

Introduction to Digital Systems

Part II (4 lectures)

2022/2023

Combinational Logic Blocks

Arnaldo Oliveira, Augusto Silva, Ioulia Skliarova

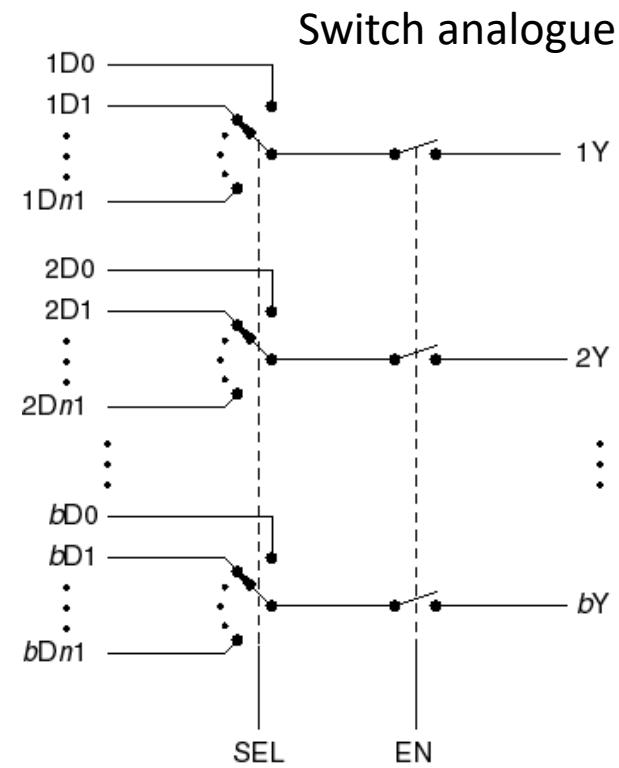
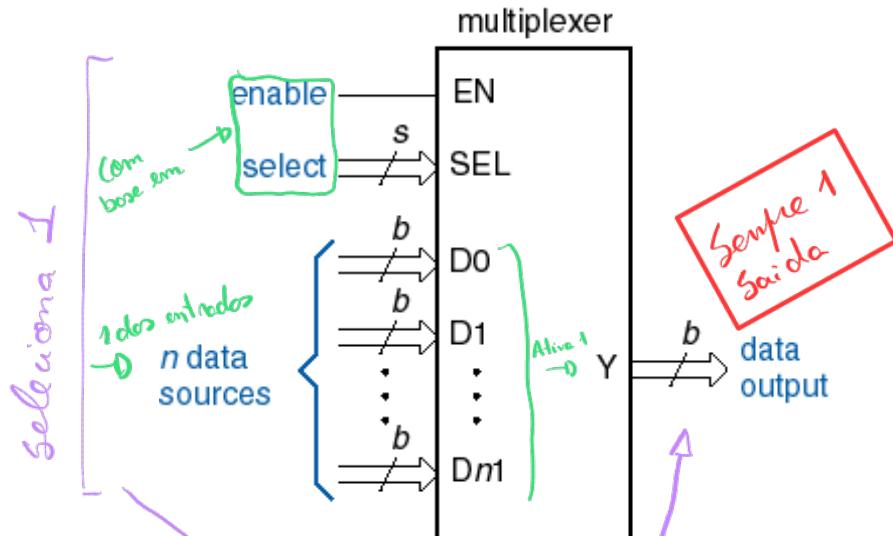
Lecture 6 contents

- Block oriented combinational logic design
- Multiplexers
- Demultiplexers

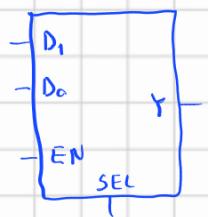
Multiplexers

↳ interruptor digital

- A multiplexer is a digital switch: one out of n data sources is passed to a single output
 - Information selector



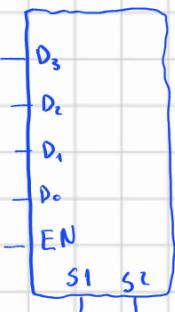
mux 2:1



EN	SEL	Y
0	X	0
1	0	D0

$$y = \overbrace{EN \cdot \overbrace{SEL}^{\text{for } 0=0}}^{\text{for } 0=1} \cdot D_0 + \overbrace{EN \cdot \overbrace{SEL}^{\text{for } 1=1}}^{\text{for } 1=0} \cdot D_1$$

mux 4:1



EN	S1	S0	Y
0	X	X	0
1	0	0	D0
1	0	1	D1
1	1	0	D2
1	1	1	D3

$$Y = EN \left(\underbrace{\overline{S1} \overline{S0}}_{m_0} D_0 + \underbrace{\overline{S1} S0}_{m_1} D_1 + \underbrace{S1 \overline{S0}}_{m_2} D_2 + \underbrace{S1 S0}_{m_3} D_3 \right)$$

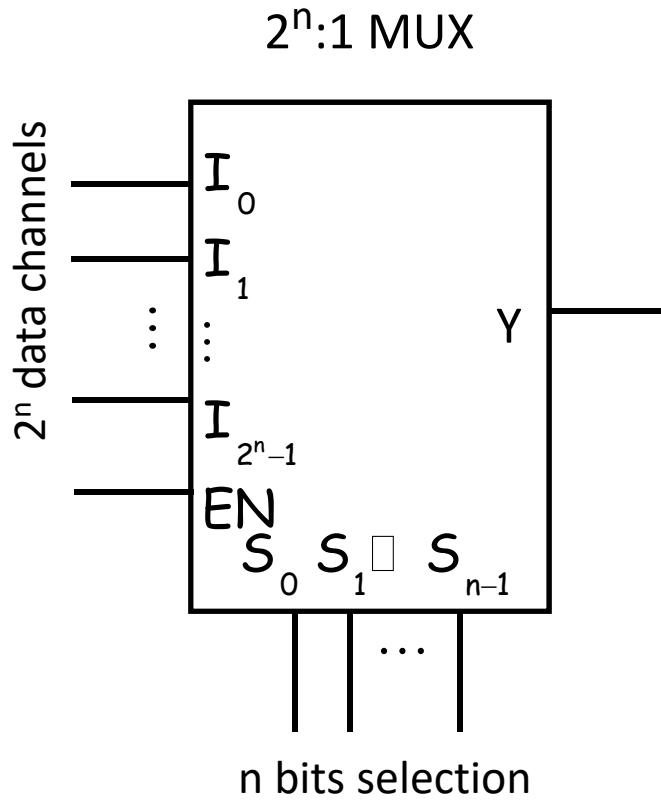
mínimo "0"

⚠ SEL = $\lceil \log_2 m \rceil$

Dados \downarrow

$\lceil \log_2 m \rceil$

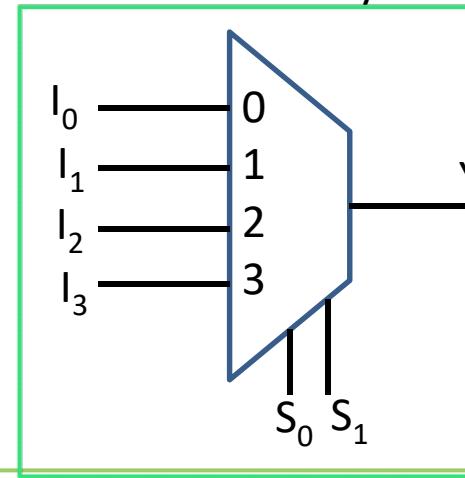
$2^n:1$ Mux models



4:1 Mux Truth Table

EN	S_1	S_0	Y
0	x	x	0
1	0	0	I_0
1	0	1	I_1
1	1	0	I_2
1	1	1	I_3

Alternate Symbol



→ Desenho
de um multiplexor!

