11.11 Sum-of-minterms form

Sum-of-minterms

Different equations may represent the same function. Ex: y = a + b, and y = a + a'b, represent the same function. The sameness is not obvious, so a standard equation form is desirable.

- A ***canonical form*** of a Boolean equation is a standard equation form for a function.
- ***Sum-of-minterms*** form is a canonical form of a Boolean equation where the right-side expression is a sum-of-products with each product a unique minterm.
- A ***minterm*** is a product term having exactly one literal for every function variable.
- A ***literal*** is a variable appearance, in true or complemented form, in an expression, such as b, or b'.

For a function of variables a and b, y = ab + a'b + a'b' is in sum-of-minterms form, but y = ab + a' is not because the second product term is missing variable b.

PARTICIPATION ACTIVITY 11.11.1: Minterms.

Given a function of a, b, c.

1) Does abc have 3 literals?
   - Yes
   - No

2) Does ab'c have 4 literals?
   - Yes
   - No

3) Is bc' a product term?
   - Yes
   - No

4) Is bc' a minterm?
   - Yes
   - No

5) Is ab'c a minterm?
   - Yes
   - No

6) Is a(b + c') a minterm?
   - Yes
   - No

PARTICIPATION ACTIVITY 11.11.2: Sum-of-minterms form.

Given a function of a, b, c, indicate if the equation is in sum-of-minterms form.

1) y = abc + a'b'c'
   - Yes
   - No

2) y = ab + abc
   - Yes
   - No

3) y = a(b + c)
   - Yes
   - No

4) y = abc
   - Yes
   - No

5) y = ac
   - Yes
   - No

6) y = abc + cb'a
   - Yes
   - No

7) y = abc + abc
   - Yes
   - No

Transforming to sum-of-minterms

A sum-of-products equation can be transformed to sum-of-minterms by multiplying each product term by (v + v') for any missing variable v to create minterm (removing redundant minterms). v + v' is 1, so multiplying a term by (v + v') doesn't change a product term's functionality. Ex:

y = ab + a'	sum-of-products, but not sum-of-minterms
y = ab + a'(b + b')	
y = ab + a'b + a'b'	sum-of-minterms

An equation not initially in sum-of-products form can first be multiplied out. Thus, transforming an equation to sum-of-minterms is done by:

- Initially multiplying out to sum-of-products
- Transform each product term to a minterm
- Remove redundant minterms

PARTICIPATION ACTIVITY 11.11.3: Transforming to sum-of-minterms.

Start 2x speed

Given variables a, b, c. Convert y = a(b + bc') to sum-of-minterms.

y = a(b + bc')
= ab + abc'
= ab (1) + abc'
= ab (c + c') + abc'
= abc + abc' + abc'
= abc + abc'

PARTICIPATION ACTIVITY 11.11.4: Transforming an equation already in sum-of-products form to sum-of-minterms.

Given variables a, b. Order the steps to transform y = ab + a' to sum of minterms.

y = ab + a'(b + b')   y = ab + a'b + a'b'   y = ab + a'(1)   y = ab + a'

Original equation

(1)

(2)

(3)

Reset

PARTICIPATION ACTIVITY 11.11.5: Transforming a general equation to sum-of-minterms form.

Given variables a, b, c. Order the steps to transform y = (a + c)b to sum-of-minterms.

y = ab + bc   y = (a + c)b   y = ab(c + c') + bc(a + a')   y = ab(1) + bc(1)

y = a'bc + abc' + abc   y = abc + abc' + abc + a'bc

Original equation

(1)

(2)

(3)

(4)

(5)

Reset

Note: Transforming directly from ab to ab(c + c') is a common shortcut. The intermediate step, ab to ab(1), is often omitted.

PARTICIPATION ACTIVITY 11.11.6: Transforming to sum-of-minterms form.

Given variables a, b, c. Transform each equation to sum-of-minterms form. Type only the ? part

1) y = a'b
y = a'b(1)
y = a'b(c + c')
y = a'bc + ?

Check Show answer

2) y = ac
y = ac(1)
y = ac(b + ?)

Check Show answer

3) y = a'c'
y = a'c'(b + ?)

Check Show answer

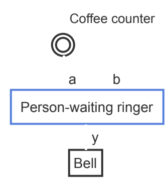
Example: Determining if two equations represent the same function

Because sum-of-minterms is canonical, one can determine whether two equations represent the same function by transforming each to sum-of-minterms equations and checking if the equations are the same.

PARTICIPATION ACTIVITY 11.11.7: Transforming to sum-of-minterms to check if equations represent same function: Person-waiting example.

Start 2x speed

y = a + a'b	y = a + b
y = a(1) + a'b	y = a(1) + b(1)
y = a(b + b') + a'b	y = a(b + b') + b(a + a')
y = ab + ab' + a'b	y = ab + ab' + ba + ba'
y = a'b + ab' + ab	y = ab + ab' + ab + a'b
	y = ab + ab' + a'b
	y = a'b + ab' + ab



Compact function notation: Minterm numbers

A compact function notation represents each minterm by a number. Given that ab'c is 1 if a/b/c are 1/0/1, that minterm is represented as m_5 because 101 in binary is 5 in decimal. A 3-variable function thus has minterms numbered 0 to 7. Ex: f(a,b,c) = a'b'c' + ab'c + abc can be written compactly as f(a,b,c) = m0 + m5 + m7. An alternative notation is f(a,b,c) = ∑(0, 5, 7).

PARTICIPATION ACTIVITY 11.11.8: Numbered minterms.

Match the minterms.

m2 m1 m5 m6 m0

a'bc'

a'b'c'

a'b'c

ab'c

abc'

Reset

PARTICIPATION ACTIVITY 11.11.9: Compact function notation.

1) Given f(a, b, c) = a'bc + abc, the compact notation is: f(abc) = ?

- m3 + m7
- m4 + m0
- Cannot determine

2) Given f(a, b, c) = ab, the compact notation is: f(abc) = ?

- m3
- m6 + m7
- Cannot determine

The compact form makes comparing equations for equivalence especially easy.

CHALLENGE ACTIVITY 11.11.1: Transform the equation to a sum-of-minterms.

Start

Expand w'x' to sum-of-minterms form w'x'z+w'x'z'

Properties

Distributive			Identit
ab+ac	=	a(b+c)	a·1
(a+b)(a+c)	=	a+bc	a+0
Commutative			**Null el**
ab	=	ba	a·0
a+b	=	b+a	a+1
Complement			**Idemp**
aa'	=	0	aa
a+a'	=	1	a+a

1 2 3

Check Next

Provide feedback on this section