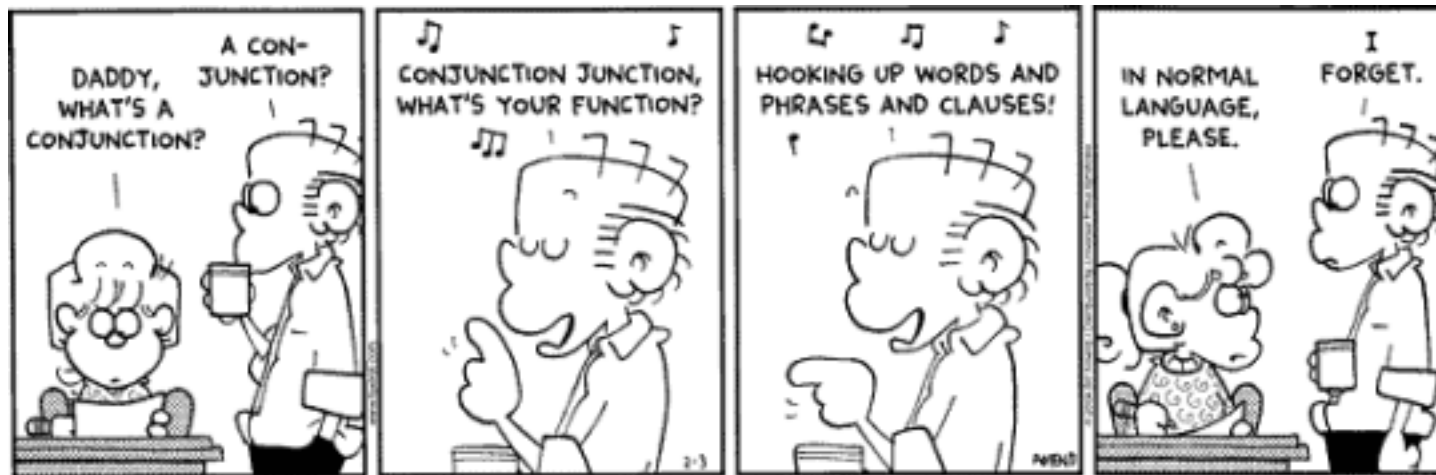


CSE 311: Foundations of Computing

Lecture 2: More Logic, Equivalence & Digital Circuits



- Sections tomorrow as usual "text cards"
- Ed discussion board activated.
 - as of last night 11 students had not yet accepted invitation...
- Gradescope - did you get email?

If you are worried about Mathy aspects of 311

- Associated 1-credit CR/NC workshop
 - CSE 390Z
 - Extra collaborative practice on 311 concepts, study skills, a small amount of assigned work
 - Meets in Loew 113
 - ZA Section Thursdays 3:30-4:50 pm
 - If sufficient demand will add a ZB Section Thursdays 5:00-6:20
 - Full participation is required for credit
 - NOT for help with 311 homework
- Anyone in 311 can sign up but enrollment is limited
 - Enrollment in CSE 390Z section ZA will open up later today FCFS
 - If you want to register but it is full, show up anyway at 3:30 tomorrow in Loew 113.

Last class: Some Connectives & Truth Tables

Negation (not)

p	$\neg p$
T	F
F	T

Conjunction (and)

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction (or)

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

Exclusive Or

p	q	$p \oplus q$
T	T	F
T	F	T
F	T	T
F	F	F

XOR

Last class: Implication

“If it’s raining, then I have my umbrella”

It’s useful to think of implications as promises. That is “Did I lie?”

