Dr.SNS RAJALAKSHMI COLLEGE OF ARTS AND SCIENCE COIMBATORE-49

Accredited by NAAC (Cycle III) with “A+” Grade

Recognized by UGC, Approved by AICTE, New Delhi and

Affiliated to Bharathiar University,

Coimbatore.

DEPARTMENT OF COMPUTER APPLICATIONS

Course Code / Course Name: **23UCU403 /Computer System Architecture**

YEAR: **2023-2024**

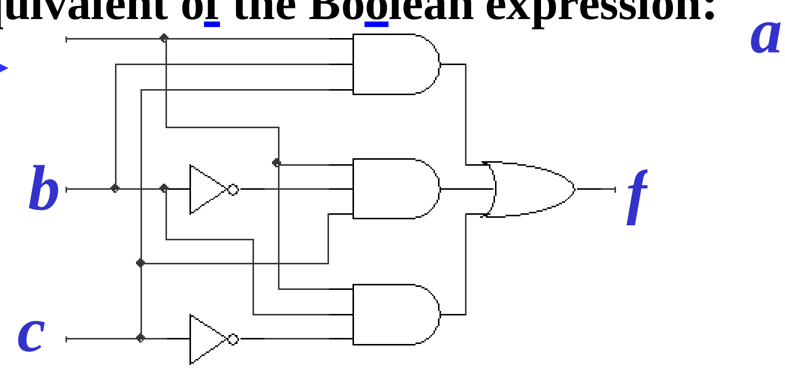
CLASS: **I BCA “B”**

STAFF NAME: **Dr.A.DEVI**

UNIT I – Data Simplification

Simplifying Logic Circuits with Karnaugh Maps

- **The circuit at the top right is the logic equivalent of the Boolean expression:**
  $f = abc + abc + abc$



- **Now, as we have seen, this expression can be simplified (reduced to fewer terms) from its original form, using the Boolean identities as shown at right.**
- **The circuit may be simplified as follows:**

  $f = abc + a\bar{b}c + ab\bar{c}$

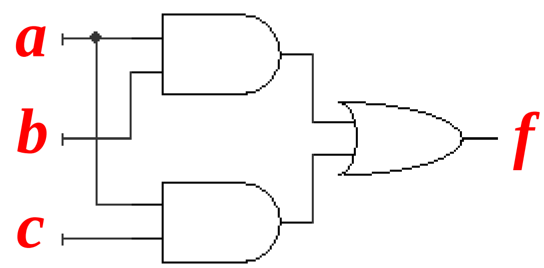
  $f = abc + abc + abc + ab\bar{c}$

  (since $x=x+x$)

  $f = (abc + abc) + (abc + ab\bar{c})$

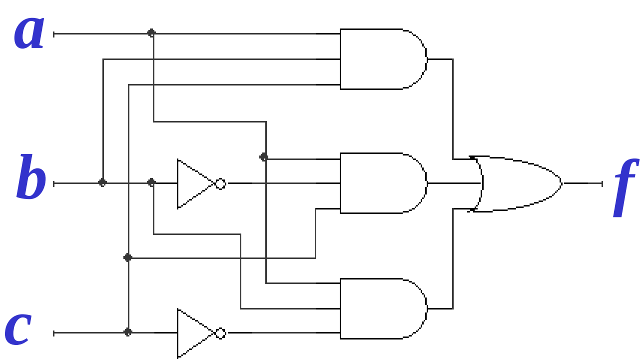
  $f = ac(b + b) + ab(c + c)$

  or, $f = ac + ab$

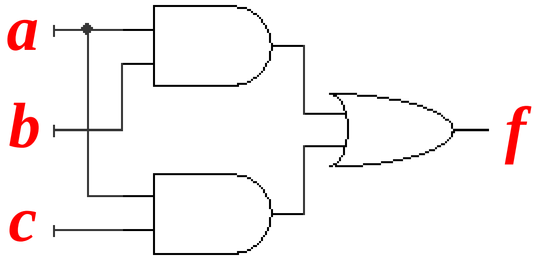


Simplifying Logic Circuits (2)

- Since you have now had some experience with simplification of Boolean expressions, this example is (hopefully) familiar and understandable.
- However, for more complex Boolean expressions, the identity/substitution approach can be VERY cumbersome (at least, for humans).
- Instead of this approach, we can use a graphical technique called the Karnaugh map.



