Computer Architecture

Keivan Navi

Cal Poly Pomona University

knavi@cpp.edu

Office hours: Tu/Th 5:25 Pm to 6:55 Pm

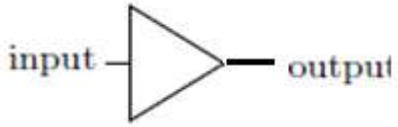
Office: 8-49

Computer Architecture

Logic Gates

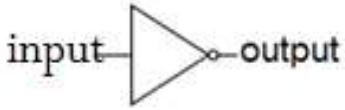
BUFFER, INVERTER (NOT)

- Buffer: Output=Input



input	output
0	0
1	1

- Inverter/Not:



input	output
0	1
1	0

AND, OR

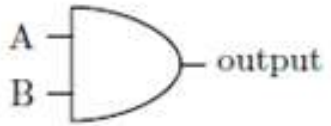
- AND:

1- one input=0 then the output =0

2- all inputs=1s then the output =1

3- The minimum of inputs

A	B	output
0	0	0
0	1	0
1	0	0
1	1	1



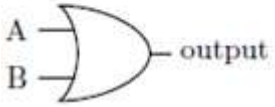
- OR:

1- one input=1 then the output =1

2- all inputs=0s then the output =0

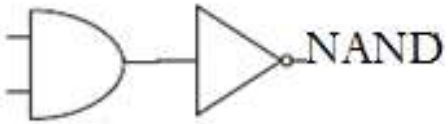
3- The maximum of inputs

A	B	output
0	0	0
0	1	1
1	0	1
1	1	1

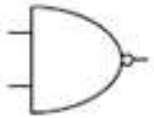


NAND, NOR

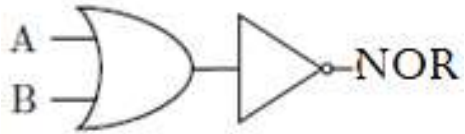
- NAND (Not AND):



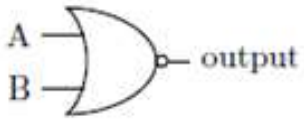
A	B	output
0	0	1
0	1	1
1	0	1
1	1	0



- NOR (Not OR):



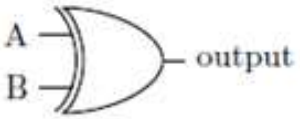
A	B	output
0	0	1
0	1	0
1	0	0
1	1	0



XOR/ XNOR

- Exclusive Or (XOR):
- AB’+A’B

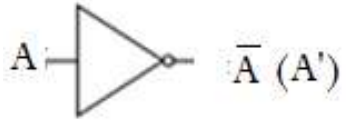
A	B	output
0	0	0
0	1	1
1	0	1
1	1	0

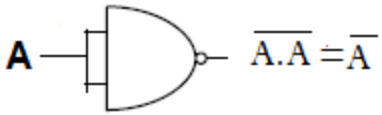


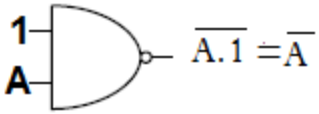
- Exclusive Nor (XNOR):
- A’B’+AB

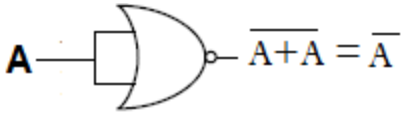
A	B	output
0	0	1
0	1	0
1	0	0
1	1	1

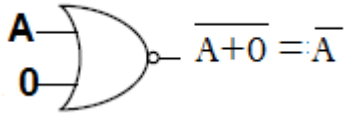
Creating NOT gate using NAND and NOR gates