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

	It’s raining	It’s not raining
I have my umbrella	No	No
I do not have my umbrella	Yes	No

The only lie is when:

(a) It’s raining AND

(b) I don’t have my umbrella

Last class: $p \rightarrow q$

Implication:

- p implies q
- whenever p is true q must be true
- if p then q
- q if p
- p is sufficient for q
- p only if q
- q is necessary for p

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Last class: Biconditional: $p \leftrightarrow q$

- p iff q
- p is equivalent to q
- p implies q and q implies p
- p is necessary and sufficient for q

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

Last class: Garfield Sentence with a Truth Table

p	q	r	$\neg r$	$q \vee \neg r$	$q \wedge r$	$(q \wedge r) \rightarrow p$	$((q \wedge r) \rightarrow p) \wedge (q \vee \neg r)$
F	F	F	T	T	F	T	T
F	F	T	F	F	F	T	F
F	T	F	T	T	F	T	T
F	T	T	F	T	T	F	F
T	F	F	T	T	F	T	T
T	F	T	F	F	F	T	F
T	T	F	T	T	F	T	T
T	T	T	F	T	T	T	T

Converse, Contrapositive

Implication:

$$p \rightarrow q$$

Converse:

$$q \rightarrow p$$

Contrapositive:

$$\neg q \rightarrow \neg p$$

Inverse:

$$\neg p \rightarrow \neg q$$

Consider

p : x is divisible by 2

q : x is divisible by 4

$p \rightarrow q$	
$q \rightarrow p$	
$\neg q \rightarrow \neg p$	
$\neg p \rightarrow \neg q$	

Converse, Contrapositive

Implication:

$$p \rightarrow q$$

Converse:

$$q \rightarrow p$$

Contrapositive:

$$\neg q \rightarrow \neg p$$

Inverse:

$$\neg p \rightarrow \neg q$$

Consider

p : x is divisible by 2

q : x is divisible by 4

$p \rightarrow q$	F
$q \rightarrow p$	T
$\neg q \rightarrow \neg p$	F
$\neg p \rightarrow \neg q$	T

Numbers that are...

	Divisible By 2	Not Divisible By 2
Divisible By 4	4, 8, 12, ...	X
Not Divisible By 4	2, 6, 10, 14, ...	1, 3, 5, 7, ...

Converse, Contrapositive

Implication:

$$p \rightarrow q$$

Converse:

$$q \rightarrow p$$

Contrapositive:

$$\neg q \rightarrow \neg p$$

Inverse:

$$\neg p \rightarrow \neg q$$

Consider

p : x is divisible by 2

q : x is divisible by 4

$p \rightarrow q$	F
$q \rightarrow p$	T
$\neg q \rightarrow \neg p$	F
$\neg p \rightarrow \neg q$	T

Numbers that are...

	p Divisible By 2	$\neg p$ Not Divisible By 2
q Divisible By 4	4,8,12,...	Impossible
$\neg q$ Not Divisible By 4	2,6,10,...	1,3,5,...

Converse, Contrapositive

Implication:

$$p \rightarrow q$$

Converse:

$$q \rightarrow p$$

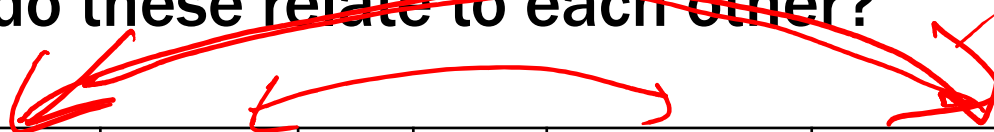
Contrapositive:

$$\neg q \rightarrow \neg p$$

Inverse:

$$\neg p \rightarrow \neg q$$

How do these relate to each other?



p	q	$p \rightarrow q$	$q \rightarrow p$	$\neg p$	$\neg q$	$\neg p \rightarrow \neg q$	$\neg q \rightarrow \neg p$
T	T	T	T	F	F	T	T
T	F	F	T	F	T	F	F
F	T	T	F	T	F	F	T
F	F	T	T	T	T	T	T

Converse, Contrapositive

Implication:

$$p \rightarrow q$$

Converse:

$$q \rightarrow p$$

Contrapositive:

$$\neg q \rightarrow \neg p$$

Inverse:

$$\neg p \rightarrow \neg q$$

An **implication** and its **contrapositive**
have the same truth value!

p	q	$p \rightarrow q$	$q \rightarrow p$	$\neg p$	$\neg q$	$\neg p \rightarrow \neg q$	$\neg q \rightarrow \neg p$
T	T	T	T	F	F	T	T
T	F	F	T	F	T	T	F
F	T	T	F	T	F	F	T
F	F	T	T	T	T	T	T

Tautologies!