Original logic circuit



Simplified equivalent logic circuit

Karnaugh Maps

- Another approach to simplification is called the <u>Karnaugh map</u>, or K-map.
- A K-map is a truth table graph, which aids in visually simplifying logic.
- It is useful for up to 5 or 6 variables, and is a good tool to help understand the process of logic simplification.
- The algebraic approach we have used previously is also used to analyze complex circuits in industry (<u>computer analysis)</u>.
- At the right is a 2-variable K-map.
- This very simple K-map demonstrates that <u>an n-variable K-map contains all the combination of the n variables in the K-map space</u>.

	$\bar{y}$	y
$\bar{x}$	00 0	01 1
x	10 2	11 3

This minterm is expressed as $f = \bar{x}y$.

<u>Two-Variable K-map</u>, labeled for SOP terms. Note the four squares represent all the com-binations of the two K-map variables, or <u>minterms</u>, in x & y (example above).

Three-Variable Karnaugh Map

- A useful K-map is one of three variables.
- Each square represents a 3-variable minterm or maxterm.
- All of the 8 possible 3-variable terms are represented on the K-map.
- When moving horizontally or vertically, <u>only 1 variable changes between adjacent squares</u>, never 2. <u>This property of the K-map, is unique and accounts for its unusual numbering system</u>.
- The K-map shown is one labeled for SOP terms. It could also be used for a POS problem, but we would have to re-label the variables.

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{x}$	000 0	001 1	011 3	010 2
x	100 4	101 5	111 7	110 6

As an example, this minterm cell (011) represents the minterm $f = \bar{x}yz$.

Four Variable Karnaugh Map

- **A 4-variable K-map can simplify problems of four Boolean variables.***
- **The K-map has one square for each possible minterm (16 in this case).**
- **Migrating one square horizontally or vertically never results in more than <u>one</u> variable changing (square designations also shown in hex).**

	$\overline{y}\,\overline{z}$	$\overline{y}z$	yz	$y\overline{z}$
$\overline{w}\,\overline{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\overline{w}x$	0100 4 4	0101 5 5	0111 7 7	0110 6 6
wx	1100 C 12	1101 D 13	1111 F 15	1110 E 14
$w\overline{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

Note that this is still an SOP K-map.

*** Note that on all K-maps, the left and right edges are a common edge, while the top and bottom edges are also the same edge. Thus, <u>the top and bottom rows are adjacent, as are the left and right columns</u>.**

Exercise 1

- We will use the Karnaugh map to simplify Boolean expressions by placing minterm or maxterm values on the map and then grouping terms to develop simpler Boolean expressions.
- Let's practice placing some terms on the K-map shown. For the SOP Boolean expression below, place 1's and zeros on the map.

$$f = \bar{w}xyz + \bar{w}\bar{x}yz + wxyz + wxy\bar{z}$$

	yz	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 5 5	0111 7 7	0110 6 6
wx	1100 C 12	1101 D 13	1111 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

Karnaugh map labeled for SOP problem solution.

Karnaugh Map Comments

- K-maps can be labeled many ways, but in EE 2310, <u>always</u> use this labeling!
- Each square is unique. We can label it in <u>binary, decimal, or hex</u>. We can also designate the Boolean function <u>by the K-map squares it occupies</u>.
- The minterms on the K-map can be labeled as f=Σm(5, 7, 13, 15) in decimal, or f= Σm(5, 7, D, F) in hex.*
- Given the Sigma (Σ) coordinates, you could immediately deduce that the SOP function was: $f = wx\,yz + wxyz + wx\,yz + wxyz$

* Σm is the symbol for the "Boolean sum" (OR) coordinates of the squares.

	yz	yz	yz	$y\bar{z}$
$\overline{wx}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 1 5 5	0111 1 7 7	0110 6 6
wx	1100 C 12	1101 1 D 13	1111 1 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

Observe that the Σ notations (in either SOP or POS) <u>completely describe the Boolean function mapped on the K-map</u>, as long as one knows what the input variables are.

