

Announcements

- If you haven't shown the grader your proof of prerequisite, please do so by **11:59 pm** on **09/08/2017** (Friday).
- HW1 is available online already (due on **September 27** in class), make sure that you do the right questions. The text of HW1 is also posted online now.

The grader's office has been changed!

- GMCS401 will no longer be a cs370 lab.
- CS teaching lab (GMCS 425) and Math lab (GMCS 422) have installed the LogicWorks software.
- These two rooms will also be the location for the grader's office hours.
- On Monday and Wednesday, the grader will have office hours from 12PM-1:45PM in GMCS 425. On Thursday, the grader will have office hours from 2:00PM-3:30PM in GMCS 422.

Overview – Canonical Forms

- **What are Canonical Forms?**
- **Minterms**
- **Index Representation of Minterms**
- **Sum-of-Minterm (SOM) Representations**

Canonical Forms

- It is useful to specify **Boolean functions in a form that:**
 - Allows comparison for equality.
 - Has a correspondence to the truth tables
- **Canonical Forms in common usage:**
 - **Sum of Minterms (SOM)**

Minterms

- **Minterms** are AND terms with **every** variable present in either true or complemented form.
- Given that each binary variable may appear normal (e.g., x) or complemented (e.g., \overline{x}), there are 2^n minterms for n variables.
- **Example:** Two variables (X and Y) produce $2 \times 2 = 4$ combinations:
 - XY (both normal)
 - $X\overline{Y}$ (X normal, Y complemented)
 - $\overline{X}Y$ (X complemented, Y normal)
 - $\overline{X}\overline{Y}$ (both complemented)
- Thus there are **four minterms** of two variables.

Minterms

- **Examples: Two variable minterms and maxterms.**

Index	Minterm	Maxterm
0	$\bar{x} \bar{y}$	$x + y$
1	$\bar{x} y$	$x + \bar{y}$
2	$x \bar{y}$	$\bar{x} + y$
3	$x y$	$\bar{x} + \bar{y}$

- **The index above is important for describing which variables in the terms are true and which are complemented.**

Standard Order

- **Minterms are designated with a subscript**
- **The subscript is a number, corresponding to a binary pattern**
- **The bits in the pattern represent the complemented or normal state of each variable listed in a standard order.**
- **All variables will be present in a minterm and will be listed in the same order (usually alphabetically)**
- **Example: For variables a, b, c:**
 - **Maxterms: $(a + b + \bar{c})$, $(a + b + c)$**
 - **Terms: $(b + a + c)$, $a \bar{c} b$, and $(c + b + a)$ are NOT in standard order.**
 - **Minterms: $a \bar{b} c$, $a b c$, $\bar{a} \bar{b} c$**
 - **Terms: $(a + c)$, $\bar{b} c$, and $(\bar{a} + b)$ do not contain all variables**

Purpose of the Index

- The index for the minterm or maxterm, expressed as a binary number, is used to determine whether the variable is shown in the true form or complemented form.
- For Minterms:
 - “1” means the variable is “Not Complemented” and
 - “0” means the variable is “Complemented”.
- For Maxterms:
 - “0” means the variable is “Not Complemented” and
 - “1” means the variable is “Complemented”.

Index Example in Three Variables

- **Example:** (for three variables)
- Assume the variables are called **X, Y, and Z.**
- The standard order is **X, then Y, then Z.**
- The **Index 0** (base 10) = **000** (base 2) for three variables). All three variables are complemented for **minterm 0** ($\bar{X}, \bar{Y}, \bar{Z}$) and no variables are complemented for **Maxterm 0** (X,Y,Z).
 - Minterm 0, called m_0 is $\bar{X}\bar{Y}\bar{Z}$.
 - Maxterm 0, called M_0 is $(X + Y + Z)$.
 - Minterm 6 ?

Index Examples – Four Variables

Index	Binary	Minterm	Maxterm
i	Pattern	m_i	M_i
0	0000	$\bar{a}\bar{b}\bar{c}\bar{d}$	$a + b + c + d$
1	0001	$\bar{a}\bar{b}\bar{c}d$?
3	0011	?	$a + b + \bar{c} + \bar{d}$
5	0101	$\bar{a}b\bar{c}d$	$a + \bar{b} + c + \bar{d}$
7	0111	?	$a + \bar{b} + \bar{c} + \bar{d}$
10	1010	$a\bar{b}c\bar{d}$	$\bar{a} + b + \bar{c} + d$
13	1101	$ab\bar{c}d$?
15	1111	$abcd$	$\bar{a} + \bar{b} + \bar{c} + \bar{d}$