Functional description

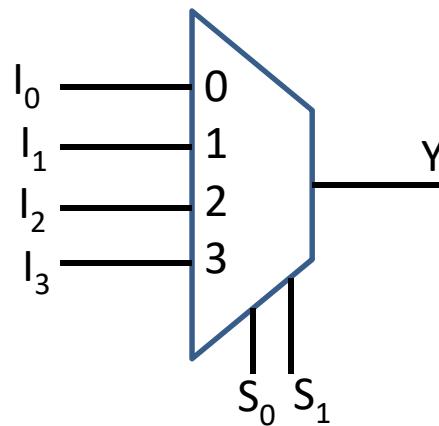
4:1 Mux Truth Table

EN	S_1	S_0	Y
0	x	x	0
1	0	0	I_0
1	0	1	I_1
1	1	0	I_2
1	1	1	I_3

$$Y = EN \cdot \left[\sum_{k=0}^{2^n-1} m(S)_k I_k \right]$$

$m(S)_k$ is the k^{th} minterm on the selection variables $S_0 \dots S_{n-1}$

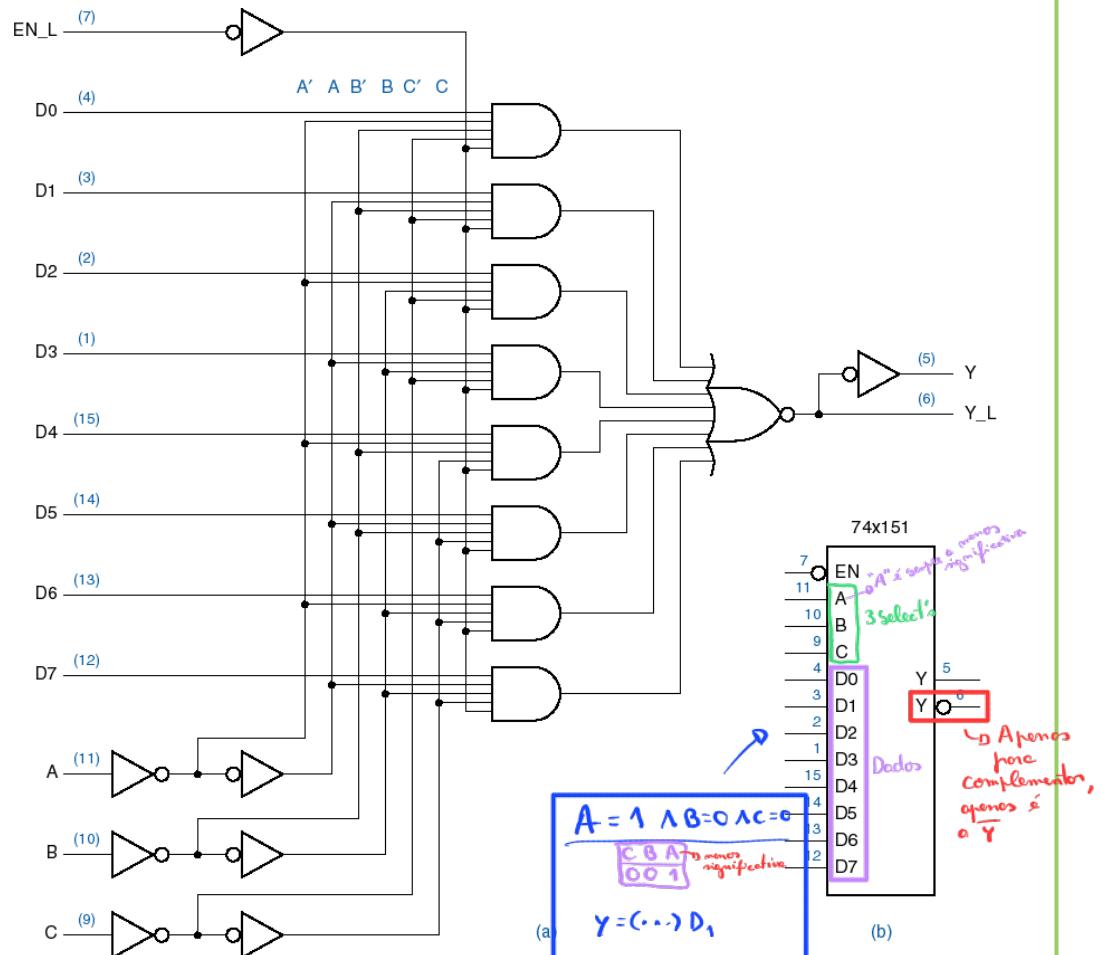
Alternate Symbol



Exercise: Draw the 4:1 Mux internal logic circuit

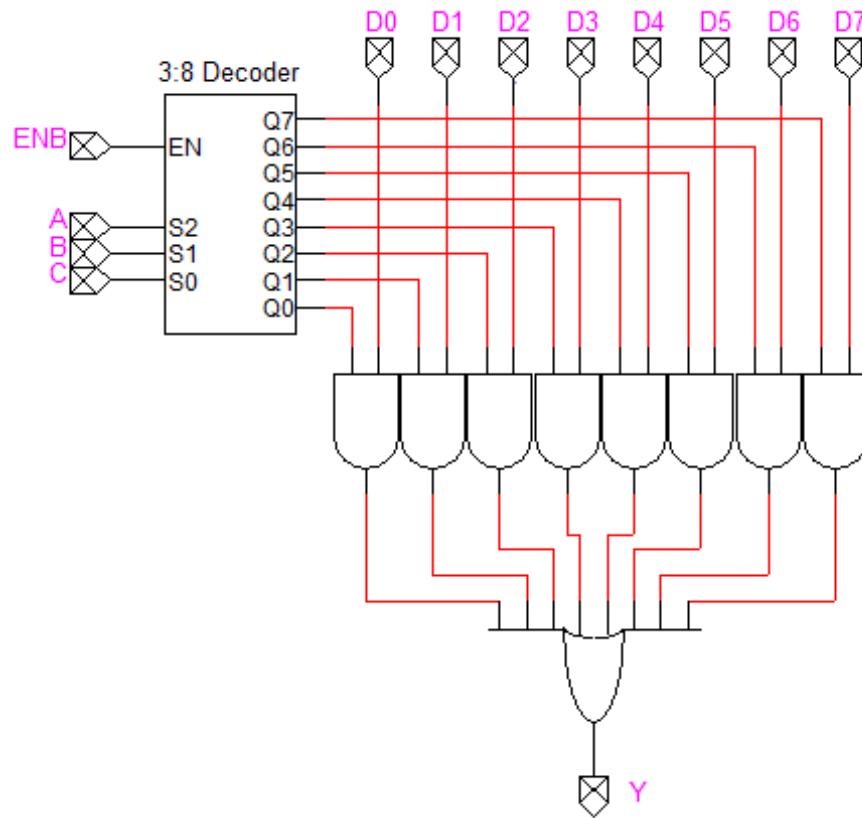
The 74151 model

- 8:1 mux
- Obtain the truth table
- Write the output equations



Mux and decoders

- Verify that the logic circuit is a 8:1 Mux



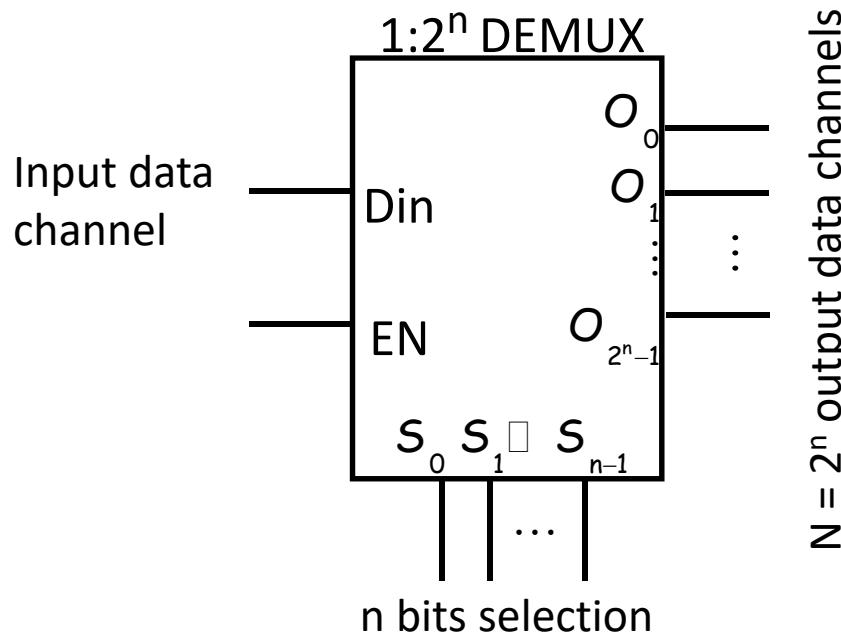


Demultiplexers

Parecido ao dec

- Functional inverse of a multiplexer
 - An inverse digital switch: a single input is “routed” to one out of N outputs

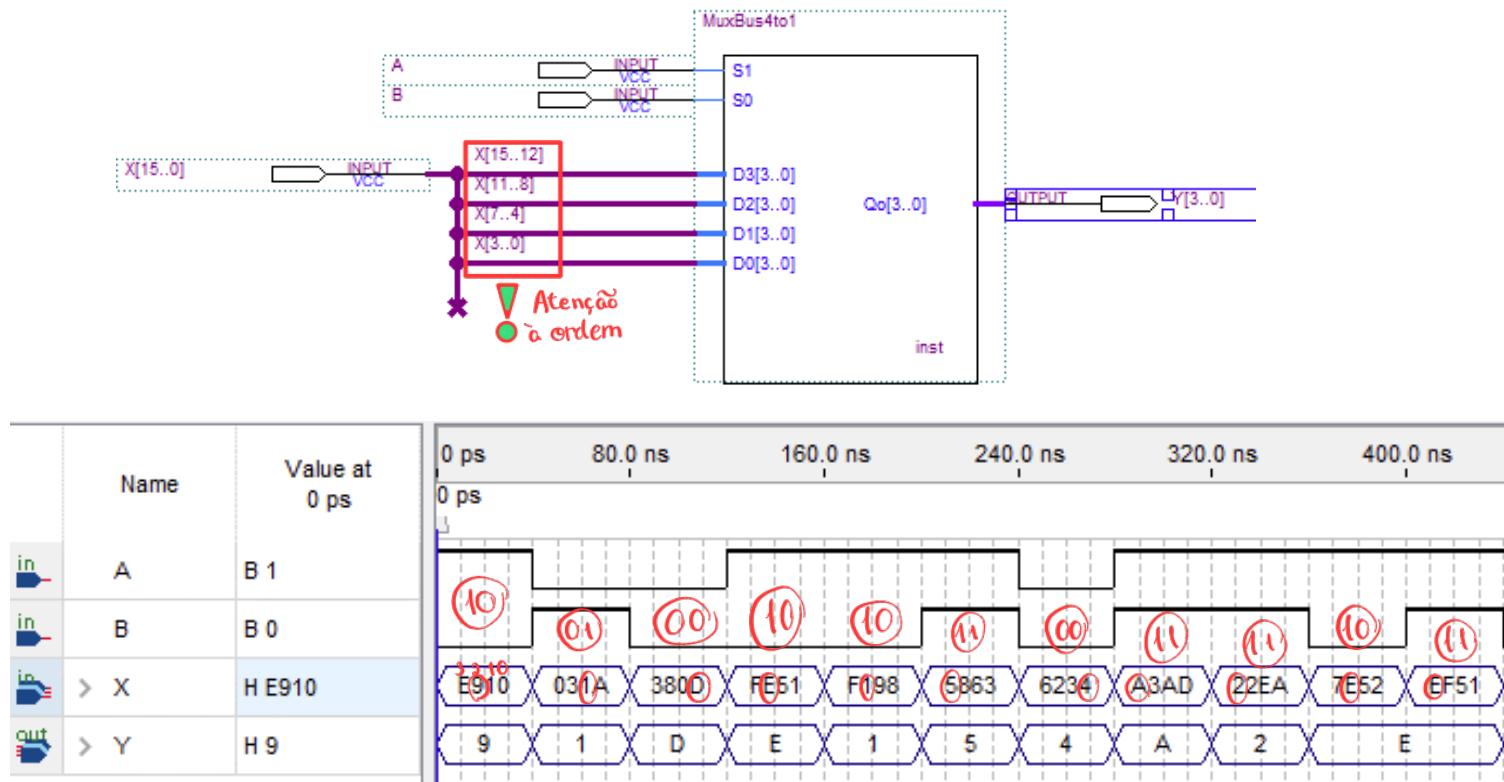
$$O_k = EN \cdot D_{in} m_k(S), \quad k = 0, \dots 2^n - 1$$



Question: How can we use a demux as a decoder?