Terminology: A compound proposition is a...

- *Tautology* if it is always true
- *Contradiction* if it is always false
- *Contingency* if it can be either true or false

$$p \vee \neg p$$

Tautology

$$p \oplus p$$

Contradiction

$$(p \rightarrow q) \wedge p$$

Contingency

Tautologies!

Terminology: A compound proposition is a...

- *Tautology* if it is always true
- *Contradiction* if it is always false
- *Contingency* if it can be either true or false

$$p \vee \neg p$$

This is a tautology. It's called the "law of the excluded middle".
If p is true, then $p \vee \neg p$ is true. If p is false, then $p \vee \neg p$ is true.

$$p \oplus p$$

This is a contradiction. It's always false no matter what truth value p takes on.

$$(p \rightarrow q) \wedge p$$

This is a contingency. When $p=T, q=T$, $(T \rightarrow T) \wedge T$ is true.
When $p=T, q=F$, $(T \rightarrow F) \wedge T$ is false.

Logical Equivalence

A = B means **A** and **B** are identical “strings”:

$$- p \wedge q = p \wedge q$$

$$- p \wedge q \neq q \wedge p$$

Logical Equivalence

A = B means **A** and **B** are identical “strings”:

- $p \wedge q = p \wedge q$

These are equal, because they are character-for-character identical.

- $p \wedge q \neq q \wedge p$

These are NOT equal, because they are different sequences of characters. They “mean” the same thing though.

A ≡ B means **A** and **B** have identical truth values:

- $p \wedge q \equiv p \wedge q$

- $p \wedge q \equiv q \wedge p$

- $p \wedge q \neq q \vee p$

eg. p is T and q is F

Logical Equivalence

A = B means **A** and **B** are identical “strings”:

- $p \wedge q = p \wedge q$

These are equal, because they are character-for-character identical.

- $p \wedge q \neq q \wedge p$

These are NOT equal, because they are different sequences of characters. They “mean” the same thing though.

A ≡ B means **A** and **B** have identical truth values:

- $p \wedge q \equiv p \wedge q$

Two formulas that are equal also are equivalent.

- $p \wedge q \equiv q \wedge p$

These two formulas have the same truth table!

- $p \wedge q \neq q \vee p$

When $p=T$ and $q=F$, $p \wedge q$ is false, but $p \vee q$ is true!

$A \leftrightarrow B$ vs. $A \equiv B$

$A \equiv B$ is an *assertion over all possible truth values* that A and B always have the same truth values.

$A \leftrightarrow B$ is a *proposition* that may be true or false depending on the truth values of the variables in A and B .

$A \equiv B$ and $\underbrace{(A \leftrightarrow B)} \equiv T$ have the same meaning.

A	B
T	T
T	F
F	T
F	F

$A \leftrightarrow B$
T
F
F
T

De Morgan's Laws

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

Negate the statement:

not ("My code compiles or there is a bug.")

To negate the statement,

ask "when is the original statement false".

not (my code compiles) \wedge not (There is a bug)

My code doesn't compile and This isn't a bug

De Morgan's Laws

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

Negate the statement:

“My code compiles or there is a bug.”

To negate the statement,

ask “when is the original statement false”.

It's false when not(my code compiles) AND not(there is a bug).

Translating back into English, we get:

My code doesn't compile and there is not a bug.

De Morgan's Laws

Example: $\neg(p \wedge q) \equiv (\neg p \vee \neg q)$ ✓

p	q	$\neg p$	$\neg q$	$\neg p \vee \neg q$	$p \wedge q$	$\neg(p \wedge q)$	$\neg(p \wedge q) \leftrightarrow (\neg p \vee \neg q)$
T	T	F	F	F	T	F	T
T	F	F	T	T	F	T	T
F	T	T	F	T	F	T	T
F	F	T	T	T	F	T	T