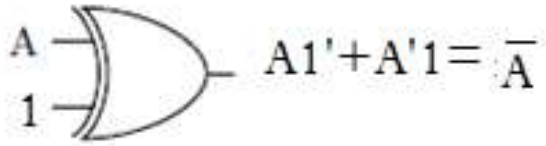


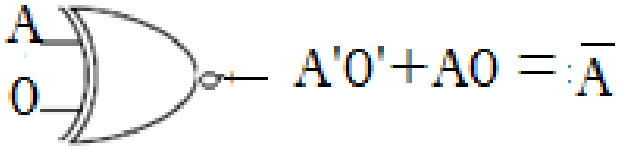












Creating AND, OR NOR using NAND

- To create a NOR, OR, AND, or NOT gate using only NAND gates, simply connect the inputs of a NAND gate in specific ways:
- **NOT gate:**
- Connect both inputs of a NAND gate together to create an inverter (NOT gate).
- **OR gate:**
- Use two NAND gates as inverters (connect each input to itself) to invert the inputs of another NAND gate.
- Feed the inverted inputs to the final NAND gate to get an OR function.
- **AND gate:**
- Invert the output of a NAND gate using another NAND gate configured as an inverter.
- Feed the inverted inputs to the final NAND gate to get an AND function.
- **NOR gate:**
- Use two NAND gates as inverters to invert the inputs.
- Connect the inverted inputs to a single NAND gate to create a NOR function.
- Key points to remember:
- A NAND gate acts like an AND gate with an inverted output.
- To create an inverter (NOT gate) from a NAND gate, simply connect both inputs together.
- By combining NAND gates with their inverted inputs, you can create any other logic gate (OR, AND, NOR).

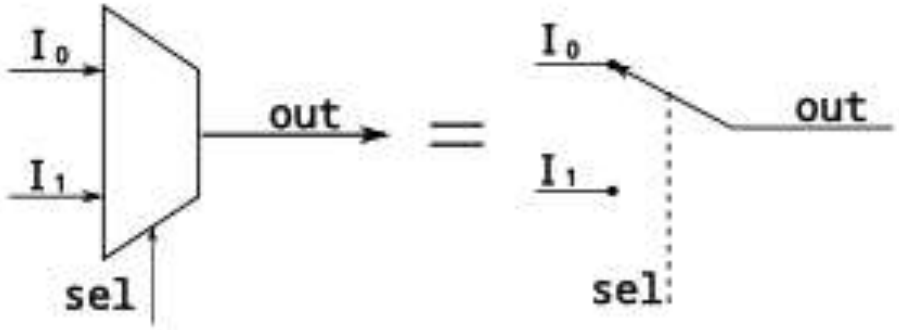
Creating AND, NAND and OR gate using NOR gate

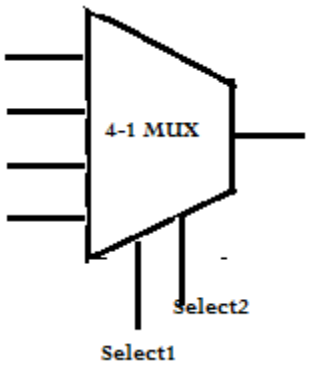
- To create a NAND, OR, and NOT gate using only NOR gates, you can leverage the fact that a NOR gate is essentially an OR gate with an inverted output; by manipulating the inputs and combining NOR gates, you can achieve the functionality of all other basic logic gates:.
- Creating a NOT gate using NOR:
- Connect the input signal to both inputs of a single NOR gate. This effectively inverts the signal, creating a NOT gate.
- Creating an AND gate using NOR:
- Feed the input signals (A and B) to separate NOR gates, where each input is connected to the same signal and the other input is tied to a logic 'high'. Then, connect the outputs of these NOR gates to the inputs of another NOR gate. This will produce the AND function.
- Creating an OR gate using NOR:
- Simply connect the input signals (A and B) directly to a single NOR gate.
- Creating a NAND gate using NOR:
- To create a NAND gate, first create a NOT gate using a single NOR gate as described above. Then, feed the inverted inputs (NOT A and NOT B) into another NOR gate.
- Key points to remember:
- A NOR gate acts like an OR gate with an inverted output.
- To create an inverter (NOT gate) with a NOR gate, connect the input signal to both inputs of the NOR gate.
- By combining NOR gates with properly manipulated inputs, you can construct any other logic gate, including AND and NAND.

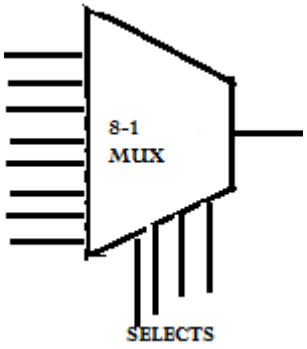
Multiplexer (MUX)

- In electronics, a multiplexer (or mux; spelled sometimes as multiplexor), also known as a data selector, is a device that selects between several analog or digital input signals and forwards the selected input to a single output line. The selection is directed by a separate set of digital inputs known as select lines.
- Digital multiplexers: In digital circuit design, the selector wires are of digital value. In the case of a 2-to-1 multiplexer, a logic value of 0 would connect I 0 to the output while a logic value of 1 would connect I 1 to the output. In larger multiplexers, the number of selector pins is equal to $\lceil \log_2 (n) \rceil$ where n is the number of inputs.