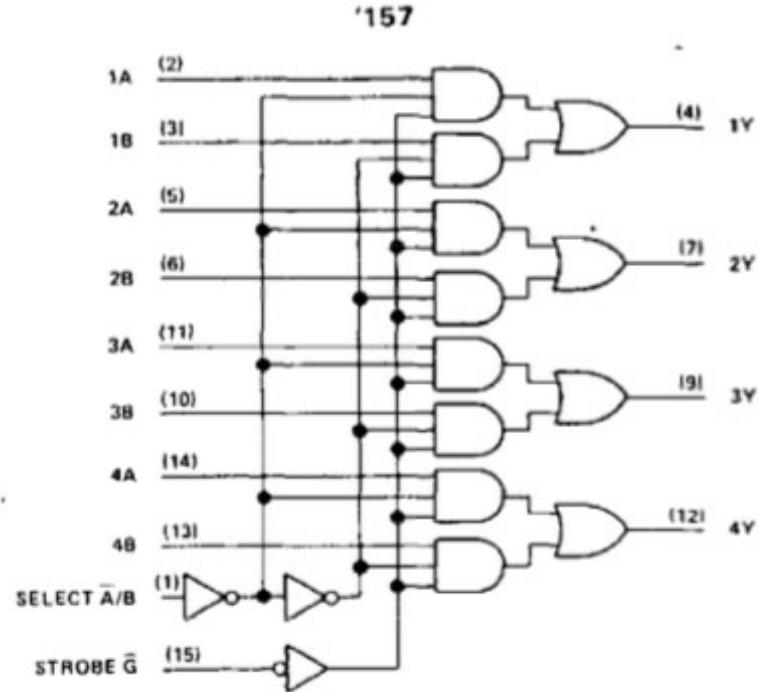
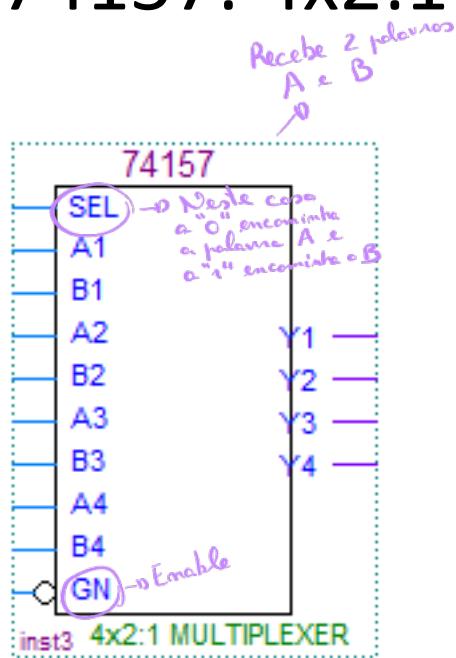
Multiplexing multibit data channels

- Mux 4:1, 4 bit input data channels
- Explain the timing diagram



The 74157 model

- 74157: 4x2:1



FUNCTION TABLE

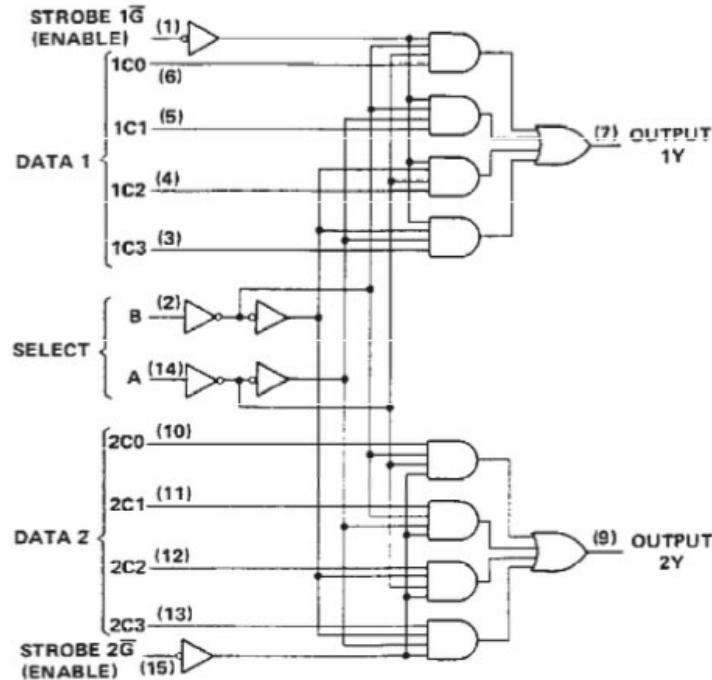
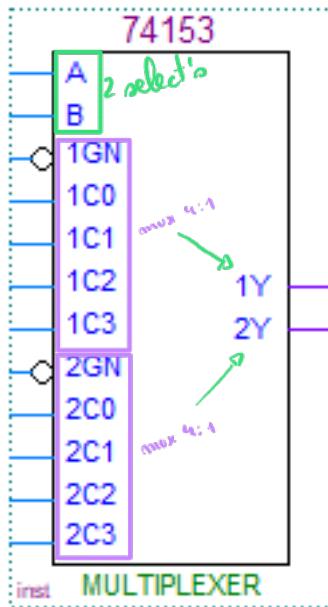
		INPUTS		OUTPUT Y	
STROBE G	SELECT A/B	A	B	'157, 'LS157, 'S157	'LS158 'S158
H	X	X	X	L	H
L	L	L	X	L	H
L	L	H	X	H	L
L	H	X	L	L	H
L	H	X	H	H	L

H = high level, L = low level, X = irrelevant

The 74153 model

- 74153: 2x4:1

2 multiplexers partilham os mesmos select's



SELECT INPUTS		DATA INPUTS				STROBE	OUTPUT
B	A	C0	C1	C2	C3	\bar{G}	Y
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
L	H	X	L	X	X	L	L
L	H	X	H	X	X	L	H
H	L	X	X	L	X	L	L
H	L	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H

Select inputs A and B are common to both sections.
H = high level, L = low level, X = irrelevant

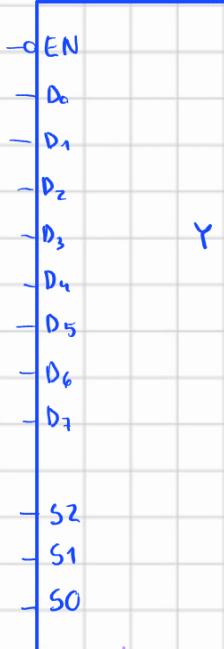
mux 8:1

$$HSEL = \lceil \log_2 8 \rceil = 3$$

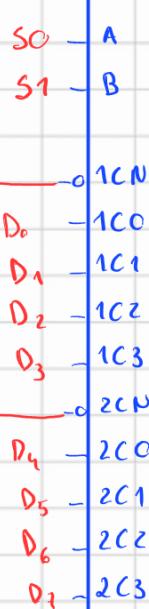
Queremos
inter.

Temos!

mux 8:1

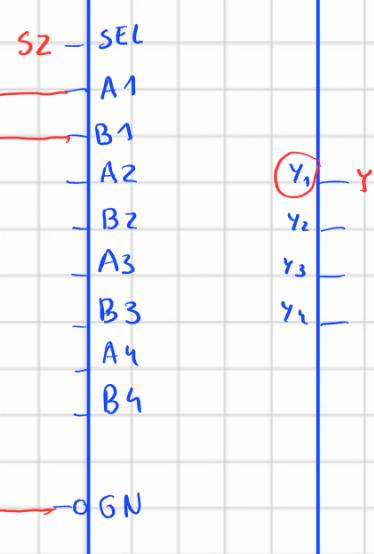


mux 4:1 dual



1Y

2Y



mux 2:1 de 4 bit

D_{0,1,2,3}

D_{4,5,6,7}

mux 8:1

EN-L S₂ S₁ S₀ Y

0	0	0	0	D ₀
0	0	0	1	D ₁
0	1	0	0	D ₂
0	1	1	1	D ₃
1	0	0	0	D ₄
1	0	0	1	D ₅
1	1	0	0	D ₆
1	1	1	1	D ₇

EN-L is connected to the first column of the truth table.

mux 4:1

mux 4:1

Exercício:

General

$$\{ \text{mux } 2^m : 1, 0, 1 \}$$

m variáveis

Esta matéria
vai sair
no teste!!

$$f(a, b, c) = \sum_{i=0}^1 \sum_{j=0}^1 \sum_{k=0}^1 a_i b_j c_k$$

	a	b	c	f
0:	0	0	0	1
1:	0	0	1	0
2:	0	1	0	1
3:	0	1	1	0
4:	1	0	0	1
5:	1	0	1	1
6:	1	1	0	0
7:	1	1	1	1

ou

$$\{ \text{mux } 2^{m-1} : 1, 0, 1, \text{Not} \}$$

m var

separam
os 4 grupos

	a	b	c	f
0:	0	0	0	1
1:	0	0	1	0
2:	0	1	0	1
3:	0	1	1	0
4:	1	0	0	1
5:	1	0	1	1
6:	1	1	0	0
7:	1	1	1	1

$m = 3$

$$\{ \text{mux } 8 : 1, 0, 1 \}$$

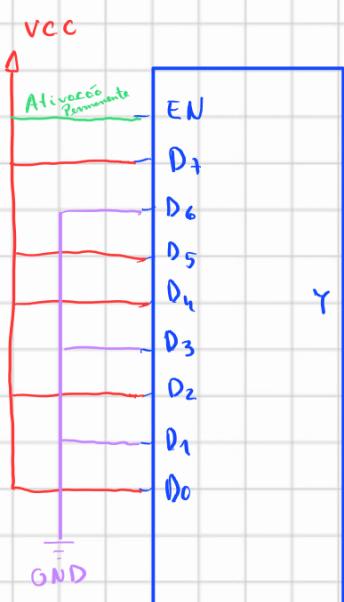
3 var.