De Morgan's Laws

Example: $\neg(p \wedge q) \equiv (\neg p \vee \neg q)$

p	q	$\neg p$	$\neg q$	$\neg p \vee \neg q$	$p \wedge q$	$\neg(p \wedge q)$	$\neg(p \wedge q) \leftrightarrow (\neg p \vee \neg q)$
T	T	F	F	F	T	F	T
T	F	F	T	T	F	T	T
F	T	T	F	T	F	T	T
F	F	T	T	T	F	T	T

De Morgan's Laws

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

```
if (! (front != null && value > front.data))  
    front = new ListNode(value, front);  
else {  
    ListNode current = front;  
    while (current.next != null && current.next.data < value))  
        current = current.next;  
    current.next = new ListNode(value, current.next);  
}
```

De Morgan's Laws

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

`!(front != null && value > front.data)`

\equiv

`front == null || value <= front.data`

You've been using these for a while!

Law of Implication

$$p \rightarrow q \equiv \neg p \vee q$$

p	q	$p \rightarrow q$	$\neg p$	$\neg p \vee q$	$(p \rightarrow q) \leftrightarrow (\neg p \vee q)$
T	T	T	F	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

Law of Implication

$$p \rightarrow q \equiv \neg p \vee q$$

p	q	$p \rightarrow q$	$\neg p$	$\neg p \vee q$	$p \rightarrow q \leftrightarrow \neg p \vee q$
T	T	T	F	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

Some Equivalences Related to Implication

$$\textcircled{\theta} \quad p \rightarrow q \quad \equiv \quad \neg p \vee q$$

$$\textcircled{e} \quad p \rightarrow q \quad \equiv \quad \neg q \rightarrow \neg p$$

$$\rightarrow \quad p \leftrightarrow q \quad \equiv \quad (p \rightarrow q) \wedge (q \rightarrow p)$$

$$p \leftrightarrow q \quad \equiv \quad \neg p \leftrightarrow \neg q$$

*Implication
Law
Contrapositive*

Properties of Logical Connectives

We will always give
you this list!

- **Identity**

- $p \wedge T \equiv p$
- $p \vee F \equiv p$

- **Domination**

- $p \vee T \equiv T$
- $p \wedge F \equiv F$

- **Idempotent**

- $p \vee p \equiv p$
- $p \wedge p \equiv p$

- **Commutative**

- $p \vee q \equiv q \vee p$
- $p \wedge q \equiv q \wedge p$

- **Associative**

- $(p \vee q) \vee r \equiv p \vee (q \vee r)$
- $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$

- **Distributive**

- $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
- $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$

- **Absorption**

- $p \vee (p \wedge q) \equiv p$
- $p \wedge (p \vee q) \equiv p$

- **Negation**

- $p \vee \neg p \equiv T$
- $p \wedge \neg p \equiv F$

Computing Equivalence

Describe an algorithm for computing if two logical expressions/circuits are equivalent.

What is the run time of the algorithm?

n variables

Compute the entire truth table for both of them!

There are 2^n entries in the column for n variables
 2^n rows of truth table

Understanding Connectives

- **Reflect basic rules of reasoning and logic**
- **Allow manipulation of logical formulas**
 - Simplification
 - Testing for equivalence
- **Applications**
 - Query optimization
 - Search optimization and caching
 - Artificial Intelligence
 - Program verification

Digital Circuits

Computing With Logic

- **T** corresponds to **1** or “high” voltage
- **F** corresponds to **0** or “low” voltage

Gates

- Take inputs and produce outputs (functions)
- Several kinds of gates
- Correspond to propositional connectives (most of them)

And Gate

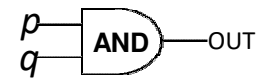
AND Connective

vs.

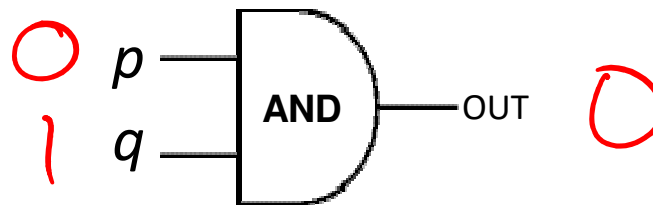
AND Gate

$p \wedge q$

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F



p	q	OUT
1	1	1
1	0	0
0	1	0
0	0	0



“block looks like D of AND”

Or Gate

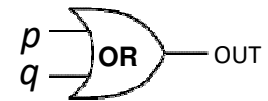
OR Connective

vs.