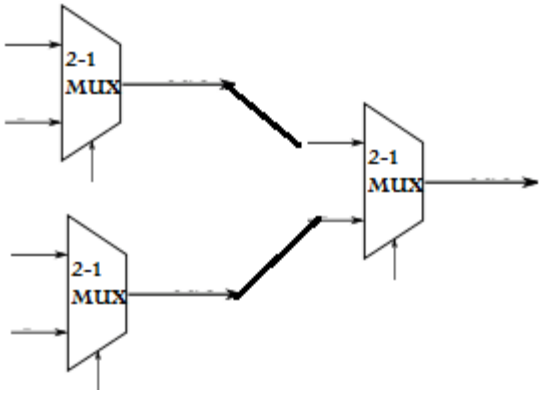
Multiplexer (continued)

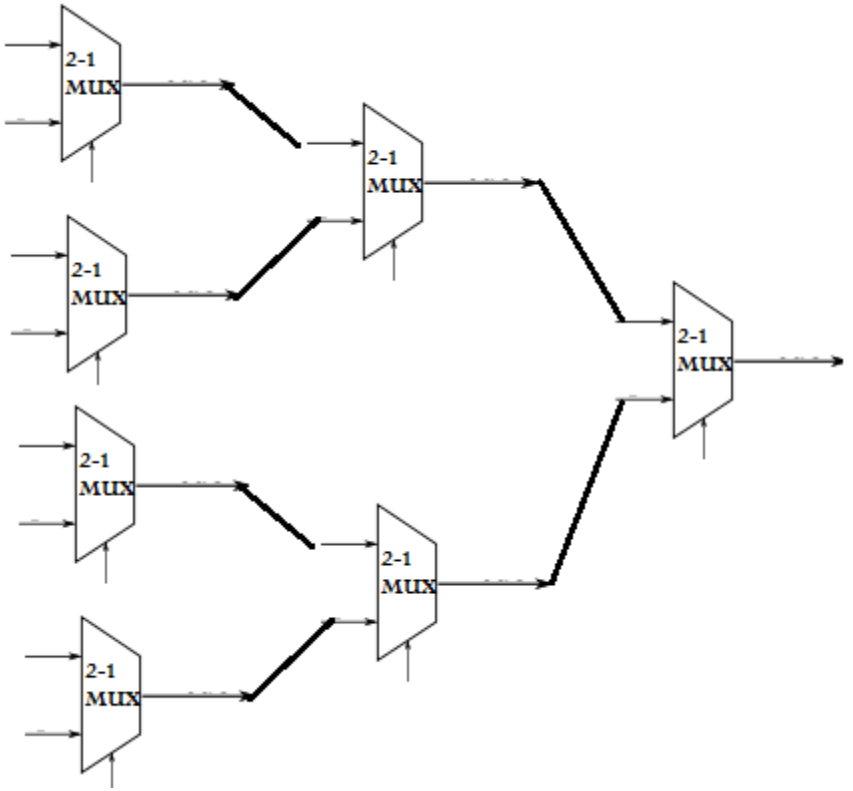






4-1 MUX using 2-1 MUX
8-1 MUX using 2-1 MUX



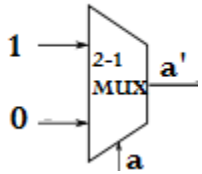


What is a Universal (Complete) gate?

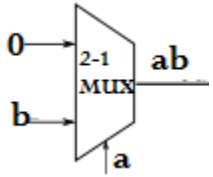
- Universal gate is a logic gate which can implement any Boolean function without the need to use actual gates.
- NAND, NOR Minority Function (a'b'+a'c'+b'c'=$\overline{ab + ac + bc}$) and MUX gate are some example.
- Is a'b a Universal (complete) gate? Why?
- Is a'+b a Universal (complete) gate? Why?
- Is Majority gate (ab+ac+bc) a Universal (Complete) gate?