Exercise 2

- Try your hand at developing the Boolean expression from the $f = \Sigma m$ designation on a K-map.
- The Σm designation of a Boolean function is given as

  $f = \Sigma m(9, B, D, F)$ (SOP).
- Find the Boolean expression by plotting the 1's on the chart and developing the expression from the minterms.

	$\overline{y}\,\overline{z}$	$\overline{y}z$	yz	$y\overline{z}$
$\overline{w}\,\overline{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\overline{w}x$	0100 4 4	0101 5 5	0111 7 7	0110 6 6
wx	1100 C 12	1101 D 13	1111 F 15	1110 E 14
$w\overline{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

Karnaugh Map Terminology

- In the K-map at right, each small square, or cell, represents one 4-variable Boolean AND function, or minterm (if we wanted, we could label it to represent OR functions or maxterms).
- Any square or rectangular group of cells that is a power of 2 (1, 2, 4, 8, 16) is called an implicant.
- All of the groups of squares in the K-map to the left (including the single square) represent implicants of different sizes.

	yz	yz	yz	$y\bar{z}$
wx	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 5 5	0111 7 7	0110 6 6
wx	1100 C 12	1101 D 13	1111 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

Examples of various cell groupings, all of which represent K-map implicants.

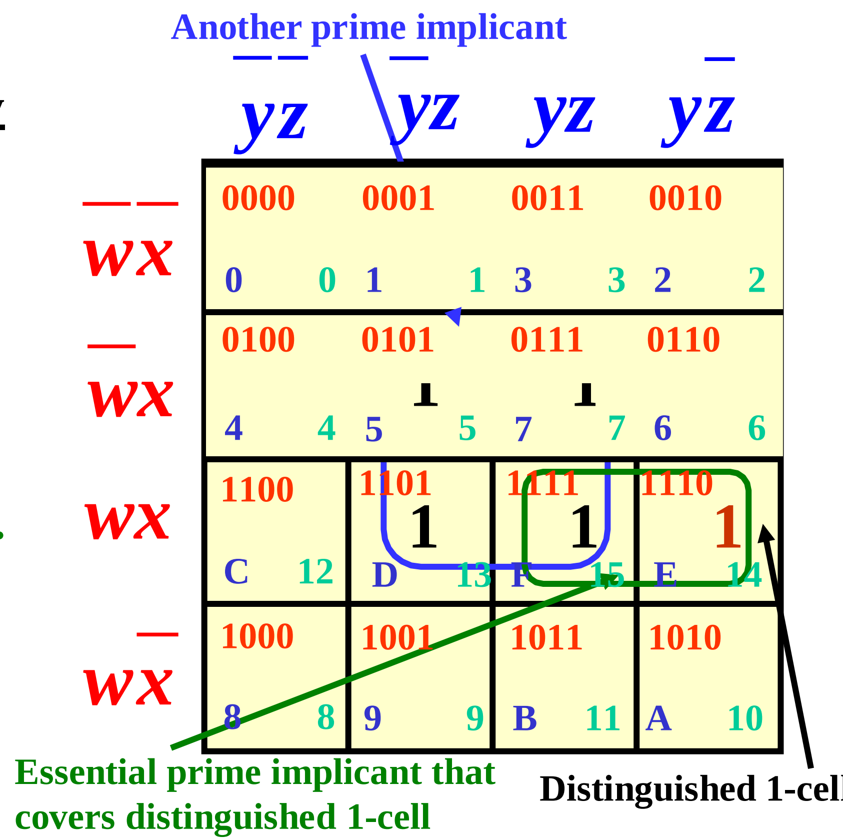
Prime Implicants

- We will be simplifying Boolean functions plotting their values on a K-map and grouping them into prime implicants.
- What is a prime implicant? It is an implicant that covers as many 1 values (SOP K-map) or 0 values (POS K-map) as possible, yet still retains the identity of implicant (# of cells = power of 2, rectangular or square shape).
- Some SOP prime implicants are shown on the adjoining K-map.