OU

$m = 3$

$$\{ \text{mux } 4 : 1, 1, 0, 1, \text{Not} \}$$

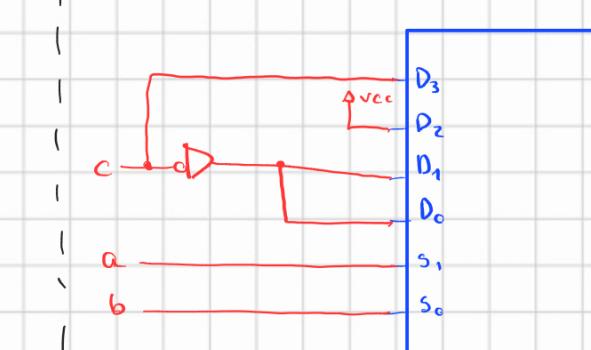
$\# \text{SEL} = 2$



OU

	a	b	c	f
0:	0	0	0	1
1:	0	0	1	0
2:	0	1	0	1
3:	0	1	1	0
4:	1	0	0	1
5:	1	0	1	1
6:	1	1	0	0
7:	1	1	1	1

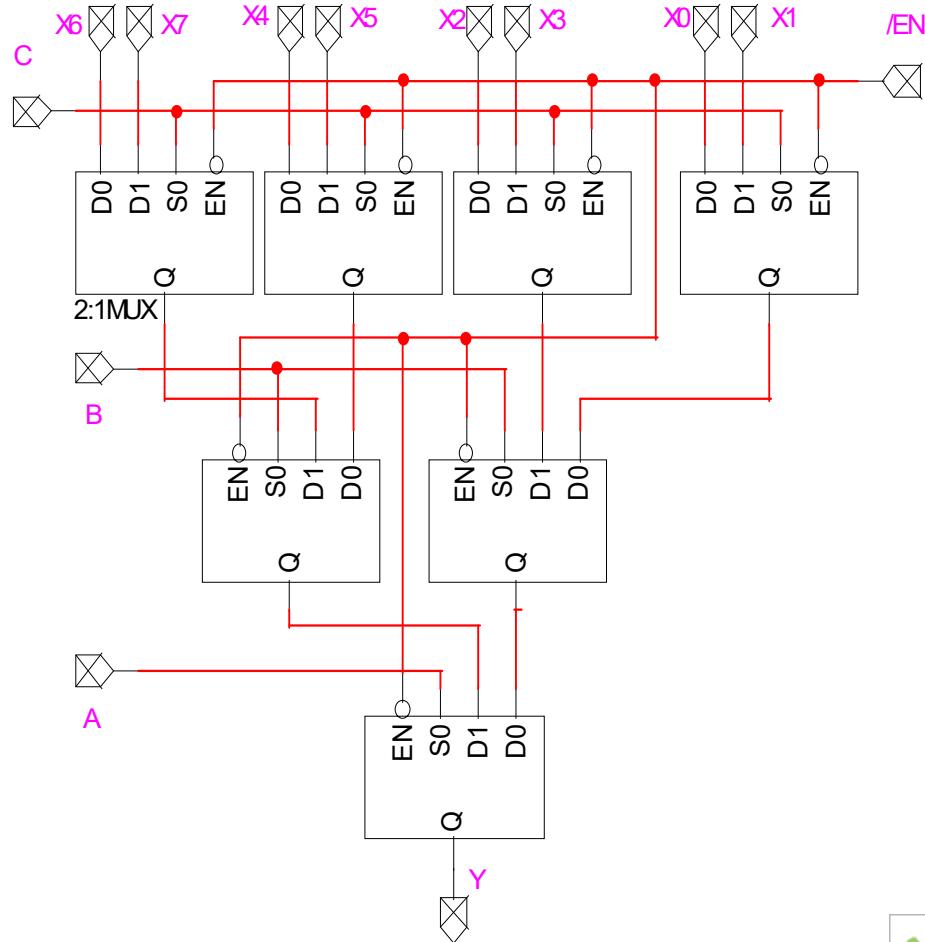
$$\{ \text{mux } 2^{m-2} : 2, 0, 1, \text{notas} \}$$



Mux hierarchies

- 8:1 with $7 \times (2:1)$

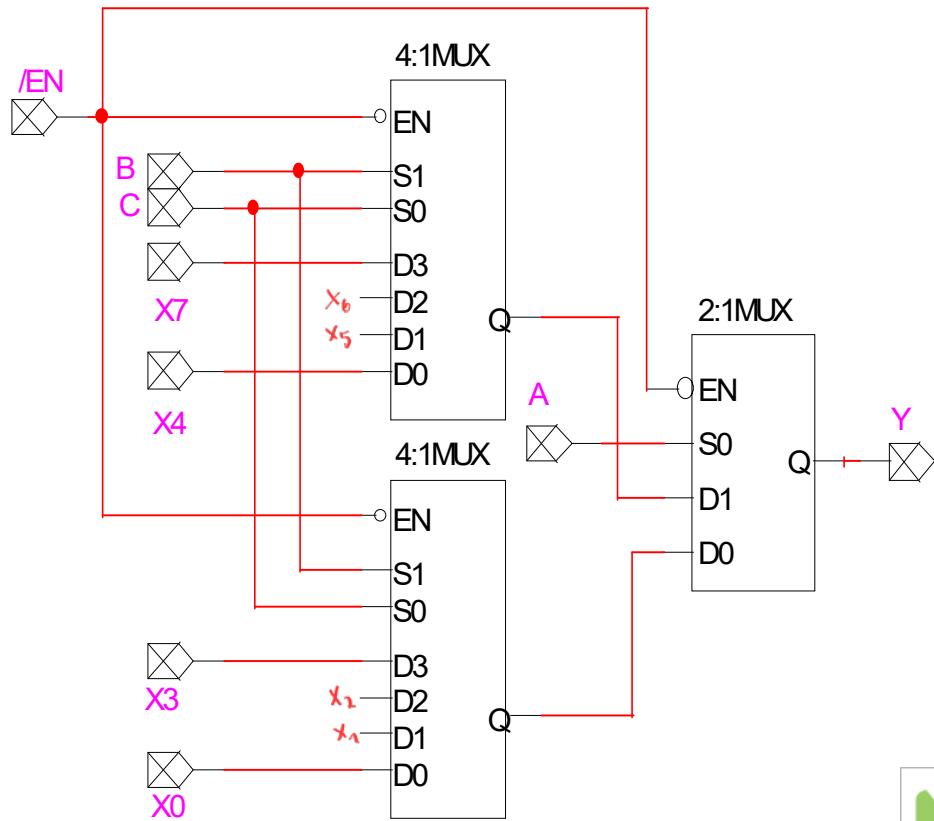
- Always check the design by obtaining the truth table.
- Note that IN THIS case A is the most significant selection variable



Mux hierarchies

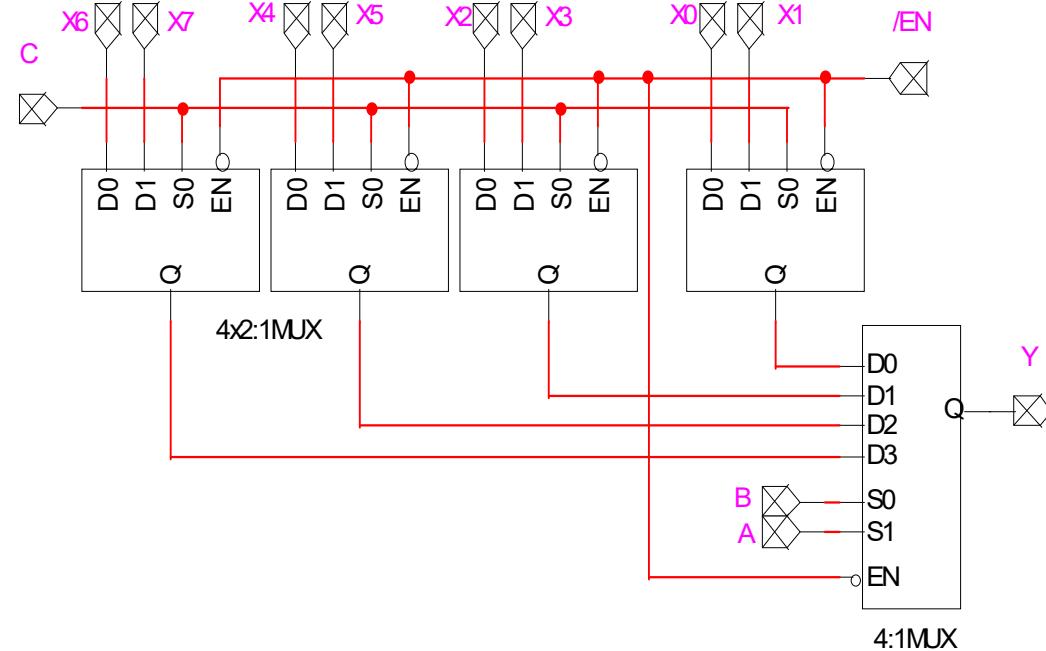
- 8:1 with $2 \times (4:1 \text{ MUX}) + 1 \times (2:1 \text{ MUX})$

- Always check the design by obtaining the truth table.
- Note that IN THIS case A is the most significant selection variable



Mux hierarchies

- 8:1 with $4 \times (2:1 \text{ MUX}) + 1 \times (4:1 \text{ MUX})$

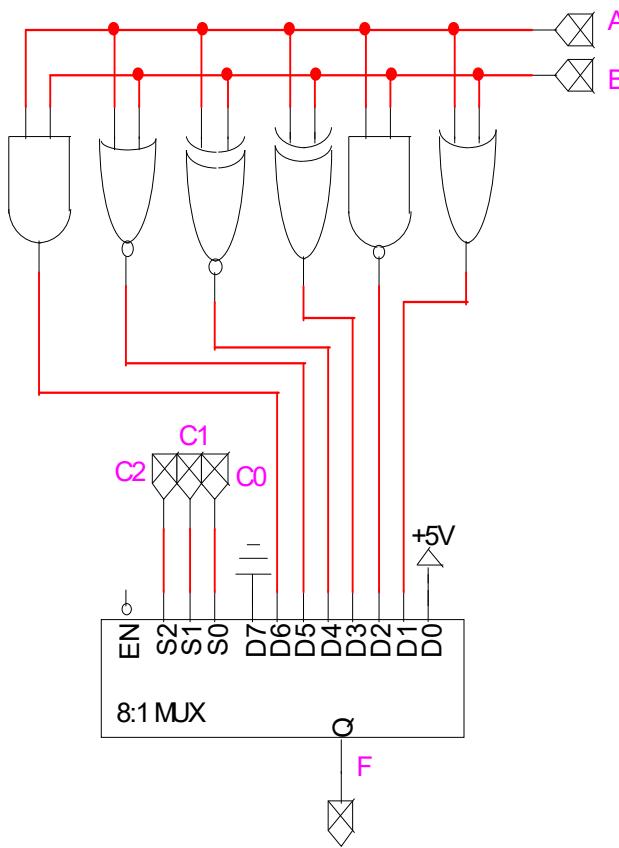


- Always check the design by obtaining the truth table.
- Note that IN THIS case A is the most significant selection variable