Creating NOT, AND, NAND, Or, NOR, XOR, XNOR

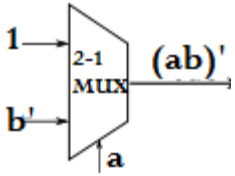
- NOT:



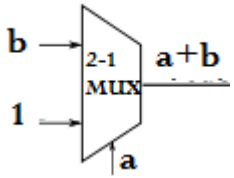
- AND:



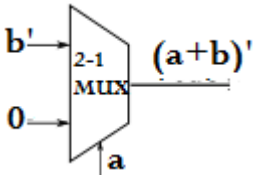
- NAND:



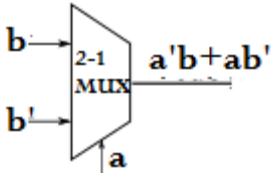
- OR:



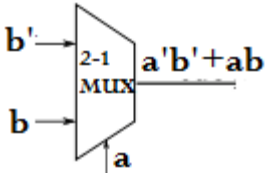
- NOR:



- XOR:

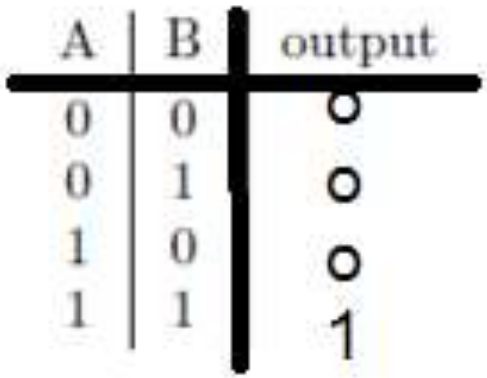


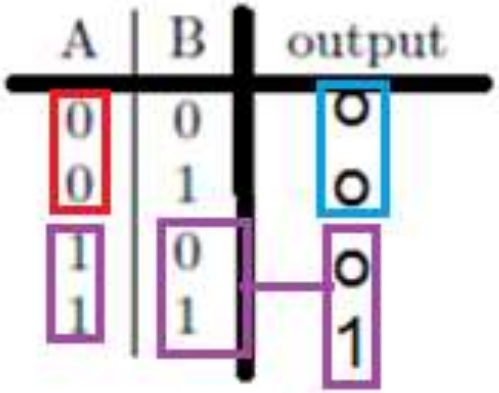
- XNOR:

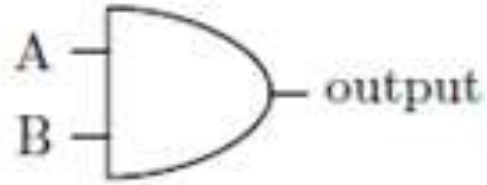


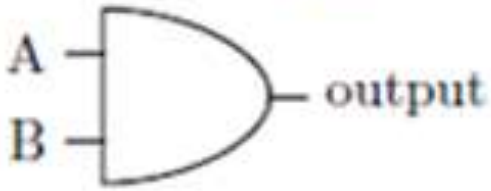
The Algorithm

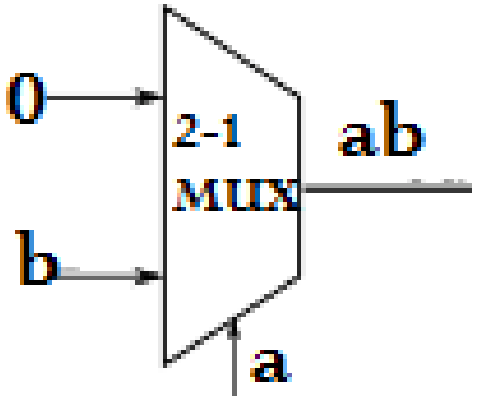
- What is the algorithm? Example: AND gate







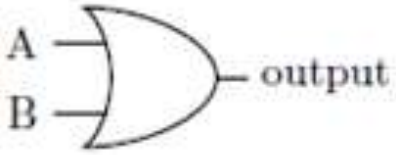




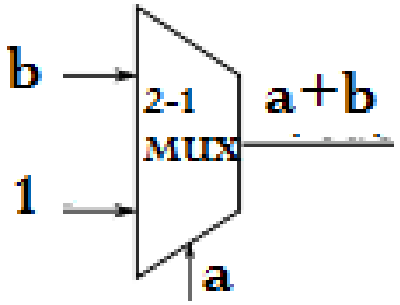
OR gate using MUX

- OR:

