Lecture 5: Combinational Logic - Part I

CS207: Digital Logic

Jialin Liu

Department of Computer Science and Engineering (CSE)
Southern University of Science and Technology (SUSTech)

9 October 2022

Acknowledgement

These slides were prepared based on the slides by Dr. Jianqiao Yu and the ones by Prof. Georgios Theodoropoulos of the Department of CSE at the SUSTech, as well as the contents of the following book:
M. M. Mano and M. Ciletti, *Digital design: with an introduction to the Verilog HDL*. Pearson, 2013

Recap: Gate-level Implementation

- Universal gates: `NAND` and `NOR` gates.
- Procedure of two-level and multi-level implementations using `NAND` or `NOR` gates.
- `Exclusive-OR` gate for constructing error detection circuits.

- Lecture 1: Binary Numbers
- Lecture 2: Boolean Algebra and Logic Gates
- Lecture 3 Gate-Level Minimisation
- Lecture 4: Gate-Level Implementation

→ Now we will use the knowledge acquired in previous weeks to formulate systematic analysis and design procedures for combinational circuits.

Types of Logic Circuits

- "A **combinational circuit** consists of logic gates whose outputs **at any time** are determined from only the present combination of inputs."
  → **Combinational logic** (Lecture 5 and Lecture 6).
- "**Sequential circuits** employ memory elements in addition to logic gates. Their outputs are a function of the inputs and the state of the memory elements."
  - State of the memory elements: a function of previous inputs.
  - Outputs of a sequential circuit depend on values of present inputs and past inputs.

  → **Memory elements** (Lecture 7).
  → **Sequential logic** (from Lecture 8).

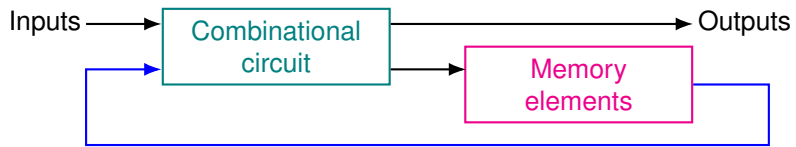


Figure: Block diagram of sequential circuit.

Lecture 5 & Lecture 6: Combinational Circuit

Introduction to Combinational Circuit

Analysis of Combinational Circuits

- Output A Set of Boolean Functions
- Output A Truth Table

Design of Combinational Circuits

Summary

Outline of This Lecture

Combinational Circuit

- A combinational circuit consists of an interconnection of logic gates.
  → No feedback paths or memory elements.
- Combinational logic gates
  - react to values of input signals,
  - produce output signal values,
  - transform binary information from the given input data to a required output data.

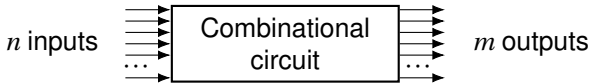
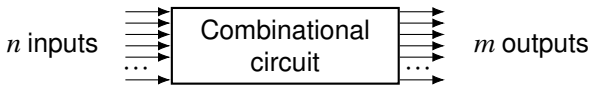


Figure: A block diagram of a combinational circuit. n input binary variables come from external sources, and m output variables (produced by the circuit) go to external destinations.

Representations of Combinational Circuits

- Two representations of a combinational circuit:
  1. 2^n possible input combinations: truth table
  2. m outputs: m Boolean functions, each expressed with the n inputs.



What We Will Learn about Combinational Circuits

- **Formulate systematic analysis and design procedures for combinational circuits**:
  1. Analyse the behaviour of a given logic circuit.
  2. Synthesise a circuit that will have a given behaviour.
  3. Write hardware description language (HDL) models for some common circuits.

- **Standard components**: combinational circuits that are employed extensively in the design of digital systems.
  - Adders, subtractors, comparators, decoders, encoders, and multiplexers.
  - Available in integrated circuits as medium-scale integration (MSI) circuits.
  - Also used as standard cells in complex very large-scale integrated (VLSI) circuits such as application-specific integrated circuits (ASICs).

Outline of This Lecture

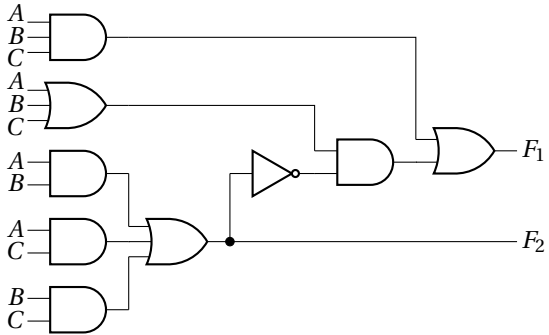
Analysis of Combinational Circuits

- **Analysis of a combinational circuit**: determine the function of the circuit.
  - Given a logic diagram.
  - Output a set of Boolean functions, a truth table, or, possibly, an explanation of the circuit operation.
- If a function name or an explanation is given along the circuit, just verify if the given information is correct.
- The analysis can be performed manually or by using a computer simulation program.

