Standard Forms of Expression

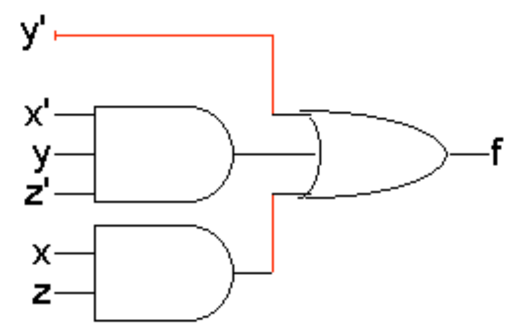
Minterms and Maxterms

Standard forms of expressions

- We can write expressions in many ways, but some ways are more useful than others
- A sum of products (SOP) expression contains:
  - Only OR (sum) operations at the "outermost" level
  - Each term that is summed must be a product of literals

$$f(x,y,z) = y' + x'yz' + xz$$

- The advantage is that any sum of products expression can be implemented using a two-level circuit
  - literals and their complements at the "0th" level
  - AND gates at the first level
  - a single OR gate at the second level
- This diagram uses some shorthands…
  - NOT gates are implicit
  - literals are reused
  - this is **not** okay in LogicWorks!



Minterms

- A minterm is a special product of literals, in which each input variable appears exactly once.
- A function with n variables has 2^n minterms (since each variable can appear complemented or not)
- A three-variable function, such as f(x,y,z), has $2^3 = 8$ minterms:

x'y'z'	x'y'z	x'yz'	x'yz
xy'z'	xy'z	xyz'	xyz

- Each minterm is true for exactly one combination of inputs:

Minterm	Is true when…	Shorthand
x'y'z'	x=0, y=0, z=0	m_0
x'y'z	x=0, y=0, z=1	m_1
x'yz'	x=0, y=1, z=0	m_2
x'yz	x=0, y=1, z=1	m_3
xy'z'	x=1, y=0, z=0	m_4
xy'z	x=1, y=0, z=1	m_5
xyz'	x=1, y=1, z=0	m_6
xyz	x=1, y=1, z=1	m_7

Sum of minterms form

- Every function can be written as a sum of minterms, which is a special kind of sum of products form
- The sum of minterms form for any function is **unique**
- If you have a truth table for a function, you can write a sum of minterms expression just by picking out the rows of the table where the function output is 1.

x	y	z	f(x,y,z)	f'(x,y,z)
0	0	0	1	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	0	1
1	1	0	1	0
1	1	1	0	1

$$f = x'y'z' + x'y'z + x'yz' + x'yz + xyz'$$
$$= m_0 + m_1 + m_2 + m_3 + m_6$$
$$= \Sigma m(0,1,2,3,6)$$

$$f' = xy'z' + xy'z + xyz$$
$$= m_4 + m_5 + m_7$$
$$= \Sigma m(4,5,7)$$

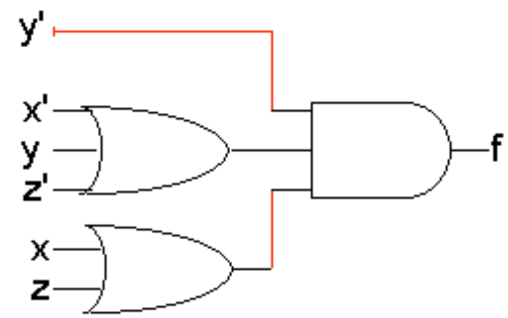
f' contains all the minterms not in f

The dual idea: products of sums

- Just to keep you on your toes...
- A product of sums (POS) expression contains:
  - Only AND (product) operations at the "outermost" level
  - Each term must be a sum of literals

$$f(x,y,z) = y' (x' + y + z') (x + z)$$

- Product of sums expressions can be implemented with two-level circuits
  - literals and their complements at the "0th" level
  - **OR gates** at the first level
  - a single **AND gate** at the second level
- Compare this with sums of products



Maxterms

- A maxterm is a **sum** of literals, in which each input variable appears exactly once.
- A function with n variables has 2^n maxterms
- The maxterms for a three-variable function f(x,y,z):