Logic Function Unit (LFU)

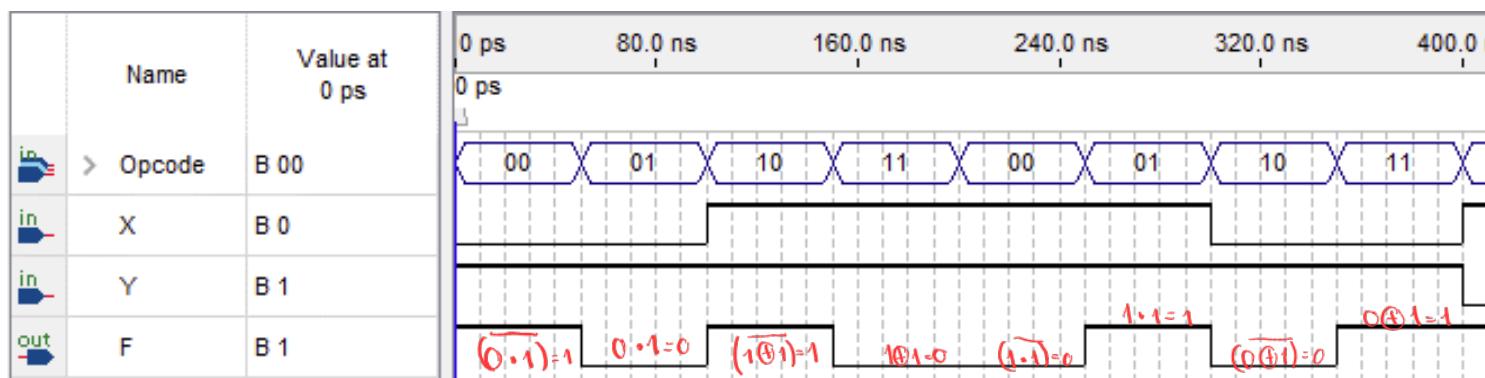
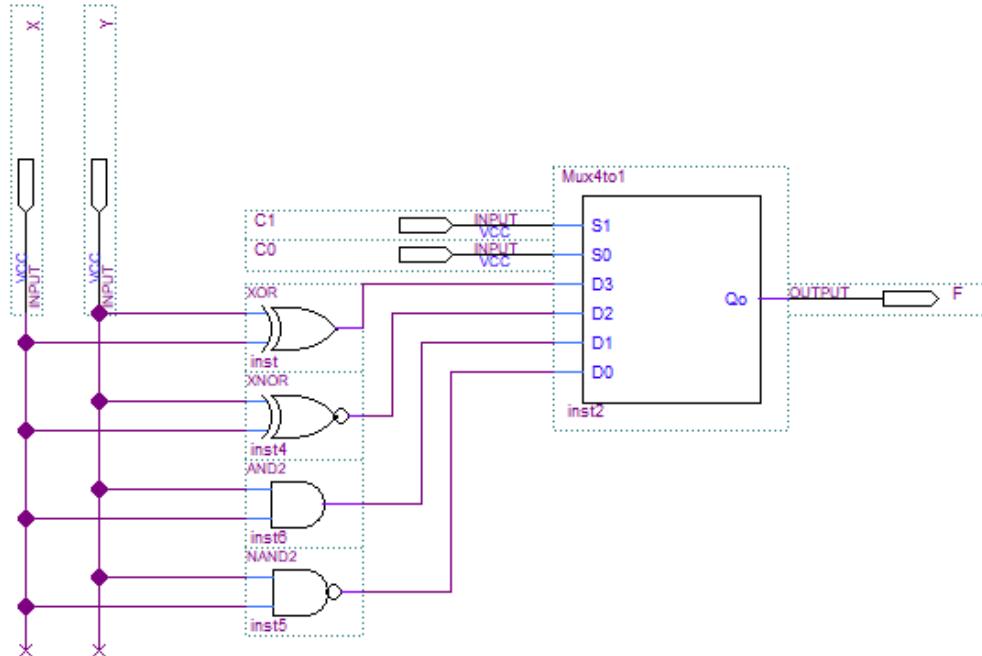
- Use $C_2C_1C_0$ as function code (Opcode)

C_2	C_1	C_0	F
0	0	0	1
0	0	1	$A+B$
0	1	0	$(A \cdot B)'$
0	1	1	$A \oplus B$
1	0	0	$(A \oplus B)'$
1	0	1	$(A+B)'$
1	1	0	$A \cdot B$
1	1	1	0



Exercise

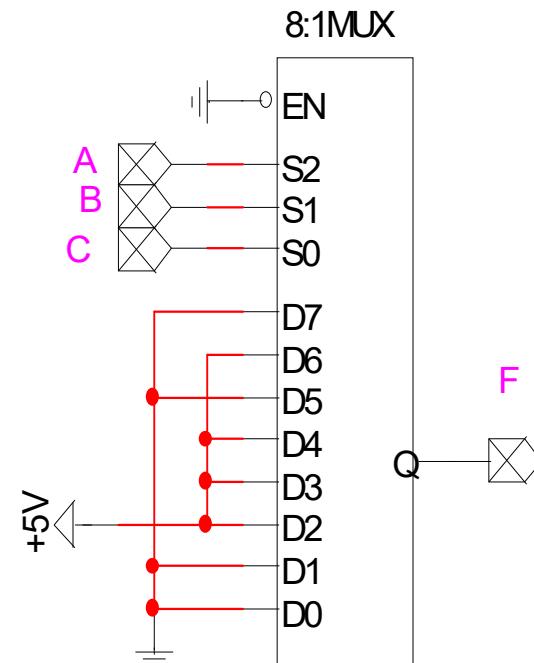
- Explain the LFU timing diagram



Boolean Functions with Multiplexers

- Simplest approach:
 - Direct mapping of the Truth Table
 - Selection = input variables
 - $D_k = F_k$

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



Boolean Functions with Multiplexers

- The general case:
 - Selection = a subset of input variables
 - $D_k = g_k$ where each g_k is a simpler Boolean function of the remaining input variables
- Example

$n-1$ input variables used for selection

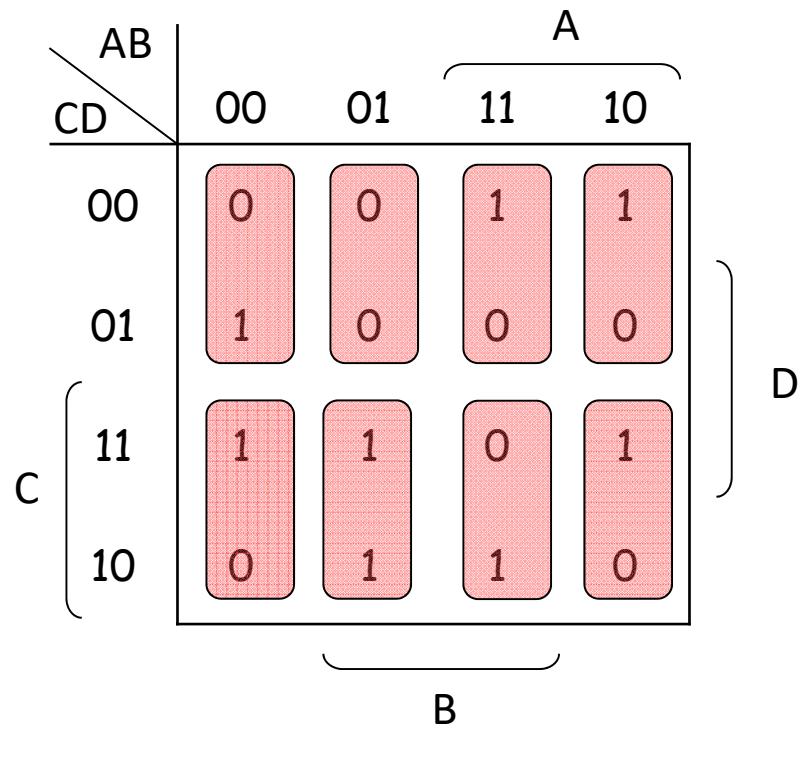
I ₁	I ₂	..	I _n	F			
...	0	0	0	1	1
...	1	0	1	0	1
				0	I _n	\bar{I}_n	1

Possible output values as a function of I_n

Example

- Implement the Boolean function $F(A,B,C,D)$ using a 8:1 Mux

$$F(A,B,C,D) = \sum m(1,3,6,7,8,11,12,14)$$



1. Use the Karnaugh map JUST to layout the truth table
2. Choose the subset of inputs to be assigned to the mux selection inputs
Eg. A,B,C
3. Find the logic values of the mux data inputs as functions of the remaining inputs
D in this case

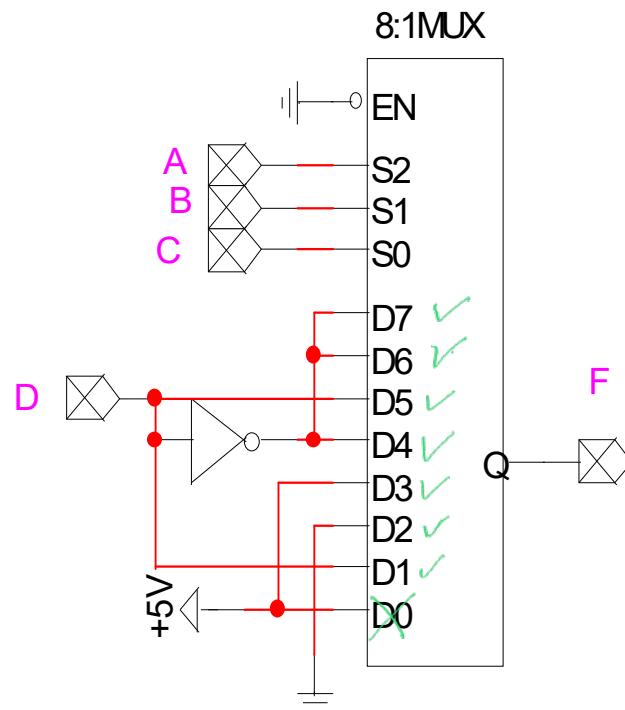
Regions of the truth sharing the same value of the selection inputs. (A,B,C) in this case. DO NOT MISINTERPRET as prime implicants