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 1 0 0	0001 0 1 1	0011 1 3 3	0010 1 2 2
$\bar{w}x$	0100 1 4 4	0101 0 5 5	0111 1 7 7	0110 1 6 6
wx	1100 1 C 12	1101 1 D 13	1111 1 F 15	1110 1 E 14
$w\bar{x}$	1000 1 8 8	1001 0 9 9	1011 0 B 11	1010 0 A 10

More Karnaugh Map Terminology

- **<u>Distinguished 1-cell</u>: A single minterm that can be covered by <u>only one prime implicant</u>.**
- **<u>Essential prime implicant</u>: A prime implicant that covers one or more <u>distinguished 1-cells</u>.**
- **Note: <u>Every fully minimized Boolean expression</u> must include <u>all</u> of the essential prime implicants of f.**
- **In the K-map at right, the Boolean minterm f = $wxy\bar{z}$ is a distinguished 1-cell, and the essential prime implicant f = wxy is the only prime implicant that includes it.**



“Prime Implicants”

- As noted two slides back, a “prime implicant” is the largest square or rectangular implicant of cells occupied by a 1 (SOP) or 0 (POS). Thus a prime implicant will have 1, 2, 4, or 8 cells (16 is a trivial prime).
- How do we determine the Boolean expression for a prime implicant?
- The Boolean expression for an SOP prime implicant is determined by creating a new minterm whose only variables are those that <u>do NOT change value (0→1 or 1→0) over the extent of the prime implicant</u>.
- Thus the prime implicant at right may be represented by the Boolean expression: $f = xz$.

Since x & z do not change value over the implicant, they are the variables in the new Boolean minterm.

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 1 5 5	0111 1 7 7	0110 6 6
wx	1100 C 12	1101 1 D 13	1111 1 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

<u>Example of a prime implicant</u>

Logic Simplification – an SOP Example

- We simplify a Boolean expression by finding its prime implicants on a K-Map.
- To do this, populate the K-map as follows:
  - For an SOP expression,* find the K-Map cell for each minterm of function f, and place a one (1) in it. Ignore 0's.
  - Circle groups of cells that contain 1's.
  - The number of cells enclosed in a circle must be a power of 2 and square or rectangular.
  - It is okay for groups of cells to overlap.
  - Each circled group of cells corresponds to a prime implicant of f.
  - Note that the more cells a given circle encloses, the fewer variables needed to specify the implicant!

	$\overline{y}\,\overline{z}$	$\overline{y}z$	yz	$y\overline{z}$
$\overline{w}\,\overline{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\overline{w}x$	0100 1 4 4	0101 1 5 5	0111 1 7 7	0110 1 6 6
wx	1100 C 12	1101 1 D 13	1111 1 F 15	1110 E 14
$w\overline{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

* A POS example will follow.

Using the K Map for Logic Simplification:

- Another example:
  - The implicants of f are 4, 6, 12, 14.
  - This corresponds to the function:

    $f = \bar{w}x\bar{y}\bar{z} + \bar{w}xy\bar{z} + wx\bar{y}\bar{z} + wxy\bar{z}$
  - We create a <u>prime implicant</u> by grouping the 4 cells representing f.
  - On the wx axis, the cells are 1 whether or not w is one, and always when x is 1 (w not needed).
  - On the yz axis, the cells are 1 whether or not y is one, and always when z is 0 (y not needed).
  - Thus the simplified expression for f is: $f = x\bar{z}$.

	yz	yz	yz	yz
$\bar{w}\bar{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 1 4 4	0101 5 5	0111 7 7	0110 1 6 6
wx	1100 1 C 12	1101 D 13	1111 F 15	1110 1 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

Remember: For purposes of grouping implicants, <u>the top and bottom rows of the K-map are considered adjacent, as are the right and left columns</u>.
This grouping takes advantage of the fact that the left and right columns are adjacent.