$x' + y' + z'$ $\quad x' + y' + z$ $\quad x' + y + z'$ $\quad x' + y + z$

$x + y' + z'$ $\quad x + y' + z$ $\quad x + y + z'$ $\quad x + y + z$

- Each maxterm is **false** for exactly one combination of inputs:

Maxterm	Is **false** when…	Shorthand
$x + y + z$	x=0, y=0, z=0	M_0
$x + y + z'$	x=0, y=0, z=1	M_1
$x + y' + z$	x=0, y=1, z=0	M_2
$x + y' + z'$	x=0, y=1, z=1	M_3
$x' + y + z$	x=1, y=0, z=0	M_4
$x' + y + z'$	x=1, y=0, z=1	M_5
$x' + y' + z$	x=1, y=1, z=0	M_6
$x' + y' + z'$	x=1, y=1, z=1	M_7

Product of maxterms form

- Every function can be written as a **unique** product of maxterms
- If you have a truth table for a function, you can write a product of maxterms expression by picking out the rows of the table where the function output is 0. (Be careful if you're writing the actual literals!)

x	y	z	f(x,y,z)	f'(x,y,z)
0	0	0	1	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	0	1
1	1	0	1	0
1	1	1	0	1

$$f = (x' + y + z)(x' + y + z')(x' + y' + z')$$
$$= M_4\ M_5\ M_7$$
$$= \Pi M(4,5,7)$$

$$f' = (x + y + z)(x + y + z')(x + y' + z)$$
$$(x + y' + z')(x' + y' + z)$$
$$= M_0\ M_1\ M_2\ M_3\ M_6$$
$$= \Pi M(0,1,2,3,6)$$

f' contains all the maxterms not in f

Minterms and maxterms are related

- Any minterm m_i is the **complement** of the corresponding maxterm M_i

Minterm	Shorthand	Maxterm	Shorthand
$x'y'z'$	m_0	$x + y + z$	M_0
$x'y'z$	m_1	$x + y + z'$	M_1
$x'yz'$	m_2	$x + y' + z$	M_2
$x'yz$	m_3	$x + y' + z'$	M_3
$xy'z'$	m_4	$x' + y + z$	M_4
$xy'z$	m_5	$x' + y + z'$	M_5
xyz'	m_6	$x' + y' + z$	M_6
xyz	m_7	$x' + y' + z'$	M_7

- For example, $m_4' = M_4$ because $(xy'z')' = x' + y + z$

Converting between standard forms

- We can convert a sum of minterms to a product of maxterms

From before	f	$= \Sigma m(0,1,2,3,6)$	
and	f'	$= \Sigma m(4,5,7)$	
		$= m_4 + m_5 + m_7$	
complementing	$(f')'$	$= (m_4 + m_5 + m_7)'$	
so	f	$= m_4' \, m_5' \, m_7'$	[DeMorgan's law]
		$= M_4 \, M_5 \, M_7$	[By the previous page]
		$= \Pi M(4,5,7)$	

- In general, just replace the minterms with maxterms, using maxterm numbers that don't appear in the sum of minterms:

$$f = \Sigma m(0,1,2,3,6)$$
$$= \Pi M(4,5,7)$$

- The same thing works for converting from a product of maxterms to a sum of minterms

Summary so far

- So far:
  - A bunch of Boolean algebra trickery for simplifying expressions and circuits
  - The algebra guarantees us that the simplified circuit is **equivalent** to the original one
  - Introducing some standard forms and terminology
- Next:
  - An alternative simplification method
  - We'll start using all this stuff to build and analyze bigger, more useful, circuits

Product of Sums

i>clicker

- If f(x, y, z) = sum of minterms (0, 1, 4, 5), represent f as a product of maxterms
  - A: product of maxterms(2, 3)
  - B: product of maxterms(2, 3, 6, 7)
  - C: product of maxterms(0, 1, 4, 5)
  - D: product of maxterms(5, 6, 7)

i>clicker

- If f(x, y, z) = sum of minterms (0, 1, 4, 5), represent f' as a product of maxterms
  - A: product of maxterms(2, 3)
  - B: product of maxterms(2, 3, 6, 7)
  - C: product of maxterms(0, 1, 4, 5)
  - D: product of maxterms(5, 6, 7)