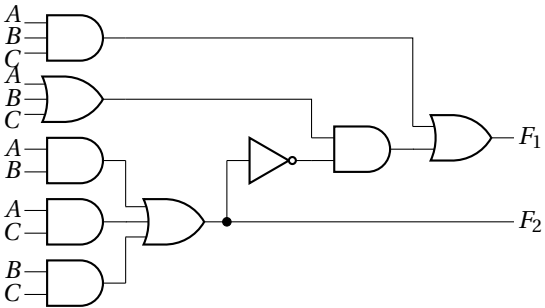
Outline

Analysis of Combinational Circuits: Example

- Input: a logic diagram.
- Output: a set of Boolean functions.

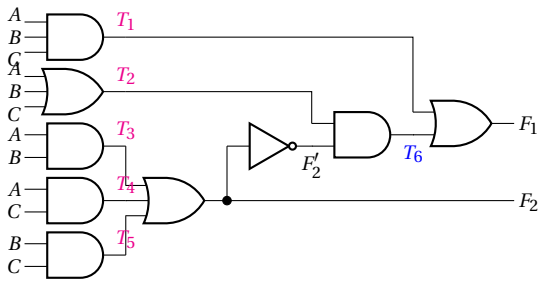


1. Make sure that the given circuit is combinational and not sequential.
   - The diagram of a combinational circuit has logic gates with no feedback paths or memory elements.
   - **Feedback path**: a connection from the output of one gate to the input of a second gate whose output forms part of the input to the first gate.



- There is no feedback path!
- It is a combinational circuit.

2. Obtain the output Boolean functions from the given diagram:
   - 2.1 Label all gate outputs that are a function of input variables with arbitrary symbols (i.e., only inputs, no other intermediate variables). Determine the Boolean functions for each gate output.
   - 2.2 Label the gates that are a function of input variables and previously labeled gates with other arbitrary symbols. Determine their Boolean functions.
   - 2.3 Repeat the process outlined in step 2.2 until the outputs of the circuit are obtained.
   - 2.4 By repeated substitution of previously defined functions, obtain the output Boolean functions in terms of input variables.



- $T_1 = ABC$
- $T_2 = A + B + C$
- $T_3 = AB$
- $T_4 = AC$
- $T_5 = BC$
- $T_6 = T_2 F_2'$
- $F_1 = T_1 + T_6$
- $F_2 = T_3 + T_4 + T_5$

- Form a series of substitutions: $F_2 = T_3 + T_4 + T_5 = AB + AC + BC$

$$F_1 = T_1 + T_6 = ABC + T_2 F_2' = ABC + (A + B + C)(AB + AC + BC)' = ABC + A'BC' + A'B'C + AB'C'$$

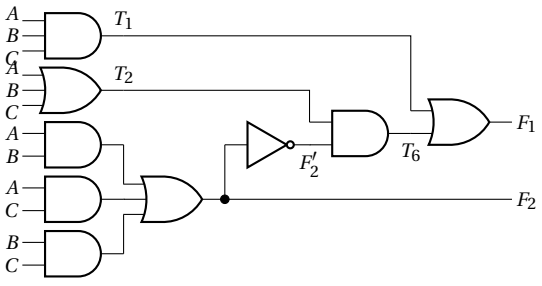
Analysis of Combinational Circuits: Steps

- **Input**: a logic diagram.
  1. **Make sure that the given circuit is combinational and not sequential.**
     - The diagram of a combinational circuit has logic gates with no feedback paths or memory elements.
     - **Feedback path**: a connection from the output of one gate to the input of a second gate whose output forms part of the input to the first gate.
  2. **Obtain the output Boolean functions from the given diagram:**
     - 2.1 Label all gate outputs that are a function of input variables with arbitrary symbols – but with meaningful names (i.e., only inputs, no other intermediate variables). Determine the Boolean functions for each gate output.
     - 2.2 Label the gates that are a function of input variables and previously labeled gates with other arbitrary symbols. Determine their Boolean functions.
     - 2.3 Repeat the process outlined in step 2.2 until the outputs of the circuit are obtained.
     - 2.4 By repeated substitution of previously defined functions, obtain the output Boolean functions in terms of input variables.
- **Output**: a set of Boolean functions.