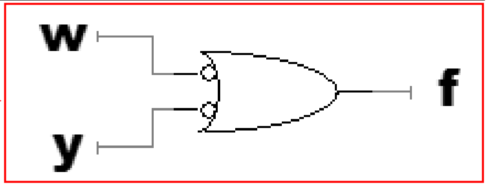
A POS K-Map

- On a POS K-map, the procedure is the same, except that we map 0’s.
- Let:

$$f = (\overline{w}+\overline{x}+\overline{y}+\overline{z})\cdot(\overline{w}+\overline{x}+\overline{y}+z)$$
$$\cdot(\overline{w}+x+\overline{y}+\overline{z})\cdot(\overline{w}+x+\overline{y}+z)$$

- We find prime implicants exactly the same way – except that we look for variable that produce 0’s.
- As shown, w and y do not change over the extent of the function.
- The simplified expression is: $f = (\overline{w} + \overline{y})$. The simplified circuit is shown at right.

	$y+z$	$y+\overline{z}$	$\overline{y}+\overline{z}$	$\overline{y}+z$
$w+x$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$w+\overline{x}$	0100 4 4	0101 5 5	0111 7 7	0110 6 6
$\overline{w}+\overline{x}$	1100 C 12	1101 D 13	1111 0 F 15	1110 0 E 14
$\overline{w}+x$	1000 8 8	1001 9 9	1011 0 B 11	1010 0 A 10



Summary of Karnaugh Map Procedure

- **In summary, to simplify a Boolean expression using a K-Map:**
  1. **Start with the truth table or Boolean expression, if you have one.**
  2. **If starting from the truth table, write the Boolean expression for each truth table term that is 1 (if SOP) or 0 (if POS).**
  3. **(Develop the full Boolean expression, if necessary, by OR-ing the AND terms together if SOP or AND-ing OR terms if POS.)**
  4. **On the K-Map, plot 1's (for SOP) or 0's (for POS).**
  5. **Group implicants together to get the largest set of prime implicants possible. Prime implicants may overlap each other. They will always be square or rectangular groups of cells that are powers of 2 (1, 2, 4, 8).**
  6. **The variables that make up the term(s) of the new expression will be those which do not vary in value over the extent of each prime implicant.**
  7. **Write the Boolean expression for each prime implicant and then OR (for SOP) or AND (for POS) terms together to get the new expression.**

K-Map Example of Prime Implicants

- The Boolean expression represented is:

$$f = \overline{w}x\overline{y}z + \overline{w}xyz + \overline{w}xy\overline{z} + wx\overline{y}z$$

$$+wxyz + wxy\overline{z} + w\overline{x}\,\overline{y}z + w\overline{x}yz$$

- Note that since prime implicants must be powers of 2, the largest group of squares we can circle is 4.
- Thus we circle three groups of 4 (in red). <u>The circles may overlap</u>.
- We now write the new SOP simplified expression. It is:

$$f = xy + xz + wz$$

	$\overline{y}\,\overline{z}$	$\overline{y}z$	yz	$y\overline{z}$
$\overline{w}\,\overline{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\overline{w}x$	0100 4 4	0101 1 5 5	0111 1 7 7	0110 1 6 6
wx	1100	1101 1 13	1111 1 5	1110 E 1 14
$w\overline{x}$	1000 8	1001 1 9	1011 1 1	1010 A 10

NOT a prime implicant!