Example

- Find the error in the logic circuit

$$F(A,B,C,D) = \sum m(1,3,6,7,8,11,12,14)$$

		AB		A					
		00	01	11	10				
CD	00	0	0	1	1				
	01	1	0	0	0				
	11	1	1	0	1				
	10	0	1	1	0				



Exercise

- Implement the Boolean function F using a MUX 4:1 and additional elementary logic

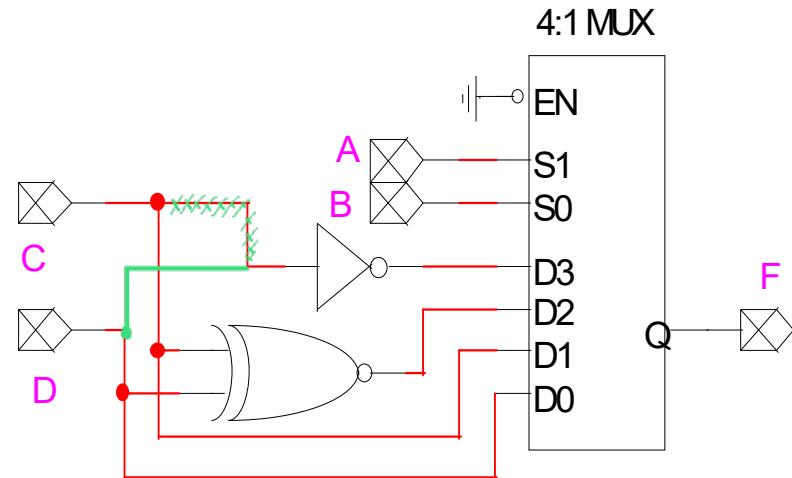
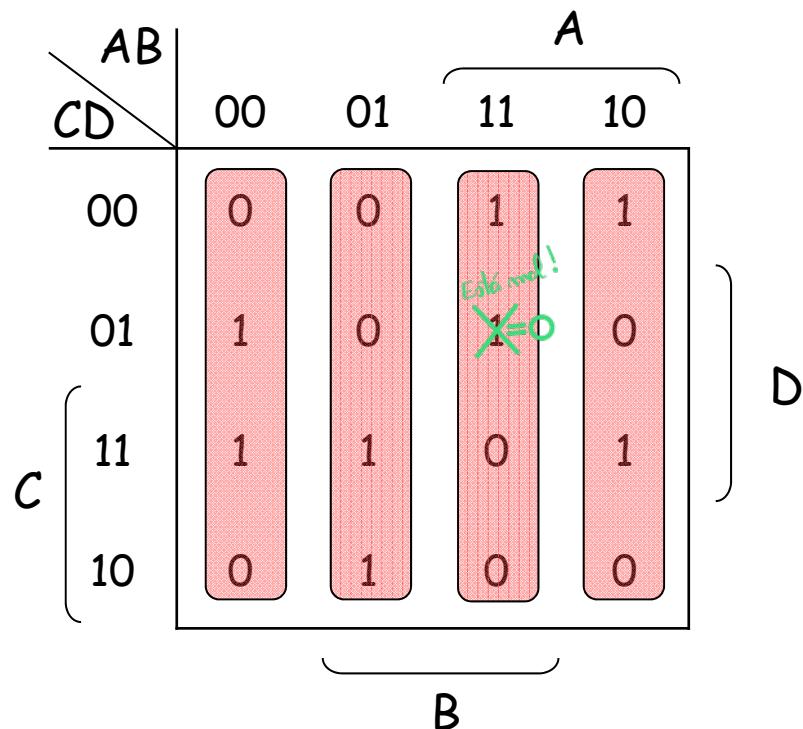
$$F(A,B,C,D) = \sum m(1,3,6,7,8,11,12,14)$$

- Several choices of input variables are possible to be assigned to the mux selection inputs. Try for example (A,B) and (C,D)

Exercise

- Using A,B for selection

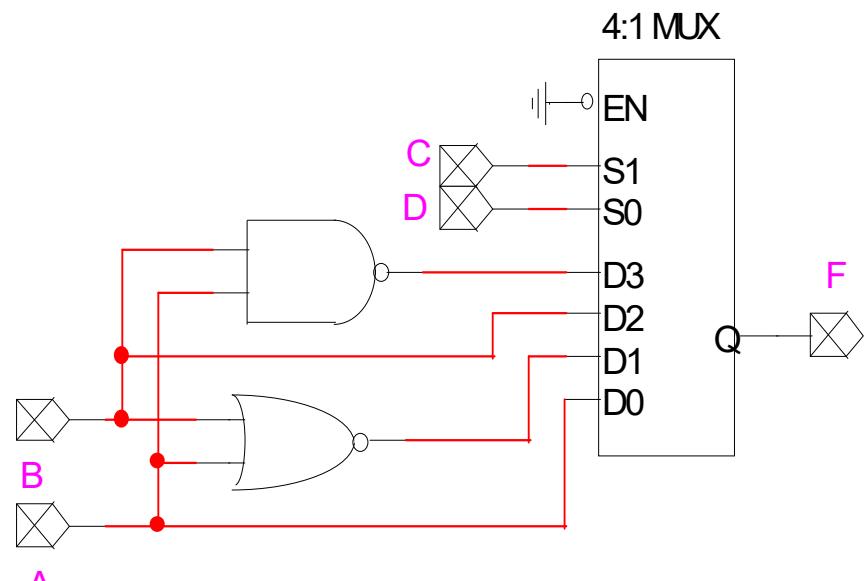
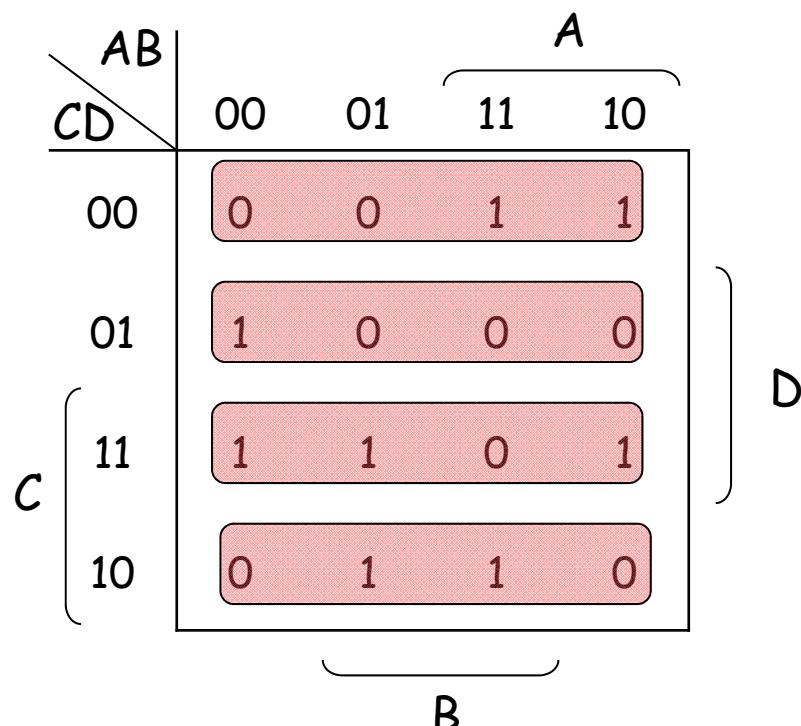
$$F(A, B, C, D) = \sum m(1, 3, 6, 7, 8, 11, 12, 14)$$



Exercise

- Using C,D for selection

$$F(A,B,C,D) = \sum m(1,3,6,7,8,11,12,14)$$

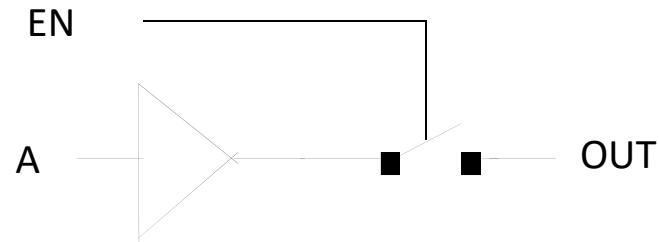


High-Impedance (High – Z)

1
0
Z

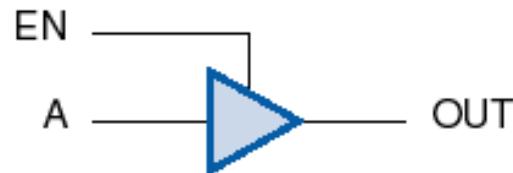
Outro estado!

- A switch model
- When the switch is open there is an almost infinite resistance (Impedance) to the signal flow through the “wire” OUT.
- The output signal is left “floating” with neither HIGH or LOW logic levels assigned.
- The output is assigned a High-Z state and the device exhibits a 3 State behavior



3 State Buffers

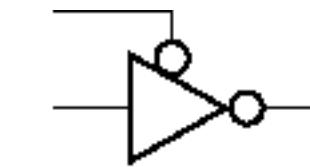
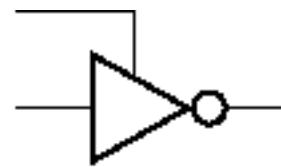
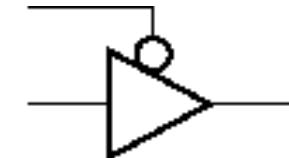
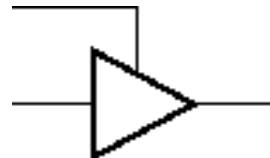
- Possible outputs: HIGH, LOW, High-Z



EN A OUT

L	L	Hi-Z
L	H	Hi-Z
H	L	L
H	H	H

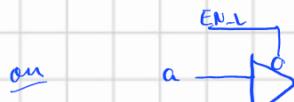
Alternatives



- tri - stat
- three - state

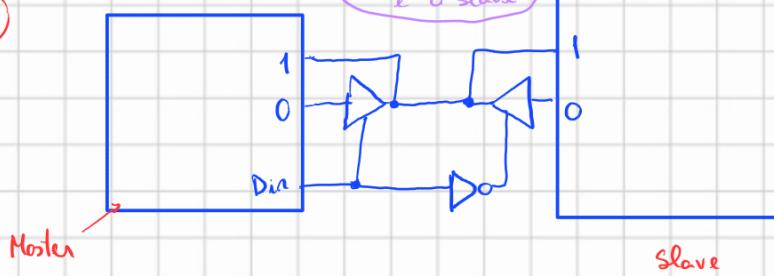


EN	f
0	Z
1	a



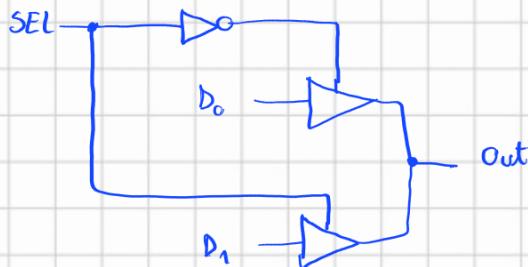
Formas de utilização:

