EN1012 Electronic Devices and Circuits
Topic 6 - The Bridge to Digital Electronics

Upeka Premaratne

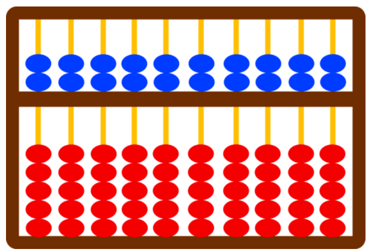
Department of Electronic and Telecommunication Engineering

October 30, 2017

Outline

Motivation

Computing

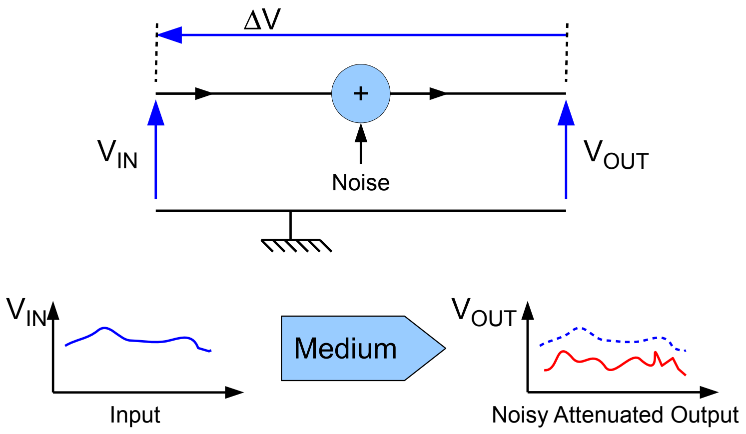


- Why was the abacus invented?
  - To make trade and commerce easier
- Deals with discrete numbers

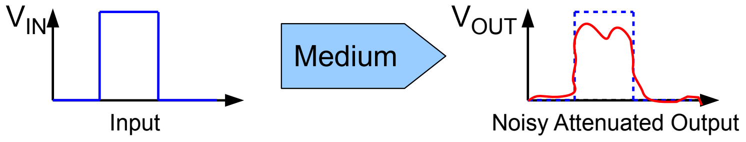
Analog to Digital

- Computing is all about discrete mathematical operations
  - Decimal in the abacus
  - The simplest discrete numbers are 0 and 1
- Computing devices need hardware to realize this discrete representation
  - Transistors in cutoff and saturation
  - Digital electronics requires analog hardware operating like switches
- Continuous values cannot be realized in a computer
  - Have to be approximated

Signal Transmission (Analog)



Signal Transmission (Digital)



- In addition it is possible to have
  - Error detection and correction (e.g. parity checks)
  - Data compression
  - Encryption

Logic and Reasoning

Prepositional Logic

- In prepositional logic a statement (*premise*) can be true or false
  1. Planet earth is flat
  2. Money does not grow on trees
  3. Capital punishment prevents crime
  4. A leopard killing a deer is bad
- Statements 1 (false) and 2 (true) fall under prepositional logic
- Statement 3 is both true and false
- Statements 4 is neither true nor false (Belnap logic)

Logical Reasoning and Computing

- Based upon the “truth” of the type of mathematical logic used
- The truth can never be absolute, depends on the axioms on which the mathematical logic is based upon (Shönfinkel 1924)
  - A system of logic cannot be both *complete* and *consistent* (Gödel's First Incompleteness Theorem)
  - A theory is *complete* if a formula Φ can either be Φ or $\overline{\Phi}$ can be reasoned from it (cannot have any *conundrums*)
  - A theory is *consistent* if an evaluated truth is always true (or vice versa) regardless of how it is evaluated. i.e., a result cannot be true when evaluated using one method and false if evaluated by some other method.

Logical Reasoning and Computing (Contd..)

A logic system can always have *conundrums* i.e., statements that “do not compute”

- Example 1:
  - Statement 1: X (a barber) says “I shave anyone who cannot shave themselves and no one else”
  - Statement 2: X is clean shaven
- Example 2:
  - Statement 1: Y says “All politicians are liars”
  - Statement 2: Y is a politician
- Nothing can be *reasoned* from the above examples.

Logical Reasoning and Computing (Contd..)

