Combinational Logic Building Blocks

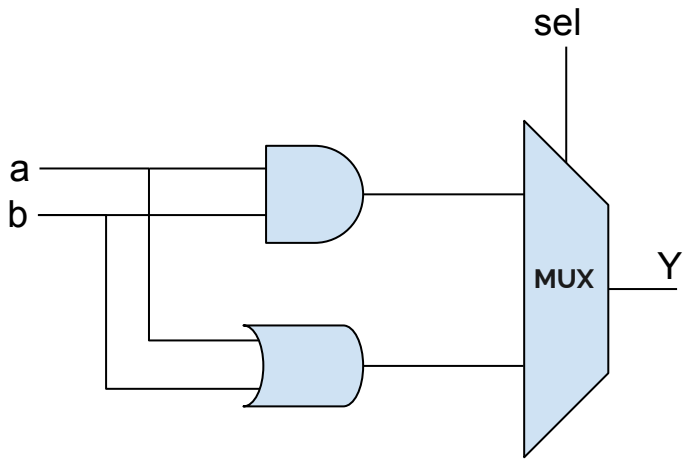
Instructor: Dr. Vinicius Prado da Fonseca (vpradodafons@online.mun.ca)

Combinational Building Blocks

- Combinational logic is often grouped into larger building blocks to build more complex systems.
- This is an application of the principle of abstraction
  - hiding the unnecessary gate-level details to emphasize the function of the building block.
- Two more commonly used building blocks:
  - multiplexers
  - decoders.

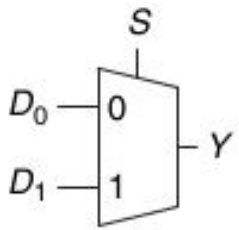
Multiplex (MUX)

- Multiplexers are among the most commonly used combinational circuits.
- They choose an output from among several possible inputs based on the value of a select signal.
- A multiplexer is sometimes affectionately called a mux.



Two to One Multiplexor

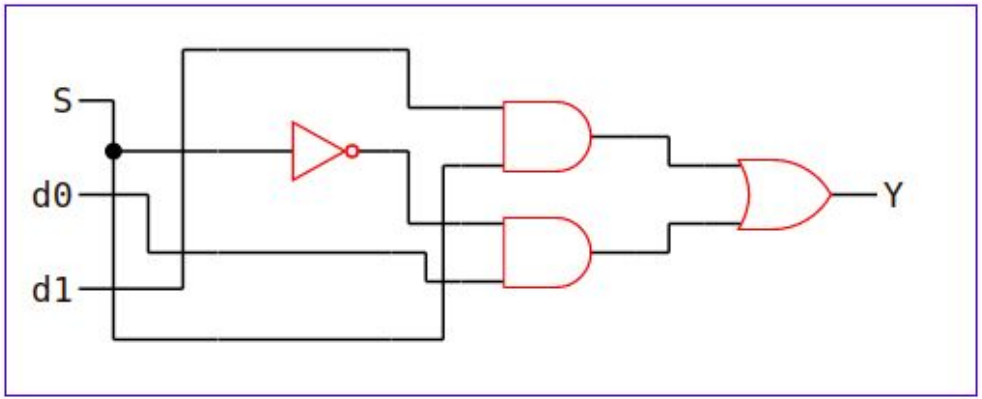
- A two-to-one multiplexor (2-1 or 2:1 MUX) used to combine other blocks
  - inputs D_0 and D_1
  - select input S
  - output Y.
- Similar to an if-then-else.
  - When S = 0, Y = d0
  - When S = 1, Y = d1
- The selection circuit can be implemented with a 1-to-2 decoder
- S is also called a control signal



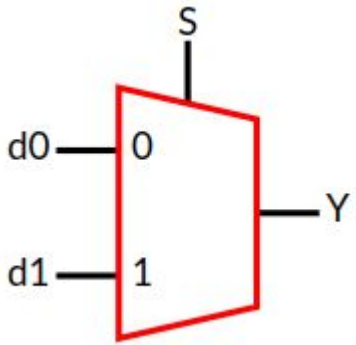
S	D_1	D_0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Two to One Multiplexor

$$Y = \overline{S} \cdot d0 + S \cdot d1$$



S \ d1,d0	00	01	11	10
0	0	1	1	0
1	0	0	1	1



S	d1	d0	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

Two Variables Boolean Function with 4-1 MUX

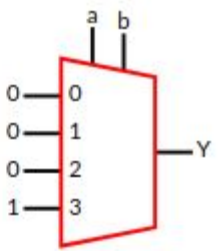
- Any two variable Boolean functions can be implemented with a 4-1 MUX. The idea can be extended to more variables.
- The data inputs to the 4-to-1 MUX are set to the truth table entries

Two Variables Boolean Function with 4-1 MUX

- Any two variable Boolean functions can be implemented with a 4-1 MUX. The idea can be extended to more variables.
- The data inputs to the 4-to-1 MUX are set to the truth table entries