A	B	output
0	0	0
0	1	1
1	0	1
1	1	1



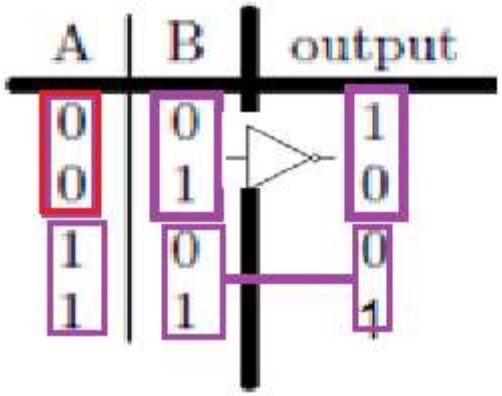
A	B	output
0	0	0
0	1	1
1	0	1
1	1	1

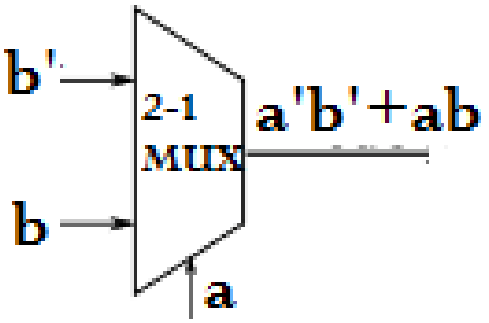


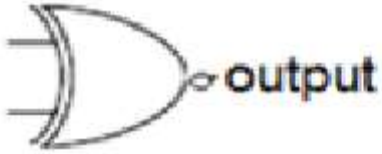
XNOR using MUX

- XNOR:

A	B	output
0	0	1
0	1	0
1	0	0
1	1	1





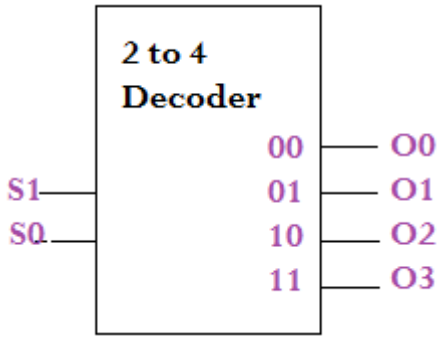


Decoder

- A decoder is a multiple-input, multiple-output logic circuit that converts coded inputs into coded outputs, where the input and output codes are different. The input code generally has fewer bits than the output code, and there is one-to-one mapping from input code words into output code words.
- For example: the 2 binary inputs labeled A and B are decoded into one of 4 outputs, hence the description of 2-to-4 binary decoder. Each output represents one of the minterms of the 2 input variables, (each output = a minterm).

Decoder (continued)

- 5–32 decoder means that only one output will be active out of the 32 output. First step - you need 32 outputs so use 4 3–8 decoders. Each one will use the 3 LSB controls of the 5 input signals. Since they will output 4 active outputs you will need to control the enable pin so only 1 will be enabled.
- The 74HC154; 74HCT154 is a 4-to-16 line decoder. It decodes four binary weighted address inputs (A0 to A3) to sixteen mutually exclusive outputs (Y0 to Y15). The device features two input enable (E0 and E1) inputs. A HIGH on either of the input enables forces the outputs HIGH.



S1	S0	O3	O2	O1	O0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

Is “Decoder” a universal gate?

- No, a decoder is not considered a universal gate; a universal gate refers to a single logic gate like NAND, NOR or Minority gate which can be used to create any other logic function, while a decoder is a combination of multiple gates designed to convert a coded input into a specific output based on the input combination, not capable of performing all logic operations on its own.
- The Decoder will be more profoundly investigated when talking about “Bus Control”.

Is Minority gate a Universal gate?

- Yes, as mentioned before, a minority gate is considered a "universal gate" meaning that it can be used alone to create any logical function, just like a NAND or NOR gate; this is because a minority gate, when combined with an inverter, can produce any logic operation.
- Explanation:
- **Definition of a universal gate:**
- A universal gate is a logic gate that, when used alone, can be combined to create any other logic function.
- **How a minority gate is universal:**
- A minority gate outputs a '1' only if the minority of its inputs are '1', which means that by inverting the output with an inverter, you can essentially create the opposite logic, allowing you to construct any logic circuit using just minority gates and inverters.
- ***Can we prove it***?

How can we prove that a Minority gate is universal gate

- In general if we are able to make either NAND gate or NOR gate it is enough for proving that the related gate is a universal gate.
- Minority gate= $\overline{ab + ac + bc}$
- If c=0 the this gate will be:
- $\overline{ab + a0 + b0}$ = $\overline{ab + 0 + 0}$ = $\overline{ab}$
- And because a NAND is a universal gate we conclude that the Minority gate is a universal gate.