①



O Master decide o fluxo a partir da entrada dir

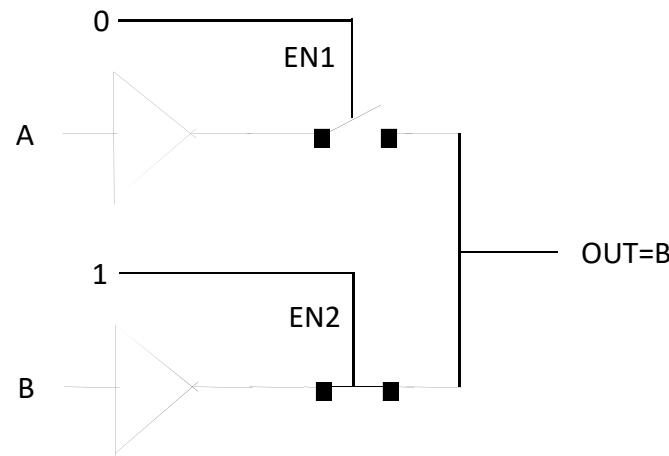
②



SEL	Out
0	D0
1	D1

Wire sharing

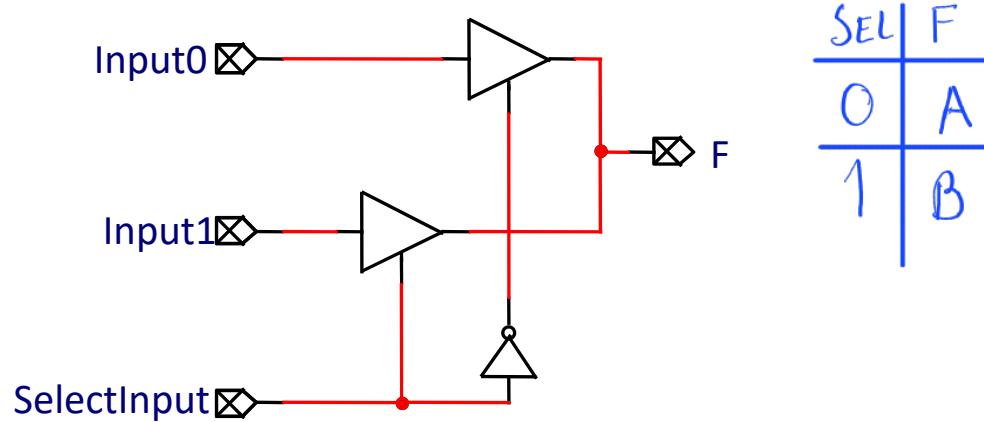
- High-Z outputs may be physically connected



- Of course $\text{EN1} = \text{EN2} = 1$ should never occur.
- Tight control of enabling inputs is required

A special kind of Mux

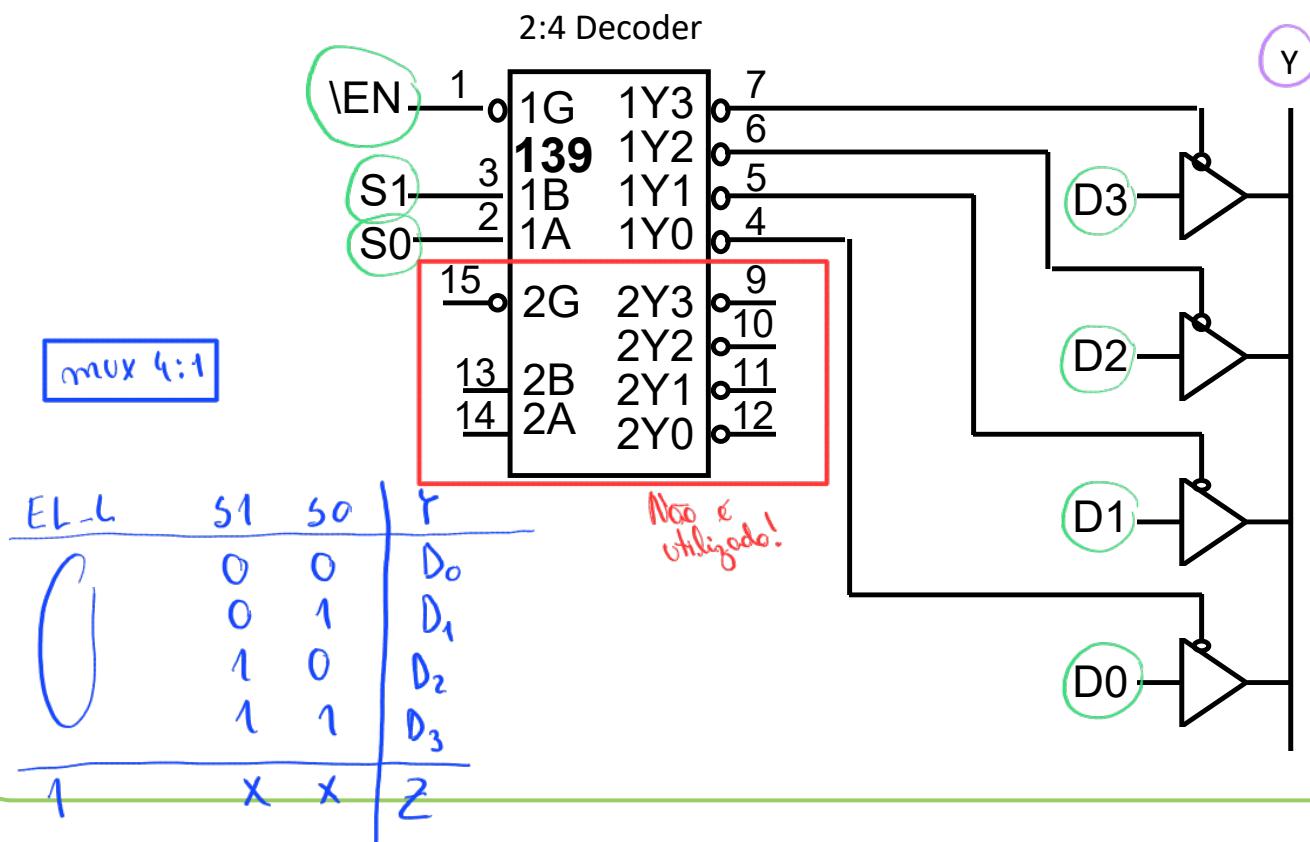
- Efficient multiplexing strategy
- Mux 2:1



- Write the Truth Table

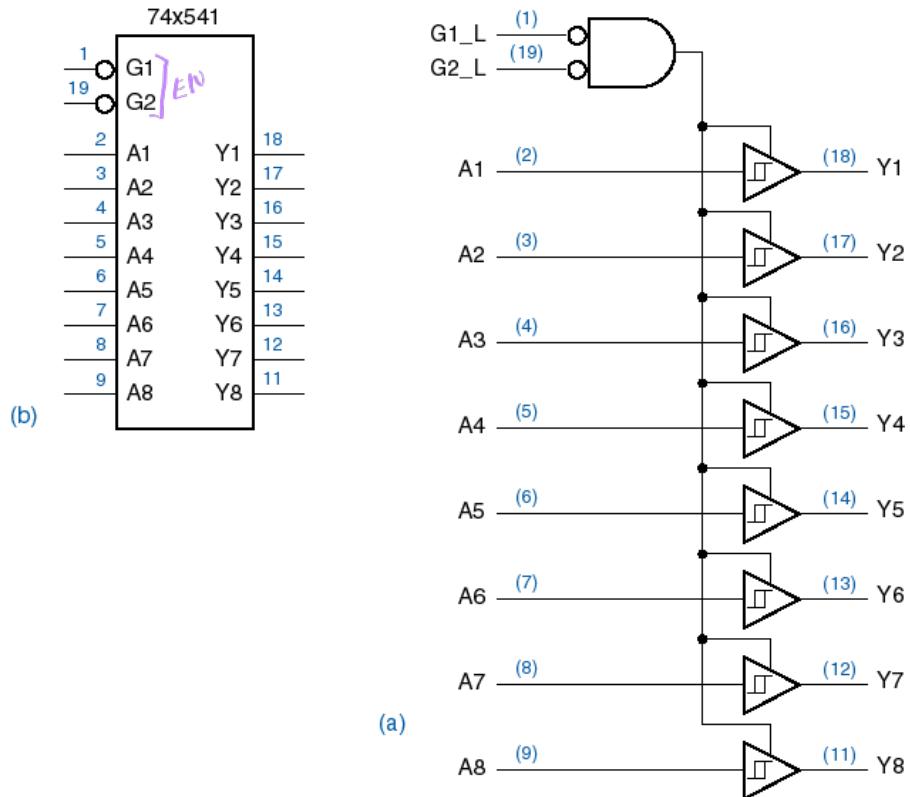
Exercise

- Write the truth table of the circuit and verify that's a 4:1 mux

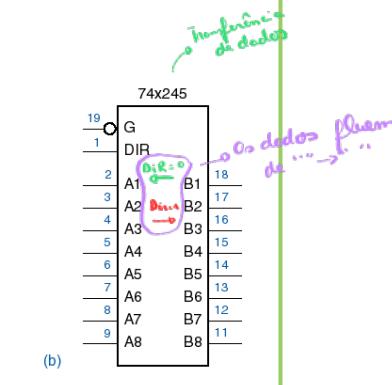
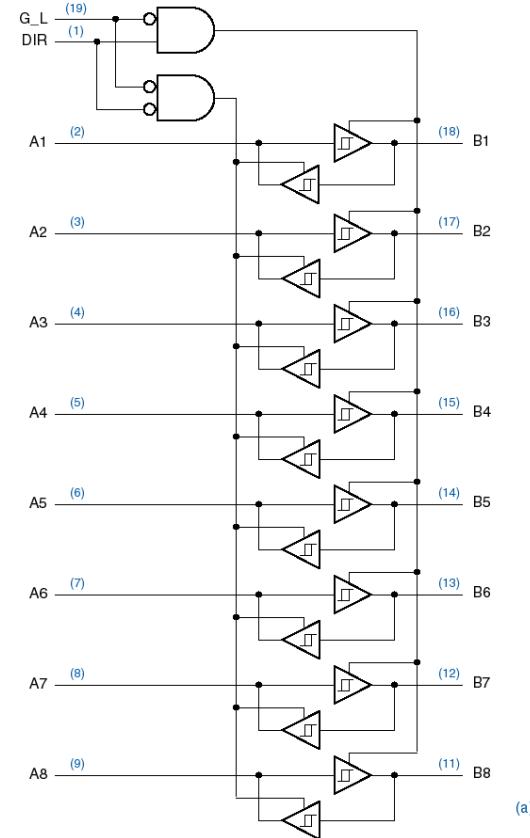