Another SOP Minimization, Given the Truth

w	x	y	z	f
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	1
0	1	0	1	1
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	1
1	1	0	1	1
1	1	1	0	
1	1	1	1	

Minterms Plotted on K-Map

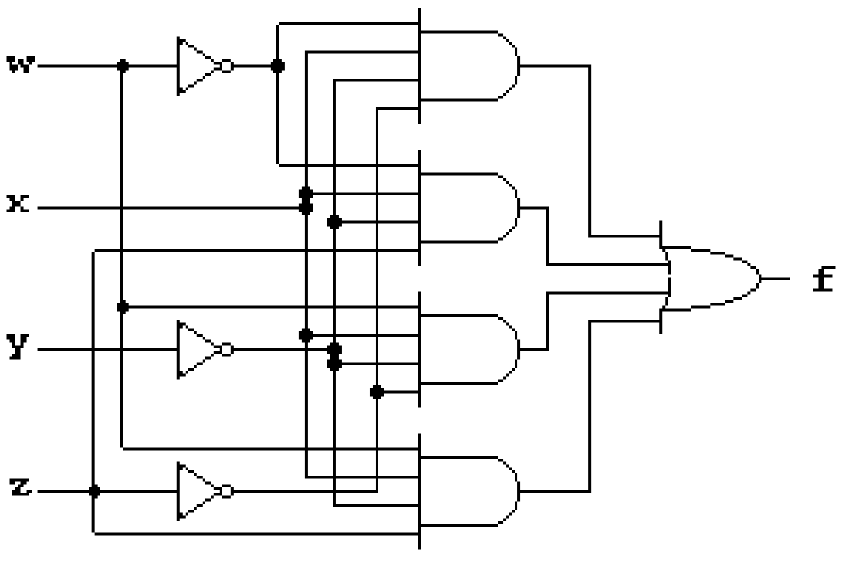
- The four minterms are plotted on the truth table.
- Note that they easily group into one prime implicant.
- Over the extent of the prime implicant, variables w & z vary, so they cannot be in the the Boolean expression for the prime implicant.
- Variables x and y do not vary.
- Thus the expression for the minimum SOP representation must be $f = x\bar{y}$.

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 1 4 4	0101 1 5 5	0111 7 7	0110 6 6
wx	1100 1 C 12	1101 1 D 13	1111 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

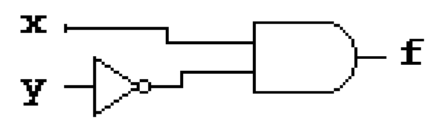
The original Boolean expression is:

$$f = \bar{w}x\bar{y}\bar{z} + \bar{w}x\bar{y}z + wx\bar{y}\bar{z} + wx\bar{y}z$$

Original and Simplified Circuits



Logic circuit from truth table.



Equivalent circuit after reduction using Karnaugh map.

Exercise 3

- The truth table below was developed from a “spec.” Show the SOP expression and then minimize it using a K-map and draw the minimized circuit.

x	y	z	f
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{x}$	000 0	001 1	011 3	010 2
x	100 4	101 5	111 7	110 6

Another Example: Biggest Prime Implicants

w	x	y	z	f
0	0	0	0	
0	0	0	1	1
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	1
0	1	1	0	
0	1	1	1	1
1	0	0	0	
1	0	0	1	1
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	1
1	1	1	0	
1	1	1	1	1

Minimization Using Four-Variable K-Map

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 1 5 5	0111 1 7 7	0110 6 6
wx	1100 C 12	01 1 D 13	1111 1 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 1 9 9	1011 B 11	1010 A 10

Incorrect solution (partial minimization): $f = xyz + \bar{y}z$

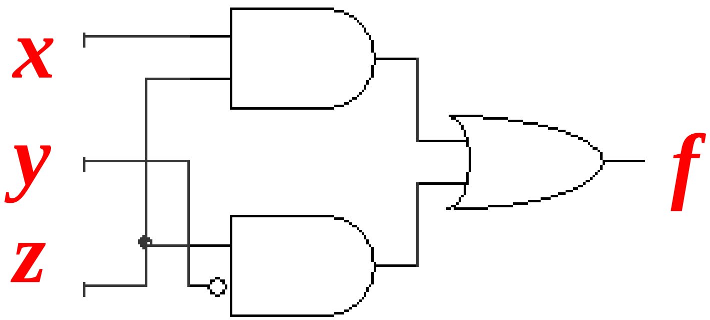
	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	01 1 5 5	0111 1 7 7	0110 6 6
wx	1100 C 12	01 1 D 13	1111 1 F 15	1110 E 14
$w\bar{x}$	1000 8 8	01 1 9 9	1011 B 11	1010 A 10

Correct Solution:

$$f = xz + \bar{y}z$$

<u>Remember: Prime implicants should overlap, if this means that they can be made larger</u>.

