Standard Boolean Forms

In this section, we develop the idea of standard forms of Boolean expressions. In part, these forms are based on some standard Boolean simplification rules.

Standard forms are either **canonical forms** or **normal forms**.

The standard expressions are in either

SOP   **S**um **of** **P**roducts form, or

POS   **P**roduct **of** **S**ums form.

This lecture will focus on the following:

Canonical Sum of Products

Normal Sum of Products

Canonical Product of Sums

Normal Product of Sums

We shall also discuss a few more variants that have no standard names.

IMPORTANT: These forms use only the 3 basic Boolean functions: AND, OR, NOT. Specifically, XOR is not used.

Variables and Literals

We start with the idea of a Boolean variable. It is a simple variable that can take one of only two values: 0 (False) or 1 (True).

Following standard digital design practice, we use the values 0 and 1.

Following standard teaching practice, we denote all Boolean variables by single letters; normally "A", "B", "C", "D", or "W", "X", "Y", "Z".

A literal is either a Boolean variable or its complement.

Literals based on the variable X: X and $\overline{X}$.

Literals based on the variable Y: Y and $\overline{Y}$.

NOTE: X and $\overline{X}$ represent the same variable,
       but they are not the same literal.

       X and $\overline{Y}$ represent different variables.

Product and Sum Terms

A **product term** is the logical AND of one or more literals,
   with no variable represented more than once.

A **sum term** is the logical OR of one or more literals,
   with no variable represented more than once.

The following are all valid product terms over the two variables X and Y.

$$\overline{X} \bullet \overline{Y} \qquad \overline{X} \bullet Y \qquad X \bullet \overline{Y} \qquad X \bullet Y$$

Forms, such as $X \bullet X \bullet \overline{Y}$ and $X \bullet \overline{X} \bullet Y$ are not considered, as $X \bullet X = X$ and $X \bullet \overline{X} = 0$, so $X \bullet X \bullet \overline{Y} = X \bullet \overline{Y}$ and $X \bullet \overline{X} \bullet Y = 0 \bullet Y = 0$.

The following are all valid sum terms over the two variables X and Y.

$$X + Y \qquad X + \overline{Y} \qquad \overline{X} + Y \qquad \overline{X} + \overline{Y}$$

Single literals
According to the strict definition, a single literal is either a sum term or a product term, depending on the context. This is necessary to avoid having to give a number of special cases in the following definitions.

Sum of Products and Product of Sums

A **SOP** (Sum of Products) expression is the logical OR of product terms.

A **POS** (Product of Sums) expression is the logical AND of sum terms.

Sample SOP expressions

$$F1(X, Y) \qquad = X \bullet Y + \overline{X} \bullet \overline{Y}$$

$$G1(X, Y) \qquad = \overline{X} \bullet Y + X \bullet \overline{Y}$$

$$H1(X, Y, Z) \qquad = X + \overline{Y} \bullet Z$$

Note: If we did not allow single literals to be product terms, we would have trouble classifying H(X, Y, Z), which is clearly SOP.

Sample POS expressions

$$F2(X, Y) \qquad = (X+Y) \bullet (\overline{X}+\overline{Y})$$

$$G2(X, Y) \qquad = (\overline{X}+Y) \bullet (X+\overline{Y})$$

$$H2(X, Y, Z) \qquad = X \bullet (\overline{Y} + Z)$$

Note: POS expressions almost always have parentheses to indicate the correct evaluation.

More on Ambiguous Forms

What is the form of the expression $F(X, Y) = X + Y$

1. SOP   It is the logical OR of two product terms.
            Each product term is a single literal.

2. POS   It is a single sum term $(X + Y)$

Both statements are true. In general, questions such as this do not concern us.
If you are asked a question like this on a test, either answer will be accepted.

This ambiguity comes from the definitional necessity of mentioning "the logical AND of one or more terms" and "the logical OR of one or more terms".

With two equally good answers to an ambiguous form, pick the one you like.

Inclusion

A product term T_1 is included in a product term T_2
   if every literal in T_1 is also in T_2.

A sum term T_1 is included in a sum term T_2
   if every literal in T_1 is also in T_2.

Consider the following examples:

F(A, B, C) = A•B + A•C + A•B•C
         Each of A•B and A•C is included in A•B•C.

G(A, B, C) = (A + B)•(A + C)•(A + B + C)
         Each of (A + B) and (A + C) is included in (A + B + C).

There is no inclusion in the next expression

F(A, B, C) = A•B + A•C + $\overline{A}$•B•C
         The literal A does no appear in the third term.

The inclusion rule is based on literals, not just variables.

More on Inclusion

Consider $\quad$ F1(A, B, C) = A•B + A•C + A•B•C

and $\qquad$ F2(A, B, C) = A•B + A•C

We claim that the two are equal for every value of A, B, and C.