Function Tables for Both

- **Minterms of 2 variables**

x y	m₀	m₁	m₂	m₃
0 0	1	0	0	0
0 1	0	1	0	0
1 0	0	0	1	0
1 1	0	0	0	1

Maxterms of 2 variables

x y	M₀	M₁	M₂	M₃
0 0	0	1	1	1
0 1	1	0	1	1
1 0	1	1	0	1
1 1	1	1	1	0

Observations

- **In the function tables:**
 - Each minterm has one and only one 1 present in the 2^n terms (a minimum of 1s). All other entries are 0.
 - Each maxterm has one and only one 0 present in the 2^n terms. All other entries are 1 (a maximum of 1s).
 - We can implement any function by "ORing" the minterms corresponding to "1" entries in the function table. These are called the minterms of the function.
 - We can implement any function by "ANDing" the maxterms corresponding to "0" entries in the function table. These are called the maxterms of the function.
 - This gives us two canonical forms:
 - Sum of Minterms (SOM)
 - Product of Maxterms (POM)
- for stating any Boolean function.**

Minterm Function Example

- Example: Find $F_1 = m_1 + m_4 + m_7$

- $F_1 = \bar{x} \bar{y} z + x \bar{y} \bar{z} + x y z$

x y z	index	m_1	+	m_4	+	m_7	= F_1
0 0 0	0	0	+	0	+	0	= 0
0 0 1	1	1	+	0	+	0	= 1
0 1 0	2	0	+	0	+	0	= 0
0 1 1	3	0	+	0	+	0	= 0
1 0 0	4	0	+	1	+	0	= 1
1 0 1	5	0	+	0	+	0	= 0
1 1 0	6	0	+	0	+	0	= 0
1 1 1	7	0	+	0	+	1	= 1

Minterm Function Example

- $F(A, B, C, D, E) = m_2 + m_9 + m_{17} + m_{23}$
- $F(A, B, C, D, E) =$

Canonical Sum of Minterms

- **Any** Boolean function can be expressed as a **Sum of Minterms**.
 - For the function table, the minterms used are the terms corresponding to the 1's
 - For expressions, expand all terms first to explicitly list all minterms. Do this by “ANDing” any term missing a variable v with a term $(v + \bar{v})$.
- **Example:** Implement $f = x + \bar{x} \bar{y}$ as a sum of minterms.

First expand terms: $f = x(y + \bar{y}) + \bar{x} \bar{y}$

Then distribute terms: $f = xy + x\bar{y} + \bar{x} \bar{y}$

Express as sum of minterms: $f = m_3 + m_2 + m_0$

Another SOM Example

- **Example: $F = A + \bar{B} C$**
- **There are three variables, A, B, and C which we take to be the standard order.**
- **Expanding the terms with missing variables:**

- **Collect terms (removing all but one of duplicate terms):**
- **Express as SOM:**

Shorthand SOM Form

- From the previous example, we started with:

$$F = A + \bar{B} C$$

- We ended up with:

$$F = m_1 + m_4 + m_5 + m_6 + m_7$$

- This can be denoted in the formal shorthand:

$$F(A, B, C) = \Sigma_m(1,4,5,6,7)$$

- Note that we explicitly show the standard variables in order and drop the “m” designators.

Standard Forms

- **Standard Sum-of-Products (SOP) form:**
equations are written as an **OR** of **AND** terms
- **Standard Product-of-Sums (POS) form:**
equations are written as an **AND** of **OR** terms
- **Examples:**
 - SOP: $A B C + \bar{A} \bar{B} C + B$
 - POS: $(A + B) \cdot (A + \bar{B} + \bar{C}) \cdot C$
- These “mixed” forms are **neither SOP nor POS**
 - $(A B + C) (A + C)$
 - $A B \bar{C} + A C (A + B)$

Standard Sum-of-Products (SOP)

- **A sum of minterms form for n variables can be written down directly from a truth table.**
 - **Implementation of this form is a two-level network of gates such that:**
 - **The first level consists of n -input AND gates, and**
 - **The second level is a single OR gate (with fewer than 2^n inputs).**
- **This form often can be simplified so that the corresponding circuit is simpler.**

Standard Sum-of-Products (SOP)

- **A Simplification Example:**

- $F(A, B, C) = \Sigma m(1,4,5,6,7)$

- **Writing the minterm expression:**

$$F = \overline{A} \overline{B} C + A \overline{B} \overline{C} + A \overline{B} C + ABC\overline{C} + ABC$$

