CPSC 121: Models of Computation

Unit 1: Propositional Logic

George Tsiknis

Based on slides by Patrice Belleville and Steve Wolfman

Pre–Lecture Learning Goals

- By the start of the class, you should be able to:
  - Translate back and forth between simple natural language statements and propositional logic.
  - Evaluate the truth of propositional logic statements using truth tables.
  - Translate back and forth between propositional logic statements and circuits that assess the truth of those statements.

Quiz 1 Feedback

-

- We will discuss the open-ended question a bit later.

In-Class Learning Goals

- By the end of this unit, you should be able to:
  - Build computational systems to solve real problems, using both propositional logic expressions and equivalent digital logic circuits.
    - The light switches problem from the 1st online quiz.
    - The 7- or 4-segment LED displays we will discuss in class.
- Building ground work for the Course Big Goals:
  - How do we model computational systems?
  - How do we build devices to compute?

Making a Truth Table

- Note: when you write a truth table, always list the combinations in the same order.
  - For instance, with 3 variables
    - the first column contains 4 false followed by 4 true.
    - the second column contains 2 false, 2 true, 2 false, 2 true.
    - and the third column alternates false with true.
  - With k variables, the first column has 2^{k-1} false and then 2^{k-1} true, the second column has 2^{k-2} false and then 2^{k-2} true (twice), etc.
- Another way is to
  - start with the last column and one variable which will be assigned F and T
  - add one variable at a time duplicating what you have so far and setting the new variable to F for the first copy and T for the second

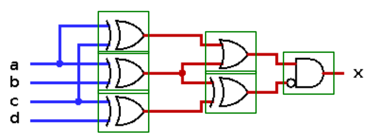
Circuits to Logic Expressions

- How do we find the logical expression that corresponds to a circuit's output?
  - First we write the operator for the gate that produces the circuit's output.
  - The operator's left argument is the expression that corresponds to the circuit for the first input of that gate.
  - The operator's right argument is the expression that corresponds to the circuit for the second input of that gate.
  - Build the logical expression for the left and right argument the same way.

This is our First algorithm!

Example 1

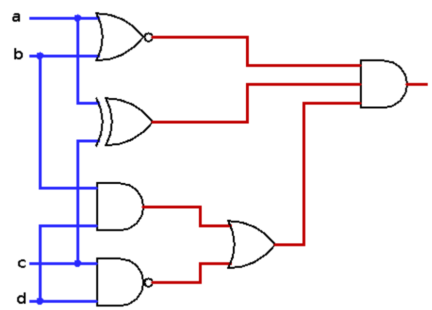
- What does this circuit compute?



((a ⊕ c) v (a ⊕ b)) ^ ~((a ⊕ b) ⊕ (c ⊕ d))

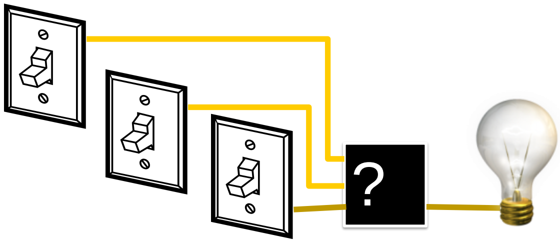
Example 2

- What logical expression corresponds to the following circuit?

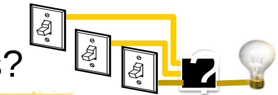


Problem: Three-Switch

- Design a circuit that changes the state of the light whenever any of the switches that control it is flipped.
- Ideally your solution would work with any number!



How do we approach this?



- **Try to understand the "story":**
  - what are the inputs and outputs?
- **Formalize the problem:**
  - "Let a,b,c represent 3 switches from left to right"
- **Solve in propositional logic:**
  - may create a truth table with the inputs and outputs
- **Try a simpler problem:**
  - o start with 1 switch; then try 2; then try 3 switches.
  - o see if we can generalize to n switches.
- **Test your answer:**
  - try some cases, or check some properties

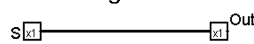
One Switch

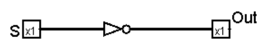
- Identifying inputs/outputs: consider these:
  - **Input$_1$**: the switch flipped
  - **Input$_2$**: the switch is on
  - **Output$_1$**: the light is on
  - **Output$_2$**: the light changed state
- Which are most useful for this problem?
  - a. **Input$_1$** and **Output$_1$**
  - b. **Input$_1$** and **Output$_2$**
  - c. **Input$_2$** and **Output$_1$**
  - d. **Input$_2$** and **Output$_2$**
  - e. None of these

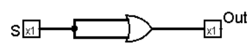
One Switch

- Let's use: S = switch is on
  out = light is on
- Truth table:

- Which of the following circuits solves the problem?