Outline

Output A Truth Table: Steps

- Truth table is simple with Boolean function
  1. Determine the number of input variables. For n inputs, form the 2^n combinations from 0 to $2^n - 1$.
  2. Label the outputs of the intermediate gates.
  3. Obtain the truth table for these outputs.
  4. Obtain the truth table for the remaining outputs.



A	B	C	F_2	F_2'	T_1	T_2	T_6	F_1
0	0	0	0	1	0	0	0	0
0	0	1	0	1	0	1	1	1
0	1	0	0	1	0	1	1	1
0	1	1	1	0	0	1	0	0
1	0	0	0	1	0	1	1	1
1	0	1	1	0	0	1	0	0
1	1	0	1	0	0	1	0	0
1	1	1	1	0	1	1	0	1

Outline of This Lecture

Design of Combinational Circuits: Steps

- Design of combinational circuits: develop a logic circuit diagram or a set of Boolean functions from specification of the design objective.
  - Input: a specification of the design objective.
  - Output: a logic circuit diagram or a set of Boolean functions.

- Steps:
  - From the specifications of the circuit, determine the required number of inputs and outputs and assign a symbol to each.
  - Derive the truth table that defines the required relationship between inputs and outputs.
  - Obtain the simplified Boolean functions for each output as a function of the input variables.
  - Draw the logic diagram and verify the correctness of the design (manually or by simulation).

Example: A Conversion Circuit (1/5)

- Convert from BCD code to excess-3 code (c.f. page 23 of [1])[1].

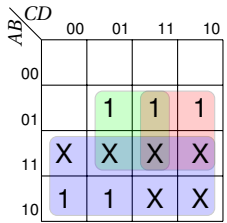
Input BCD				Output Code			
A	B	C	D	w	x	y	z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

- Each code uses 4 bits, therefore 4 inputs and 4 outputs.
- And we already have the truth table.
- Then the Boolean functions. How?
  - Remember K-maps?

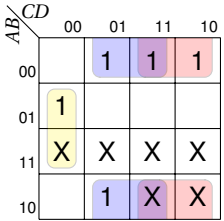
[1] Self-complementary BCD code used to represent the decimal numbers.

Example: A Conversion Circuit (2/5)

$$w = A + BC + BD$$

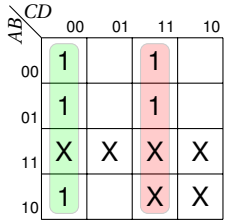


$$x = B'C + B'D + BC'D'$$

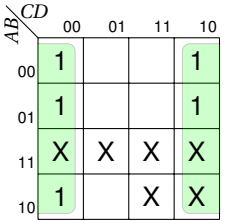


Example: A Conversion Circuit (3/5)

$$y = CD + C'D'$$



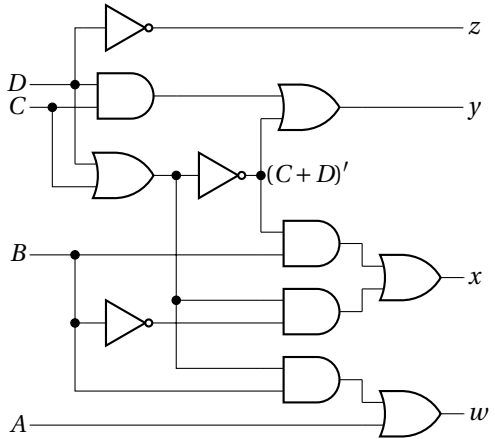
$$z = D'$$



Example: A Conversion Circuit (4/5)

- We manipulate these Boolean functions to reuse common gates:
  - $w = A + BC + BD = A + B(C + D)$.
  - $x = B'C + B'D + BC'D' = B'(C + D) + B(C + D)'$.
  - $y = CD + C'D' = CD + (C + D)'$.
  - $z = D'$.

Example: A Conversion Circuit (5/5)



Outline of This Lecture

Summary

- Combinational circuit: consists of logic gates whose outputs at any time are determined from only the present combination of inputs.
  - Procedure of analysing combinational circuits.
  - Procedure of designing combinational circuits.
- Today's lab:
  - Analysis of Assignment 1 - Part 2.
  - Behavioural modelling in Verilog.

Essential Reading

- Essential reading for this lecture: pages 125–132 of the textbook.
- Essential reading for next lecture: pages 148–158 of the textbook.

[1] M. M. Mano and M. Ciletti, *Digital design: with an introduction to the Verilog HDL*. Pearson, 2013