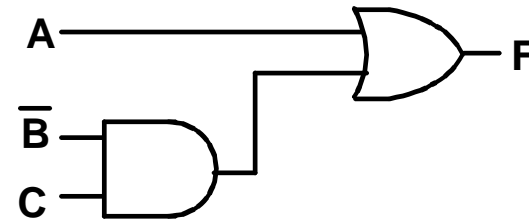
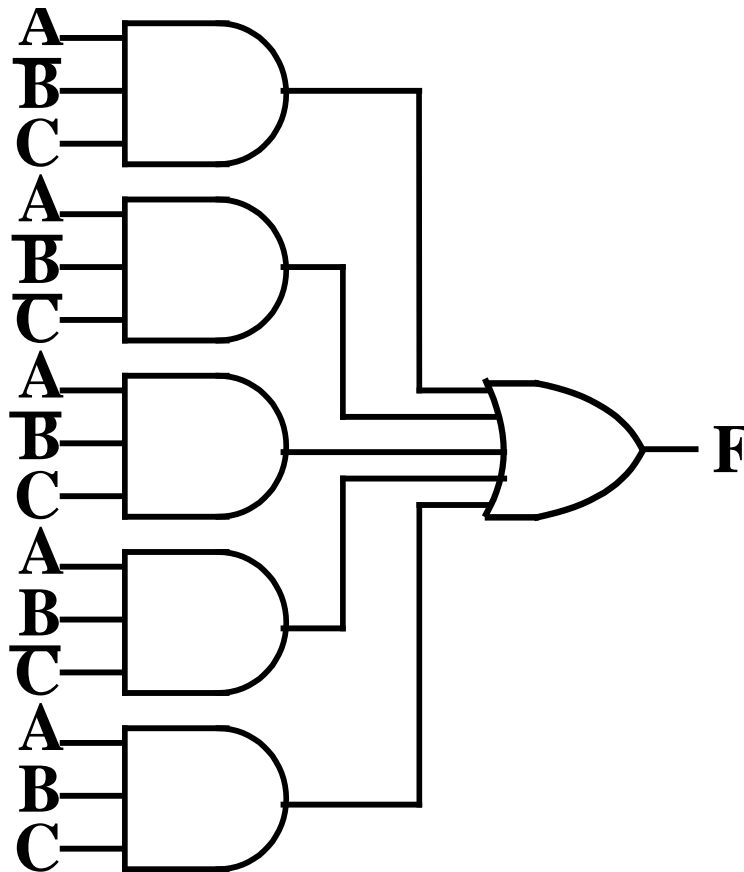
- **Simplifying:**

$$F =$$

- **Simplified F contains 3 literals compared to 15 in minterm F**

AND/OR Two-level Implementation of SOP Expression

- The two implementations for F are shown below – it is quite apparent which is simpler!



Weekly Exercise 1-2

- Problems (P102 on text book)
- 2-1; 2-2; 2-7; 2-9(c); 2-10(c); 2-12(b)
- Get familiar with Table 2-6 (P49 on text book)
- Get familiar with the 6 identities on P53.

Hi-Impedance Outputs

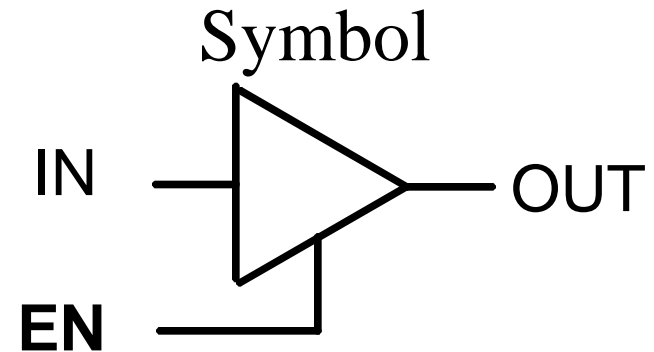
- Logic gates introduced thus far
 - have 1 and 0 output values,
 - cannot have their outputs connected together, and
 - transmit signals on connections in only one direction.
- Three-state logic adds a third logic value, Hi-Impedance (Hi-Z), giving three states: 0, 1, and Hi-Z on the outputs.
- The presence of a Hi-Z state makes a gate output as described above behave quite differently:
 - “1 and 0” become “1, 0, and Hi-Z”
 - “cannot” becomes “can,” and
 - “only one” becomes “two”

Hi-Impedance Outputs (continued)

- What is a Hi-Z value?
 - The Hi-Z value behaves as an open circuit
 - This means that, looking back into the circuit, the output appears to be disconnected.
 - It is as if a switch between the internal circuitry and the output has been opened.
- Hi-Z may appear on the output of any gate, but we restrict gates to:
 - a 3-state buffer, or
 - a transmission gate,each of which has one data input and one control input.

The 3-State Buffer

- For the symbol and truth table, IN is the data input, and EN, the control input.
- For $EN = 0$, regardless of the value on IN (denoted by X), the output value is Hi-Z.
- For $EN = 1$, the output value follows the input value.
- Variations:
 - Data input, IN, can be inverted
 - Control input, EN, can be inverted by addition of “bubbles” to signals.



Truth Table

EN	IN	OUT
0	X	Hi-Z
1	0	0
1	1	1

Terms of Use

- © 2004 by Pearson Education, Inc. All rights reserved.
- The following terms of use apply in addition to the standard Pearson Education [Legal Notice](#).
- Permission is given to incorporate these materials into classroom presentations and handouts only to instructors adopting Logic and Computer Design Fundamentals as the course text.
- Permission is granted to the instructors adopting the book to post these materials on a protected website or protected ftp site in original or modified form. All other website or ftp postings, including those offering the materials for a fee, are prohibited.
- You may not remove or in any way alter this Terms of Use notice or any trademark, copyright, or other proprietary notice, including the copyright watermark on each slide.
- [Return to Title Page](#)