1. 

2. 

3. 

Two Switches

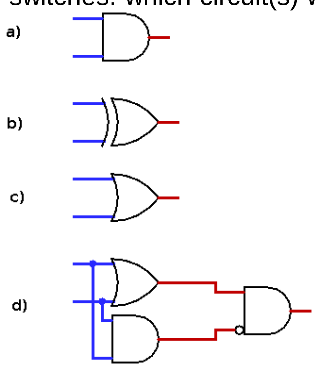
- Make sure we understand the problem first.
- Is the light on or off when both switches are “on”?
  - A. On in every correct solution.
  - B. Off in every correct solution.
  - C. Depends, but a correct solution should always do the same thing with the same settings for the switches..
  - D. Depends, and a correct solution might do different things at different times with the same settings for the switches.
  - E. Neither on nor off.

Two Switches

- Circuit design tip: if you are not sure where to start while designing a circuit,
  - First build the truth table.
  - Then turn it into a circuit.
- For the two switches problem:
  - We can decide arbitrarily what the output is when all switches are OFF.
  - This determines the output for all other cases!
  - Let's see how...
- Truth table (suppose light OFF when all switches are OFF):

Two Switches

- Two switches: which circuit(s) work(s) ?

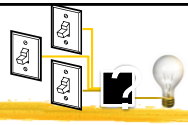


Three Switches

- Fill in the circuit’s truth table:
- Which output column(s) is(are) correct ?

			A	B	C	D	E
s_1	s_2	s_3	out	out	out	out	None of them
T	T	T	T	F	F	T	
T	T	F	F	T	T	F	
T	F	T	F	T	F	T	
T	F	F	T	F	T	F	
F	T	T	F	T	F	T	
F	T	F	T	F	T	F	
F	F	T	T	F	F	T	
F	F	F	F	T	T	F	

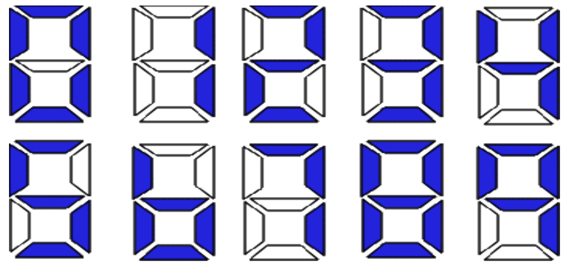
Three Switches



- Suppose we decided to have the light OFF when all switches are OFF. What pattern do we observe?
  - The light is ON if
- What is the formula for it?
  -
- Now to generalize to n switches...
  - What do you think the answer is?
  -
- How can we convince ourselves that it is correct?
  - Mathematical induction

7-Segment LED Display

- **Problem:** design a circuit that displays the numbers 0 through 9 using seven LEDs (lights) in the shape illustrated below.



7-Segment LED Display

- Understanding the story:
  How many inputs *to our circuit* are there?

a. One
b. Seven
c. Ten
d. Four
e. None of these

7-LED Display

- **Problem:** Design a circuit that displays the numbers 0 through 9 using seven LEDs (lights) in the shape illustrated above.
- **First: what's the circuit's job?**
  -

7-LED Display–Input **Values**

- How many different values (messages) must the circuit understand?
  (This is **different than** "how many inputs are there".)

a. One
b. Seven
c. Ten
d. Four
e. None of these

7-LED Display-Input **Lines**

- How many different "parameters" (wires) carry those messages?
  (Not *quite* parameter like in CPSC 110… more like an input wire).

a. One
b. Seven
c. Ten
d. Four
e. None of these

7-LED Display-Inputs

- How do we represent the inputs?
  - Use ? logical (true/false) values.
  - Each integer represented by 1 specific combination.
  - Could we do this randomly?
    - Yes!
  - But we won't
    - How many of you know about binary representation?
- How many values we can represent with
  - 1 propositional variable :
  - 2 propositional variables :
  - 3 " " :
  - n " " :

7-LED Display-Representing Inputs

- Here's how we will represent the inputs:

#	a	b	c	d
0	F	F	F	F
1	F	F	F	T
2	F	F	T	F
3	F	F	T	T
4	F	T	F	F
5	F	T	F	T
6	F	T	T	F
7	F	T	T	T
8	T	F	F	F
9	T	F	F	T
…				

Notice the order: F's first.