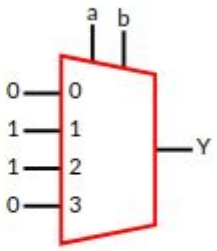
AND

a	b	y
0	0	0
0	1	0
1	0	0
1	1	1



XOR

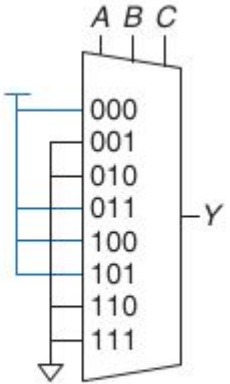
a	b	y
0	0	0
0	1	1
1	0	1
1	1	0



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

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

(a)



(b)

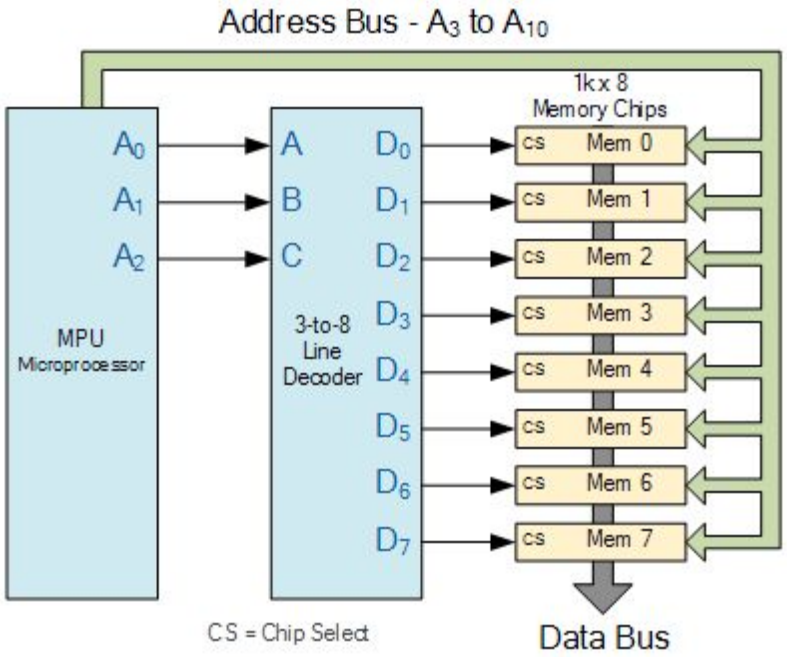
Decoders

- A decoder has N inputs and 2^N outputs
- Asserts (set it to true) exactly one of its outputs depending on the input combination.
- Used to select only one component in a set
    - e.g. Access multiple memory modules

Decoders

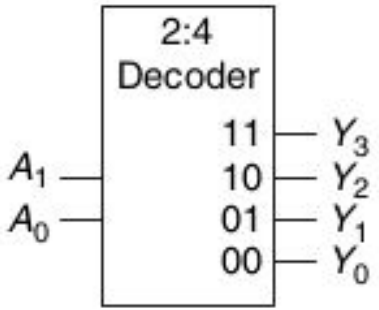
- A decoder has N inputs and 2^N outputs
- Asserts (set it to true) exactly one of its outputs depending on the input combination.
- Used to select only one component in a set
  - e.g. Access multiple memory modules

https://www.electronics-tutorials.ws/combination/comb_5.html



Decoders

- A decoder has N inputs and 2^N outputs
- Asserts (set it to true) exactly one of its outputs depending on the input combination.

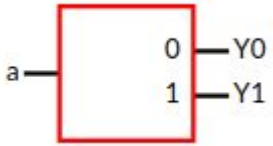


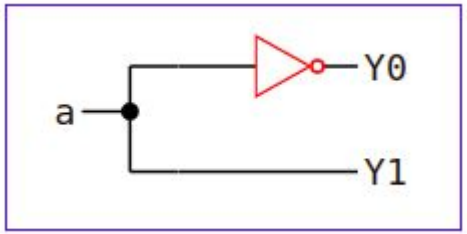
A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

One to Two Decoders

- Decoders are used to select one element out of N elements
- A N input decoder can select up to 2^N elements
- Provides all the minterms for the inputs

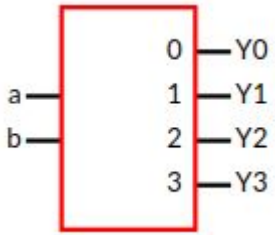
a	Y0	Y1
0	1	0
1	0	1

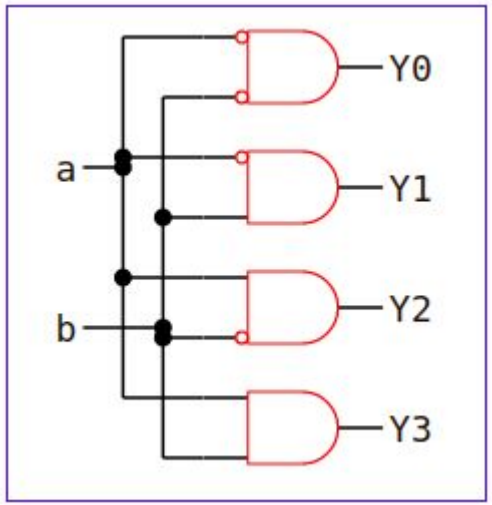




Two to Four Decoders

a	b	Y0	Y1	Y2	Y3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1





Muxes and Decoders

- Larger and multifunctional designs
- Help group other logic blocs
- Implement functions

Questions?

Next: Bitwise operations