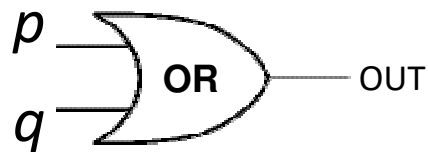
OR Gate

$p \vee q$

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F



p	q	OUT
1	1	1
1	0	1
0	1	1
0	0	0



“arrowhead block looks like V”

Not Gates

NOT Connective

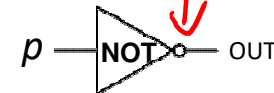
vs.

NOT Gate

most important

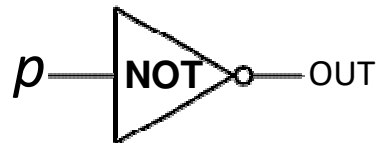
$\neg p$

p	$\neg p$
T	F
F	T



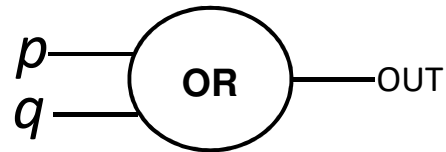
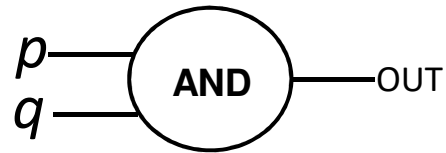
Also called
inverter

