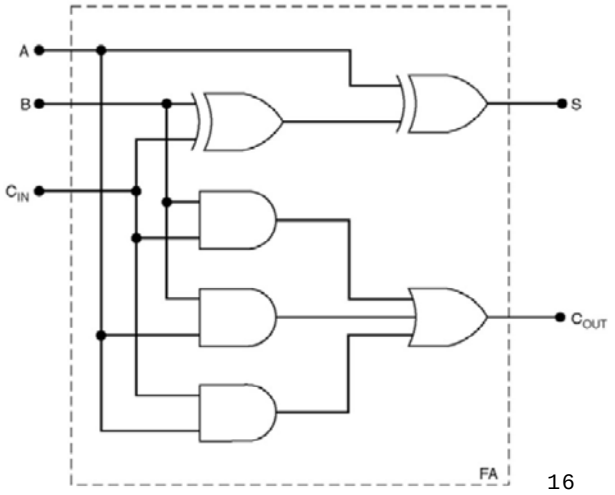
$$S = \overline{A}B + A\overline{B} = A \oplus B$$
$$C = AB$$



Full Adder

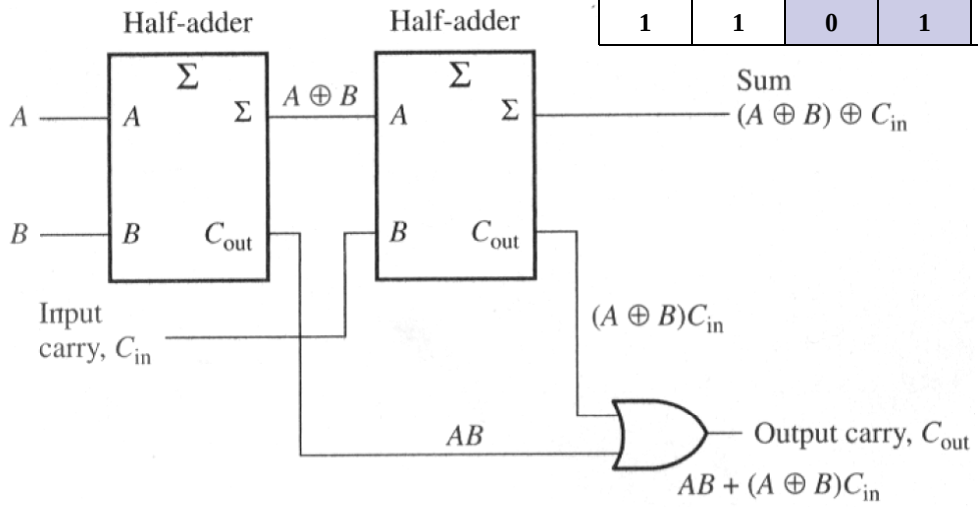
A	B	C_i	S	C_o
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S = \overline{A}.\overline{B}.C_i + \overline{A}.B.\overline{C_i} + A.\overline{B}.\overline{C_i} + A.B.C_i$$
$$= A \oplus B \oplus C_i$$
$$C_o = \overline{A}BC_i + A\overline{B}C_i + AB\overline{C_i} + ABC_i$$
$$= AB + AC_i + BC_i$$
$$= AB + C_i(A + B)$$



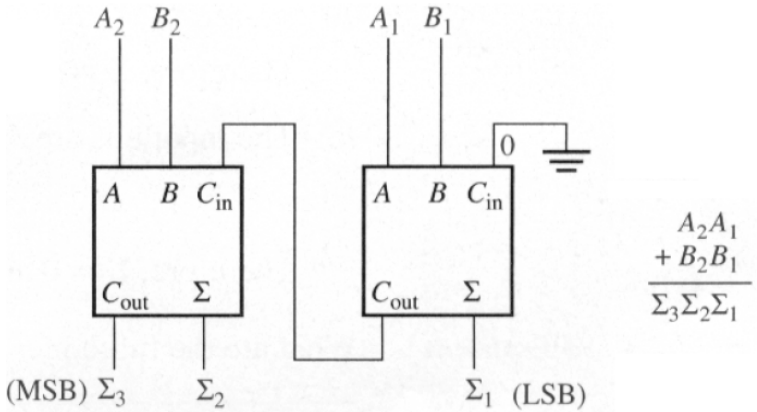
Full Adder from Half Adders

A	B	HA_S	HA_C	C_i	S	C_o
0	0	0	0	0	0	0
0	0	0	0	1	1	0
0	1	1	0	0	1	0
0	1	1	0	1	0	1
1	0	1	0	0	1	0
1	0	1	0	1	0	1
1	1	0	1	0	0	1
1	1	0	1	1	1	1



Parallel Adder

- Uses 1 full adder per bit of the numbers
- The carry is propagated from one stage to the next most significant stage



4-bit Parallel Binary Adders