Aggregate 3 State Buffer Models

“BUS” Driver



Transceiver



Aggregate 3 State Buffer Models

74HC244

PIN ASSIGNMENT

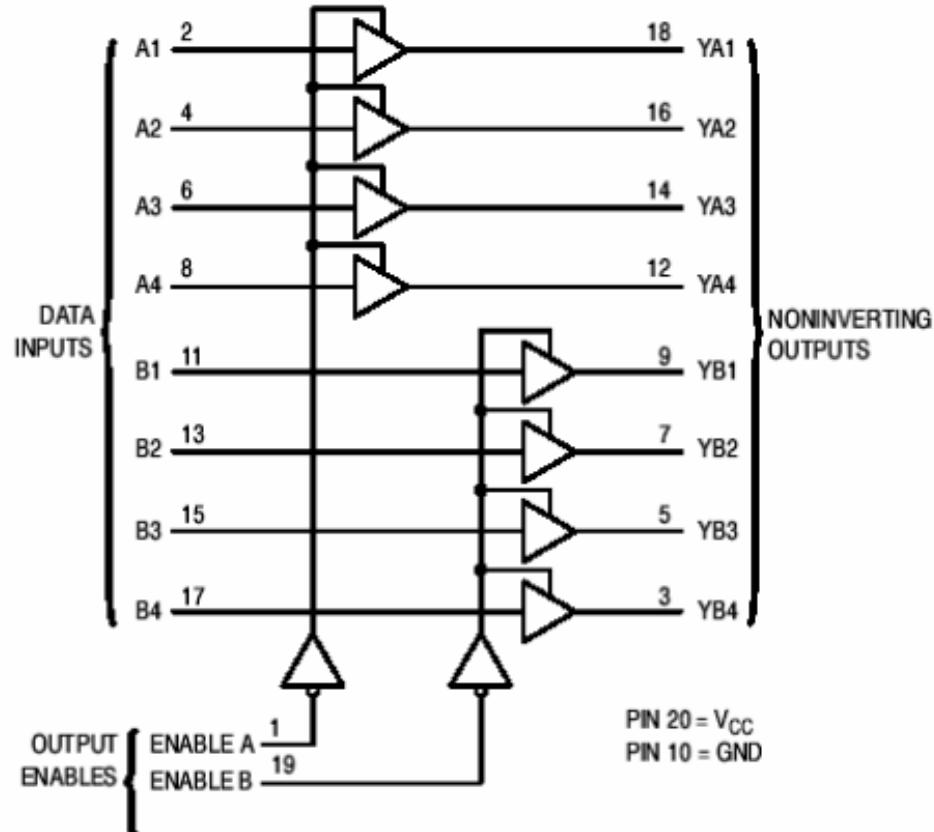
ENABLE A	1	20	V _{CC}
A ₁	2	19	ENABLE B
YB ₄	3	18	YA ₁
A ₂	4	17	B ₄
YB ₃	5	16	YA ₂
A ₃	6	15	B ₃
YB ₂	7	14	YA ₃
A ₄	8	13	B ₂
YB ₁	9	12	YA ₄
GND	10	11	B ₁

FUNCTION TABLE

Inputs		Outputs
Enable A, Enable B	A, B	YA, YB
L	L	L
L	H	H
H	X	Z

Z = high impedance

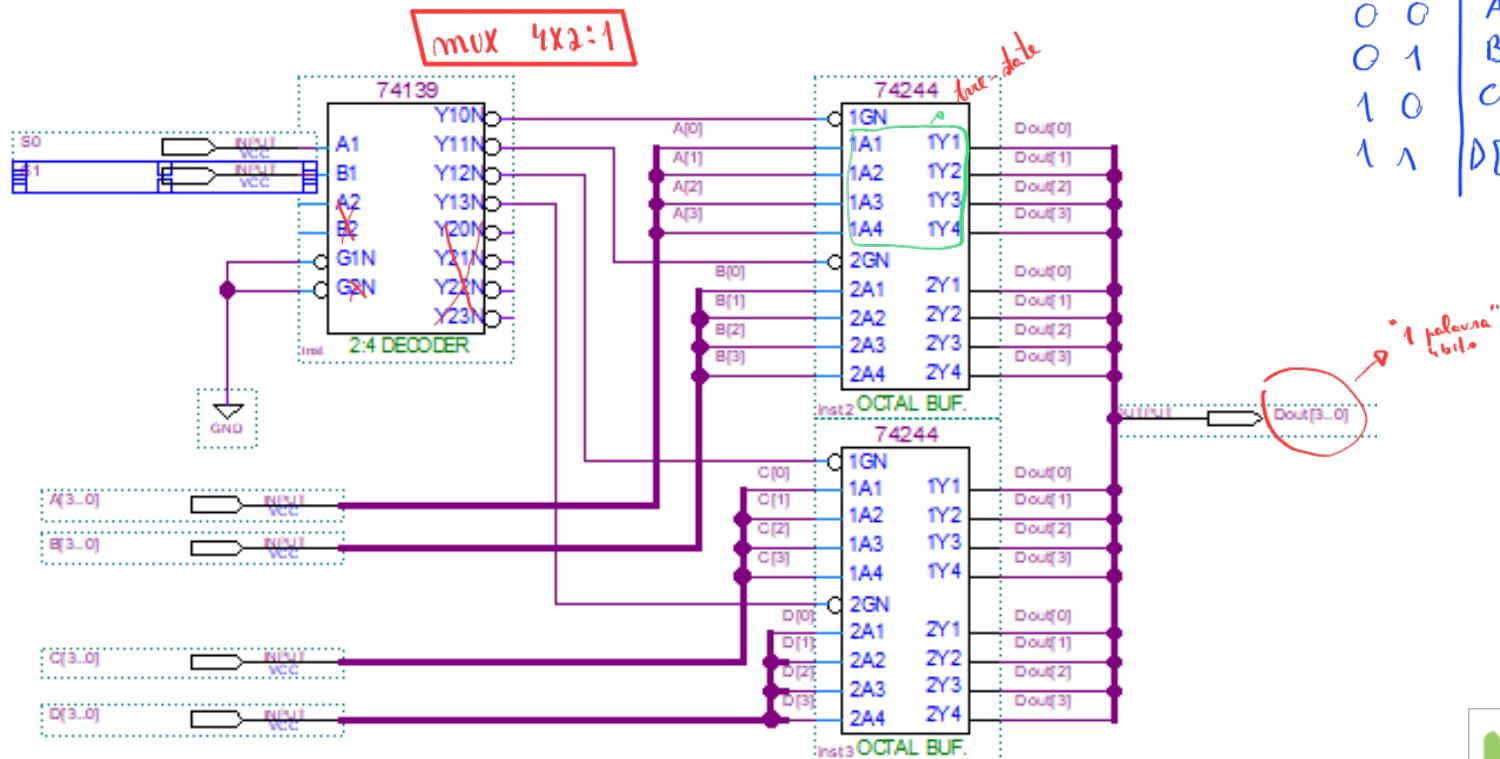
LOGIC DIAGRAM



Word Multiplexing

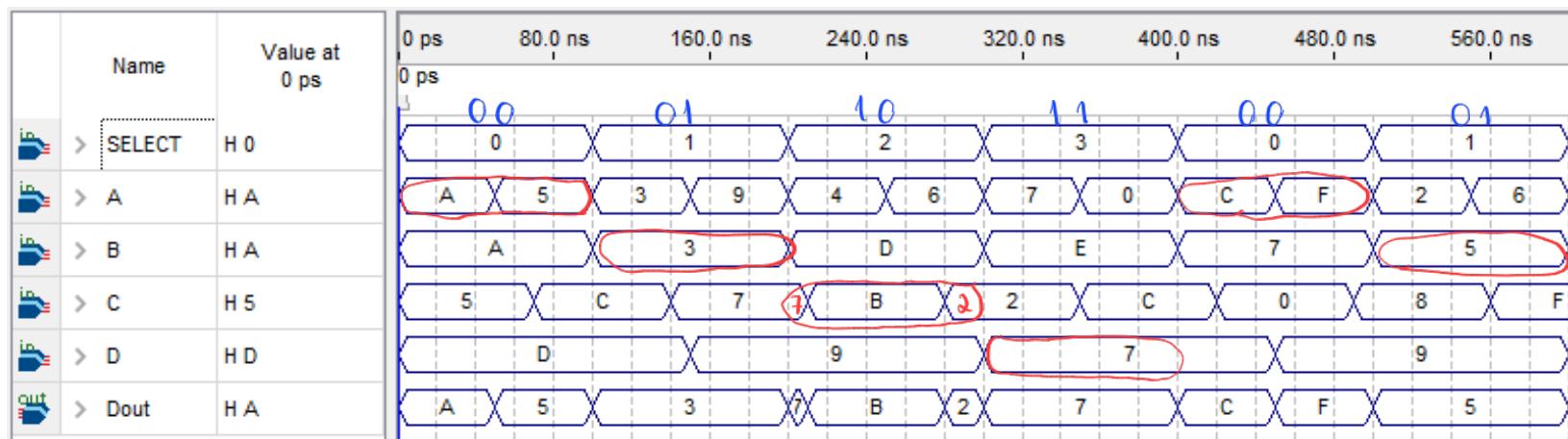
! Importante

- Main idea:
 - Decode the 3 State buffers enabling inputs
 - Share the output data bus



Exercise

- Explain the timing diagram of the previous circuit
 - $\text{SELECT} = (\text{S1}, \text{S0})$



Final Remarks

- Always recall
 - The block symbol
 - The types of inputs and outputs
 - Data
 - Control
 - The truth table
 - The output equations
- Design with encapsulated logic requires mastering all the functional details of each block