p	OUT
1	0
0	1

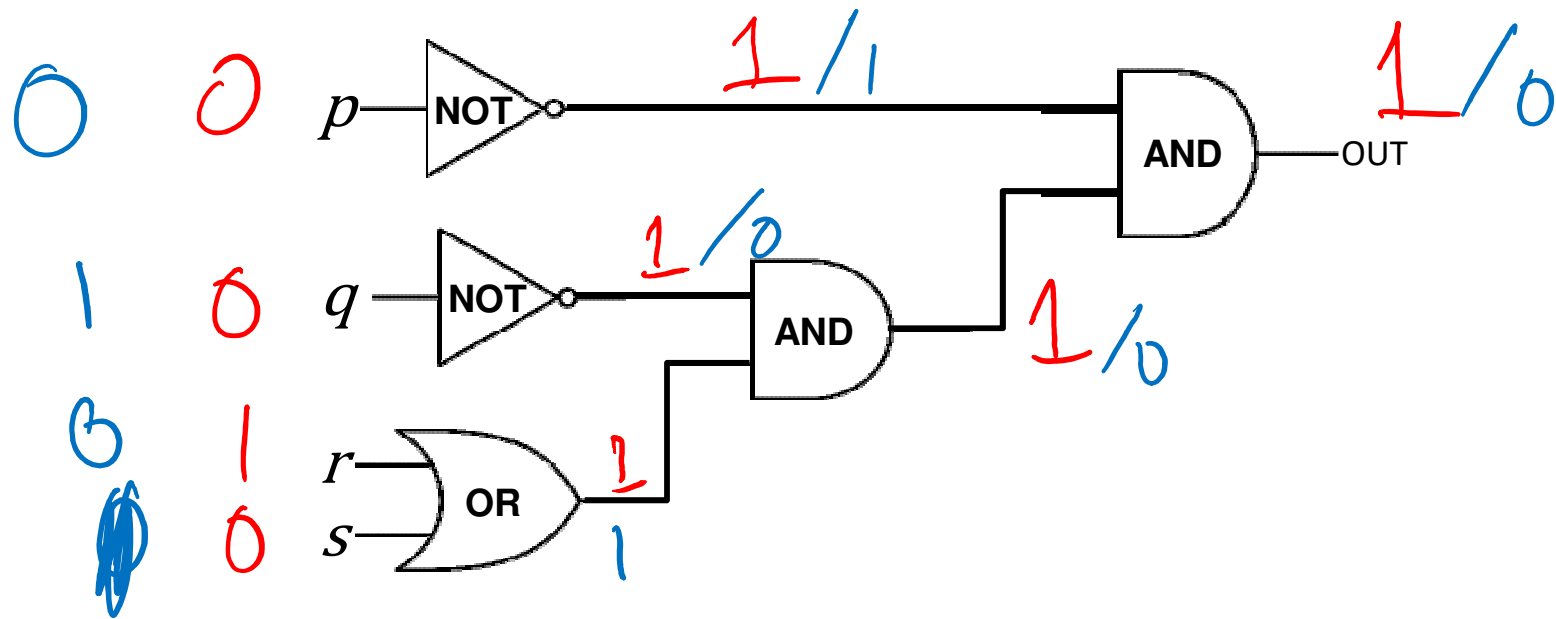


Blobs are Okay!

You may write gates using blobs instead of shapes!