Logic and *common sense* in conflict

- Example 1 (Karl Popper's Paradox):
  - In a tolerant society should you tolerate intolerance?
- Example 2:
  - Do not use the lift in case of fire.

For the two examples, the results of logical and common sense based reasoning are in conflict.

Boolean Algebra

- Boolean algebra is a mathematical representation of prepositional logic
  - A boolean variable X represents a logical premise
  - $X \in \mathcal{B} = \{T, F\} = \{1, 0\}$ ($\mathcal{B}$ is the Boolean space)
- Basic operators
  - NOT operator (negation) $\neg A$ is False if A is True
  - OR operator (addition) True of either A or B or both are True
  - AND operator (multiplication) True iff both A and B are True

Boolean Algebra (Contd..)

- Tautology: an expression that is always True
- Contradiction: an expression that is always False
- Implication $A \Rightarrow B$
  - A is the *antecedent* and B is the *consequent*
  - An implication is False only if A is True and B is False
- Equivalence $A \Leftrightarrow B$
  - An equivalence is True iff A and B are both True or both False

Axioms of Boolean Algebra

Huntingdon Postulates

No.	Name	Properties
1	Closure	$0, 1 \in \mathcal{B}$
2	Identity Elements	$X + 0 = X$
		$X \cdot 1 = X$
3	Commutativity	$X + Y = Y + X$
		$X \cdot Y = Y \cdot X$

- *Commutativity:* elements are independent of order

Axioms of Boolean Algebra (Contd..)

Huntingdon Postulates (Contd..)

No.	Name	Properties
4	Distributivity	$X \cdot (Y + Z) = X \cdot Y + X \cdot Z$
		$X + Y \cdot Z = (X + Y) \cdot (X + Z)$
5	Complement	$X + \overline{X} = 1$
		$X \cdot \overline{X} = 0$
6	Distinct Elements	$0 \neq 1$

- *Duality:* every algebraic expression obtained using Hundingdon's Postulates are true when the operators $(+ \leftrightarrow \cdot)$ and identity elements $(0 \leftrightarrow 1)$ are interchanged.

Theorems of Boolean Algebra

Basic Theorems

No.	Name	Result
1		$X + X = X$
		$X \cdot X = X$
2		$X + 1 = 1$
		$X \cdot 0 = 0$
3	Involution (Double Negation)	$\overline{\overline{X}} = X$

Theorems of Boolean Algebra (Contd..)

Basic Theorems (Contd..)

No.	Name	Result
4	Associativity	$(X + Y) + Z = X + (Y + Z)$
		$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z)$
5	De Morgan	$\overline{(X + Y)} = \overline{X} \cdot \overline{Y}$
		$\overline{X \cdot Y} = \overline{X} + \overline{Y}$
6	Absorption	$X + X \cdot Y = X$
		$X \cdot (X + Y) = X$

Logic Gates

Logic Gate Types

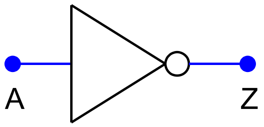
Logic gates realize Boolean operators in electronic circuits

- Basic Gates (Basic Operators)
  - NOT, OR, AND
- Derived Gates
  - NOR, NAND, XOR, XNOR
- Special Gates
  - Tri-State buffers and inverters

Each gate consists of a *symbol* and *truth table* that describes its operation. Later on it will be shown how logic gates are implemented using electronic components.

