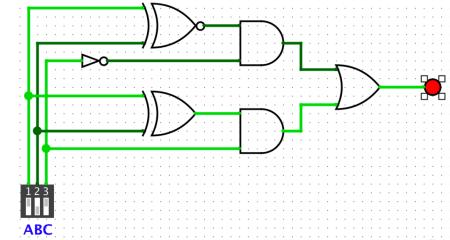


03 –GATES TO CIRCUITS

COMP256 – COMPUTING ABSTRACTIONS
DICKINSON COLLEGE

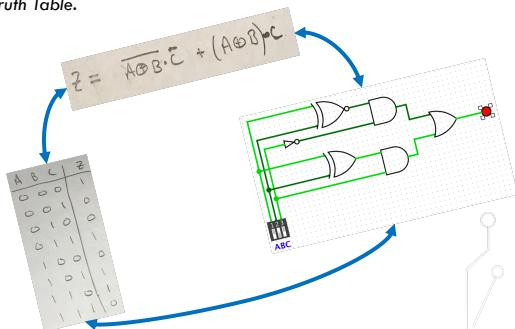
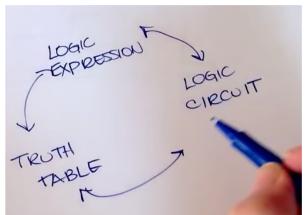
LOGIC CIRCUITS & BOOLEAN FUNCTIONS

- A *Logic Circuit* is a combination of logic gates that perform a computation.
- Logic circuits compute *Boolean Functions*.
 - A Boolean Function is a function that maps Boolean (true/false, 1/0, +5V/ground) input(s) to a Boolean output.

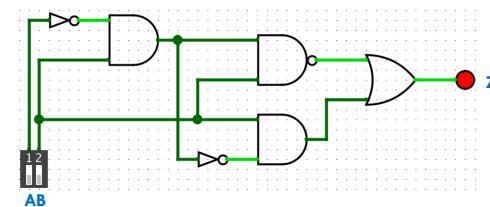


REPRESENTING BOOLEAN FUNCTIONS

- Boolean Functions have three equivalent representations:
 - Logic Expression, Logic Circuit and Truth Table.

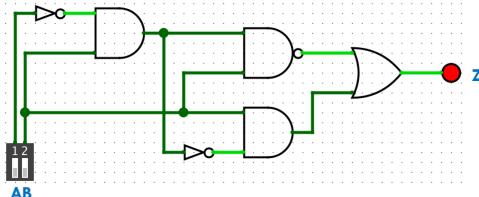


GIVEN A LOGIC CIRCUIT ...



- ... generate a logic expression.
 - Label intermediate gate outputs
 - Write expressions for intermediate gate outputs
 - Use substitution to propagate through to output
- ... generate a truth table.
 - Apply an input pattern
 - Propagate values through gates to output
 - Repeat for all input patterns

GIVEN A LOGIC CIRCUIT ...



- ... generate a logic expression.

$$Z = \overline{A \cdot B} + B \cdot \overline{A \cdot B}$$

- ... generate a truth table.

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

GIVEN A LOGIC EXPRESSION ...

$$Z = \overline{A \cdot B} \cdot (\overline{A} + B)$$

- ... generate a truth table.

- Rows for all possible inputs
- Columns for intermediate values
- Continue until output is reached

- ... generate a logic circuit.

- Create sub-circuits for terms
- Combine sub-circuits with additional gates

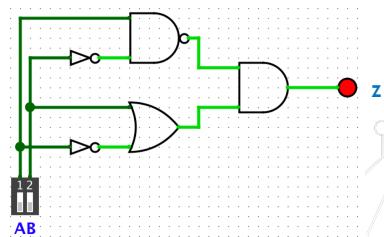
GIVEN A LOGIC EXPRESSION ...

$$Z = \overline{A \cdot B} \cdot (\overline{A} + B)$$

- ... generate a truth table.

A	B	\overline{A}	\overline{B}	$A \cdot \overline{B}$	$\overline{A} \cdot \overline{B}$	$\overline{A} + B$	Z
0	0	1	1	0	1	1	1
0	1	1	0	0	1	1	1
1	0	0	1	1	0	0	0
1	1	0	0	0	1	1	1

- ... generate a logic circuit.



GIVEN A TRUTH TABLE ...

A	B	C	Z
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

- ... generate a logic expression.

- Add one AND term for each 1 in output
- Add negation to make each term true for its inputs.
- Give a Sum-of-Products (SOP) expression.

- ... generate a logic circuit.

- Directly implementation of the SOP expression.

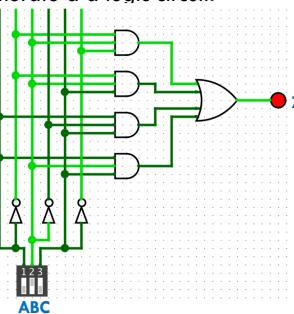
GIVEN A TRUTH TABLE ...

A	B	C	Z
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

- ... generate a logic expression.

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

- ... generate a logic circuit.



LOGIC CIRCUIT DESIGN

- Challenge: Design a circuit that computes an odd parity check on three inputs.

- Logical circuit design process:

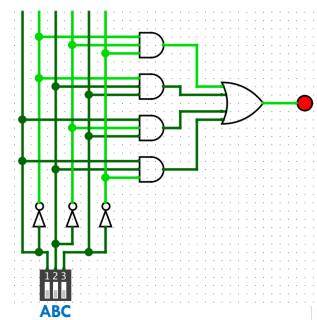
- Given a problem statement generate a truth table.
- Convert the truth table to a logic expression in SOP form
- Implement the SOP logic expression as a logic circuit.

LOGIC CIRCUIT DESIGN

- Challenge: Design a circuit that computes an odd parity check on three inputs.

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

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



ABSTRACTING CIRCUITS

- Logic circuits can be packaged into integrated circuits.

- 9 Bit even/odd parity generator/checker

<https://www.ti.com/lit/ds/sdls152/sdls152.pdf> (for the curious)