Resulting Circuit



Note that the resulting circuit uses 3 logic gates, whereas the original expression, with six minterms, would have used a minimum of seven gates and four inverters.

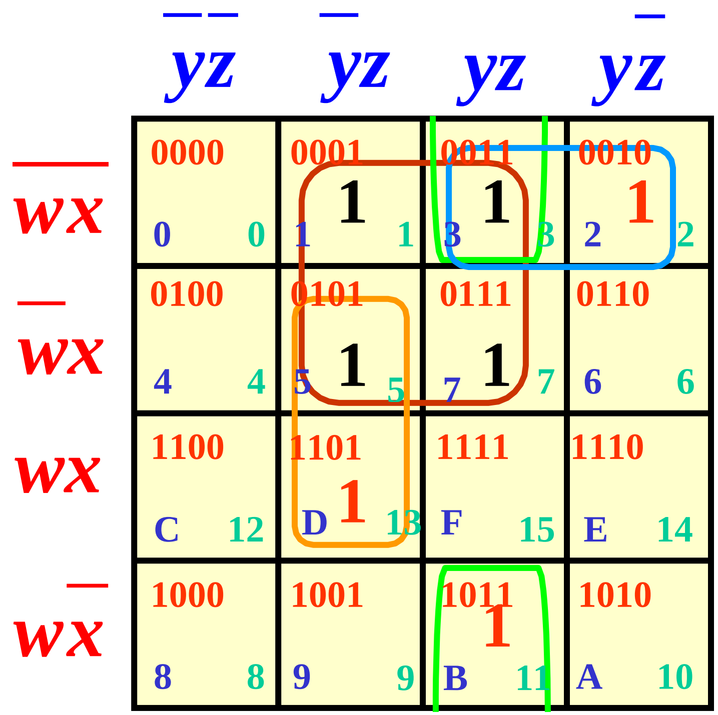
Sometimes Major Simplification is Not Possible

- Using the Σ notation:

  $f = \Sigma m(1,2,3,5,7,11,13)$

- Note that there are 3 distinguished one-cells (in red).
- There must therefore be at least 3 prime implicants (4 in this case, 3 essential).
- The simplified expression (and not very simplified at that) is:

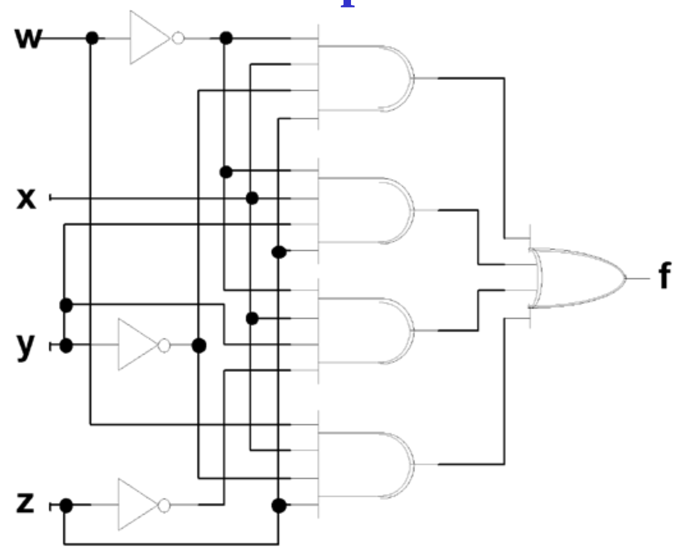
$$f = \bar{w}z + \bar{w}\bar{x}y + x\bar{y}z + \bar{x}yz$$