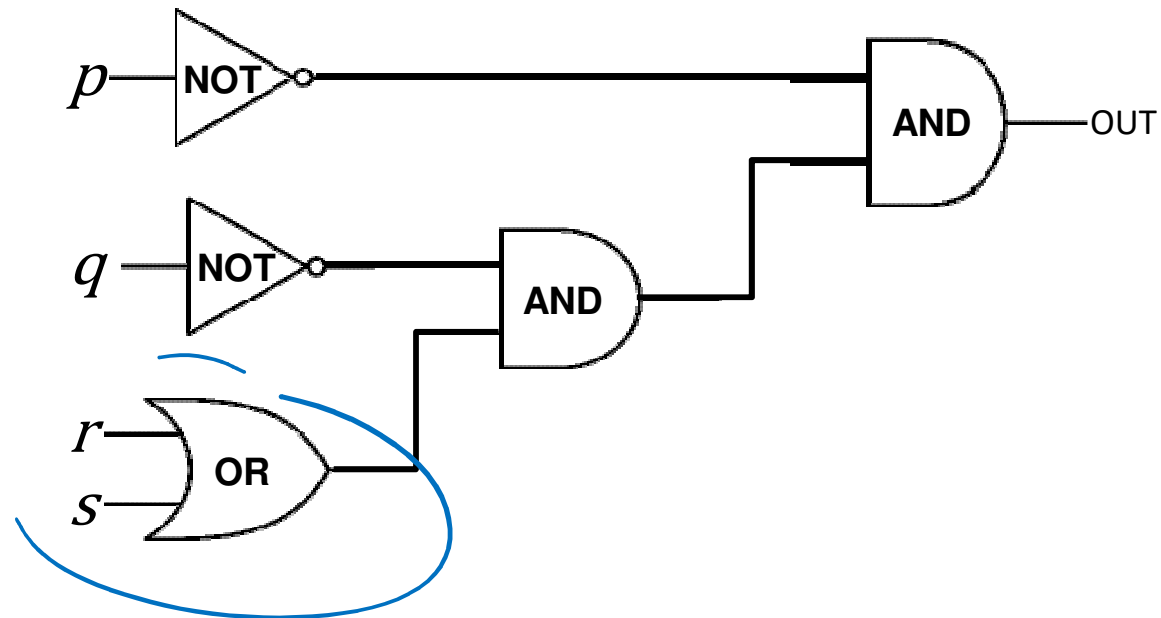


Combinational Logic Circuits



Values get sent along wires connecting gates

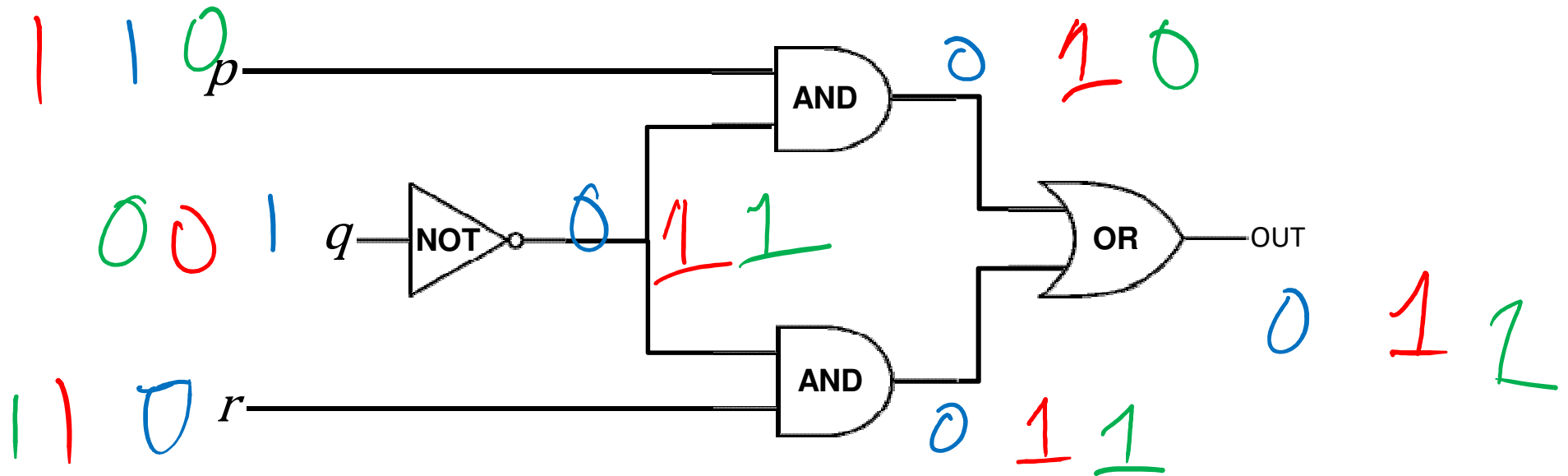
Combinational Logic Circuits



Values get sent along wires connecting gates

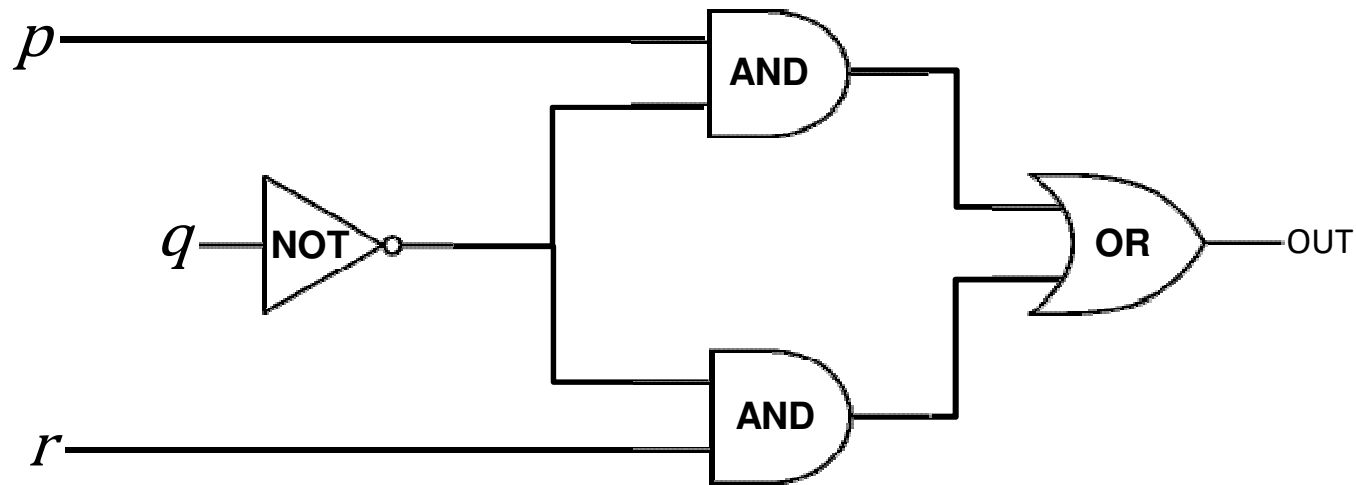
$$\neg p \wedge (\neg q \wedge (r \vee s))$$

Combinational Logic Circuits



Wires can send one value to multiple gates!

Combinational Logic Circuits



Wires can send one value to multiple gates!

$$(p \wedge \neg q) \vee (\neg q \wedge r)$$