NOT Gate

Symbol



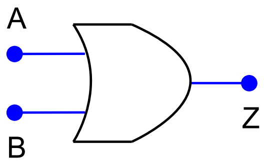
Truth Table

A	Z
0	1
1	0

$$Z = \overline{A}$$

OR Gate

Symbol



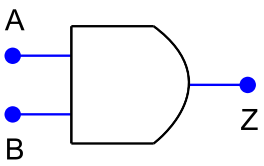
Truth Table

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

$$Z = A + B$$

AND Gate

Symbol



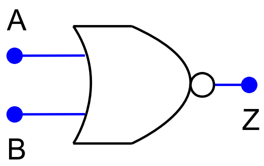
Truth Table

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

$$Z = A \cdot B$$

NOR Gate

Symbol



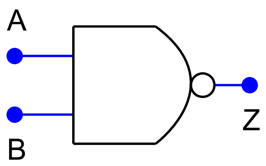
Truth Table

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

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

NAND Gate

Symbol



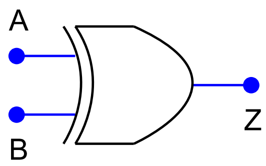
Truth Table

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

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

XOR Gate

Symbol



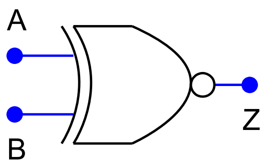
Truth Table

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

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

XNOR Gate

Symbol



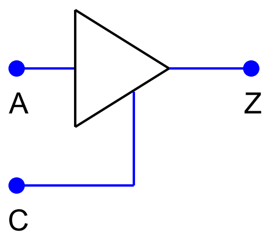
Truth Table

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

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

Tri-State Buffer

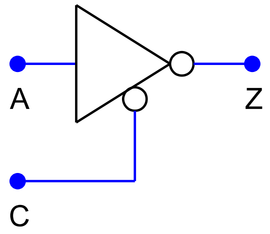
Symbol



- When $C = 0$
  - Open Circuit (hi-Z)
- When $C = 1$
  - $Z = A$

Tri-State Inverter

Symbol



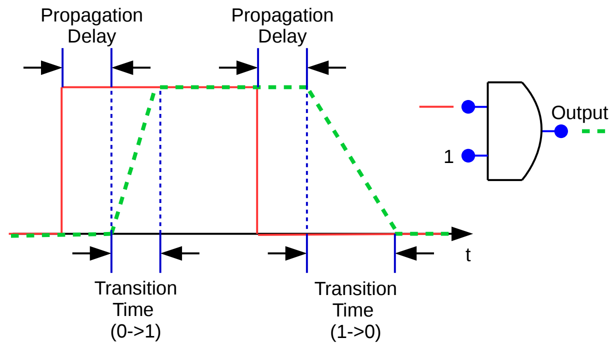
- When $C = 1$
  - Open Circuit (hi-Z)
- When $C = 0$
  - $Z = \overline{A}$

Gate Parameters

Gate Parameters

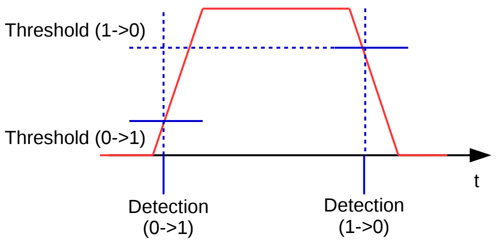
- Main two families
  - Transistor Transistor Logic (TTL)
  - Complementary Metal Oxide Semiconductor (CMOS) Logic
- CMOS is sensitive to static electricity, requires less energy than TTL but also slower
- Main parameters
  - Propagation delay
  - Transition time
  - Detection thresholds
  - Fan out

Transition Time and Propagation Delay



- Propagation delay can result in *glitches* and *race-hazards*

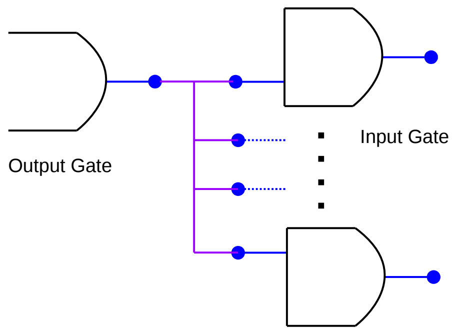
Detection Thresholds



- Logic ‘0’ and ‘1’ are detected differently

Motivation Logic and Reasoning Logic Gates **Gate Parameters** Conclusion

Fan Out



- Loading due to fan out can result in *incorrect detection*

Upeka Premaratne EN1012 Electronic Devices and Circuits

Conclusion

Upeka Premaratne EN1012 Electronic Devices and Circuits

Motivation Logic and Reasoning Logic Gates Gate Parameters **Conclusion**

Summary

- Digital electronics (discrete) is more versatile than analog electronics
  - However, digital electronic devices are fabricated using analog components
- Has a mathematical foundation
  - Prepositional logic
  - Boolean algebra
  - Analog components have a quantum mechanic foundation
- The fundamental elements of a digital circuits are logic gates
  - Realized using TTL or CMOS technology

Upeka Premaratne EN1012 Electronic Devices and Circuits

Next Lecture...

Basic Digital Circuits