Exercise 4

A truth table and its Boolean expression are shown below, along with the circuit of this unsimplified expression. Use the K-map on the following page to simplify and draw the simplified circuit.

w	x	y	z	f
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	1
1	1	1	0	
1	1	1	1	



Original Circuit

$$f = \bar{w}x\bar{y}z + \bar{w}\bar{x}yz + \bar{w}xyz + wx\bar{y}z$$

Karnaugh Map of Last Example

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 5 5	0111 7 7	0110 6 6
wx	1100 C 12	1101 D 13	1111 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

The Concept of "Don't Cares"

- The six implicants on the K-Map shown can be represented by the simplified expression $f = x\,y + x\,z$.
- Suppose for our particular logic system, we know that *y* and *z* will <u>never be 0 together</u>.
- Then the $\bar{y}\bar{z}$ implicants do not matter, <u>since they cannot happen</u>.
- To show the condition cannot occur, we put X's in the column -- they are "don't cares" -- conditions that cannot happen.

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 X 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 X 4 4	0101 1 5 5	0111 1 7 7	0110 1 6 6
wx	1100 X C 12	1101 1 D 13	1111 1 F 15	1110 1 E 14
$w\bar{x}$	1000 X 8 8	1001 9 9	1011 B 11	1010 A 10

"Don't Cares" (2)

- Since the function will never get into the left column, we "don't care" how those minterms are represented. Why not make them 1's?
- Doing that, we can make a much larger prime implicant and a simpler Boolean expression. For this implicant, $f = x$.
- The expression is valid, since the forbidden condition will never allow the two left squares to be occupied.
- "Don't cares" let us further simplify an expression.

	$\overline{yz}$	$\bar{y}z$	yz	$y\bar{z}$
$\overline{wx}$	0000 1 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 1 4 4	0101 1 5 5	0111 1 7 7	0110 1 6 6
wx	1100 1 C 12	1101 1 D 13	1111 1 F 15	1110 1 E 14
$w\bar{x}$	1000 1 8 8	1001 9 9	1011 B 11	1010 A 10

"Don't Cares" – Another Example

- Assume the Boolean expression is as shown on the K-map (black 1's and black prime implicants).
- The simplest SOP expressions for the function is $f = \bar{w}xz + wxy$.
- Also assume that $wx\bar{y}\bar{z}$ and $wx\bar{y}z$ cannot occur.
- Since these cannot ever happen, they are "don't cares," and since they are "don't cares," make them 1 (<u>red</u>)!
- We can then further simplify the expression to $f = wx + xz$ (larger prime implicants).

	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 1 5 5	0111 1 7 7	0110 6 6
wx	1100 1 C 12	1101 1 D 13	1111 1 F 15	1110 1 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10

“Don’t Cares” -- Summary

- **“Don’t Cares” occur when there are variable combinations, represented by squares on a Karnaugh map, <u>that cannot occur in a digital circuit or Boolean expression</u>.**
- **Such a square is called a “Don’t Care.” Since it can never happen, we can assign a value of <u>1</u> to the square and use that 1 to (possibly) construct larger prime implicants, further simplifying the circuit (you could assign it the value 0 in a POS representation).**
- **Circuits or Boolean expressions derived using “Don’t Care” squares are as valid as any other expression or circuit.**
- **We will see later in sequential logic that the concept of “Don’t Cares” will help in simplifying counter circuits.**

Exercise 5

- n SOP Boolean expression is defined as $f=\Sigma_m(2, 6, C, D, F, E)$.
- The inputs are such that w is never 1 when $x = 0$.
- Find the simplified expression and draw the simplified circuit.

	yz	yz	yz	$y\bar{z}$
$\bar{w}\bar{x}$	0000 0 0	0001 1 1	0011 3 3	0010 2 2
$\bar{w}x$	0100 4 4	0101 5 5	0111 7 7	0110 6 6
wx	1100 C 12	1101 D 13	1111 F 15	1110 E 14
$w\bar{x}$	1000 8 8	1001 9 9	1011 B 11	1010 A 10