7-LED Display-Outputs

- Understanding Outputs:
- How many outputs are there?
  a. 1
  b. 4
  c. 7
  d. 10
  e. None of the above

Simulate 7-LED Display

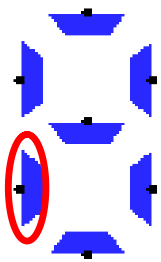
- Let's simulate it with people (raising their hands) ...
- Which other person's algorithm do you need to know about?
  a. No one else's
  b. Your neighbours
  c. The person opposite to you
  d. Everybody else's
  e. None of the above.

Analyzing One Segment

- What's the truth table for the lower-left segment?

#	a	b	c	d	a. out	b. out	c. out	d. out
0	F	F	F	F	F	F	T	0
1	F	F	F	T	T	T	F	2
2	F	F	T	F	F	F	T	6
3	F	F	T	T	T	T	F	8
4	F	T	F	F	F	T	F	4
5	F	T	F	T	T	T	F	7
6	F	T	T	F	F	F	T	9
7	F	T	T	T	T	T	F	3
8	T	F	F	F	F	F	T	1
9	T	F	F	T	T	T	F	5

e. None of these.



Analyzing One Segment

- From the truth table, we can make an expression for each **true** row and **OR** them together.

#	a	b	c	d	out
0	F	F	F	F	T
1	F	F	F	T	F
2	F	F	T	F	T
3	F	F	T	T	F
4	F	T	F	F	F
5	F	T	F	T	F
6	F	T	T	F	T
7	F	T	T	T	F
8	T	F	F	F	T
9	T	F	F	T	F

Which logical statement is true only in this row?

a. $\sim a \vee \sim b \vee c \vee \sim d$
b. $a \wedge b \vee c \wedge d$
c. $\sim a \wedge \sim b \wedge c \wedge \sim d$
d. $a \wedge b \wedge \sim c \wedge d$
e. None of these

Analyzing One Segment

- Let' s complete the expression for the lower-left segment

#	a	b	c	d	out
0	F	F	F	F	T
1	F	F	F	T	F
2	F	F	T	F	T
3	F	F	T	T	F
4	F	T	F	F	F
5	F	T	F	T	F
6	F	T	T	F	T
7	F	T	T	T	F
8	T	F	F	F	T
9	T	F	F	T	F

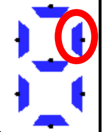
( ) ∨

( ) ∨

( ) ∨

( )

Alternative to Table with Many Ts



- We can use the F rows and negate the statement!

#	a	b	c	d	out
0	F	F	F	F	T
1	F	F	F	T	T
2	F	F	T	F	T
3	F	F	T	T	T
4	F	T	F	F	T
5	F	T	F	T	F
6	F	T	T	F	F
7	F	T	T	T	T
8	T	F	F	F	T
9	T	F	F	T	T

Which of these correctly models this LED?

a. $\sim(\sim a \wedge b \wedge \sim c \wedge d) \vee \sim(\sim a \wedge b \wedge c \wedge \sim d)$

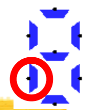
b. $\sim(a \wedge \sim b \wedge c \wedge \sim d) \vee \sim(a \wedge \sim b \wedge \sim c \wedge d)$

c. $\sim[(\sim a \wedge b \wedge \sim c \wedge d) \vee (\sim a \wedge b \wedge c \wedge \sim d)]$

d. $\sim[(a \wedge \sim b \wedge c \wedge \sim d) \vee (a \wedge \sim b \wedge \sim c \wedge d)]$

e. None of these

Another Way



#	a	b	c	d	out
0	F	F	F	F	T
1	F	F	F	T	F
2	F	F	T	F	T
3	F	F	T	T	F
4	F	T	F	F	F
5	F	T	F	T	F
6	F	T	T	F	T
7	F	T	T	T	F
8	T	F	F	F	T
9	T	F	F	T	F

- Looking back at the bottom-left segment:
  - There may be a simpler proposition, which will translate into a smaller circuit.
  - Can you find a pattern in the rows for which the segment should be “on”?

What is coming up?

- Second online quiz is due on **WEDNESDAY, SEPT 10, 7:00pm**.
- Assigned reading for the quiz:
  - Epp, 4th edition: 2.2
  - Epp, 3rd edition: 1.2
  - Rosen, 6th edition: 1.1 from page 6 onwards.

Exercises:

- Prove that our two solutions for the lower-left segment are not logically equivalent.
  - You should do this by providing values for the variables, so the two propositions have different truth values.
  - Why are they both correct solutions, despite that?
- Finish the problem by building circuits for the other 5 segments.
- Design a circuit that takes three bits as input, and outputs the binary representation for their sum.