Let A = 0

$\qquad$ Clearly $\;$ F1(A, B, C) = 0
$\qquad\qquad\qquad$ F2(A, B, C) = 0

Let A = 1

$\qquad$ Then $\qquad$ F1(A, B, C) = B + C + B•C
$\qquad$ and $\qquad$ F2(A, B, C) = B + C

Notice that we still have inclusion in F1, as each of B and C is included in B•C.
We prove these versions of F1(A, B, C) = F2(A, B, C) using a truth table.

B	C	B•C	B + C	B + C + B•C
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	1

Last Word on Inclusion

If a SOP or POS expression has included terms, it can be simplified.

$F1(A, B, C) = A \bullet B + A \bullet C + A \bullet B \bullet C$ is identically equal to
$F2(A, B, C) = A \bullet B + A \bullet C$

$G1(A, B, C) = (A + B) \bullet (A + C) \bullet (A + B + C)$ is identically equal to
$G2(A, B, C) = (A + B) \bullet (A + C)$

The conclusion is that Boolean expressions with included terms are needlessly complicated. We can simplify them by the application of trivial rules.

Note that duplication is a form of inclusion.

The expression $F3(A, B) = A \bullet B + A \bullet B$ has 2 terms, each included in the other.

Non–Standard Expressions

Not every useful Boolean expression is in a standard form.

$F(X, Y) = X \oplus Y$ is not a standard form, due to the exclusive OR.

$G(X, Y) = X \bullet Y + (X + Y) \bullet (\overline{X} + Y)$ is not in a standard form.
     This has both a product term and a sum term.

The fact that $G(X, Y)$ can easily be converted to a standard form
    does not make it already in a standard form.

Let's convert this to SOP. I usually have difficulty in conversion to POS,
    unless I am using a method I have yet to describe.

The term $X \bullet Y$ is already a product term, so we convert $(X + Y) \bullet (\overline{X} + Y)$ to SOP.

$$(X + Y) \bullet (\overline{X} + Y) = X \bullet (\overline{X} + Y) + Y \bullet (\overline{X} + Y)$$
$$= X \bullet \overline{X} + X \bullet Y + Y \bullet \overline{X} + Y \bullet Y$$
$$= 0 + X \bullet Y + \overline{X} \bullet Y + Y \qquad = (X + \overline{X}) \bullet Y + Y$$
$$= 1 \bullet Y + Y = Y$$

So $G(X, Y) = X \bullet Y + Y = Y$ **$G(X, Y) = Y$**.

More on Non–Standard Forms

Look at the Boolean function G(X, Y) = X•Y + Y. There are two ways to at look at this. Try both ways.

Let G(X, Y) = X•Y + Y = X•Y + 1•Y = (X + 1) •Y = 1 •Y = Y

Do a truth table proof of the equality.

X	Y	X•Y	X•Y + Y
0	0	0	0
0	1	0	1
1	0	0	0
1	1	1	1

Note that the column marked X•Y matches the one marked X•Y + Y. The two functions are identical. Also note that the term Y is included in the term X•Y, so that, by inclusión, the term X•Y can be eliminated from the expression.

Normal and Canonical Forms

A **Normal SOP** expression is a Sum of Products expression
with no included product terms.

A **Normal POS** expression is a Product of Sums expression
with no included sum terms.

A **Canonical SOP** expression over a set of Boolean variables is
a Normal SOP expression in which each product term contains
a literal for each of the Boolean variables.

A **Canonical POS** expression over a set of Boolean variables is
a Normal POS expression in which each sum term contains
a literal for each of the Boolean variables.

Note: A canonical expression on N Boolean variables is made up
of terms, each of which has exactly N literals.

Note: One can do digital design based on either normal or canonical forms.
The choice usually depends on the technology used.

Equivalence of Canonical Forms and Truth Tables

We can directly translate between either of the canonical forms and
a truth table using a standard set of rules.

To produce the **Sum of Products** representation from a truth table,
   a) Generate a product term for each row where the value of the function is 1.
   b) The variable is complemented if its value in the row is 0, otherwise it is not.

To produce the **Product of Sums** representation from a truth table,
   a) Generate a sum term for each row where the value of the function is 0.
   b) The variable is complemented if its value in the row is 1, otherwise it is not.

Row	X	Y	X $\oplus$ Y
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

SOP: Terms for rows 1 and 2. Row 1: $\overline{X} \bullet Y$, Row 2: $X \bullet \overline{Y}$ $F = \overline{X} \bullet Y + X \bullet \overline{Y}$

POS: Terms for rows 0 and 3. Row 0: $(X + Y)$, Row 3: $(\overline{X} + \overline{Y})$
         $F = (X + Y) \bullet (\overline{X} + \overline{Y})$

Examples: Conversions between the Three Forms

We have three equivalent ways to define a Boolean expression.

1. The Truth Table.

2. The Σ and Π lists.

3. The Canonical Form.

In each of these depictions of the expression, we need to know the number of Boolean variables and labels to be assigned these variables.

It is easier to consider the SOP and POS cases separately, because the rules for conversion from the truth tables are so different for the two cases.

The figure below depicts the translations we shall consider for the SOP case.
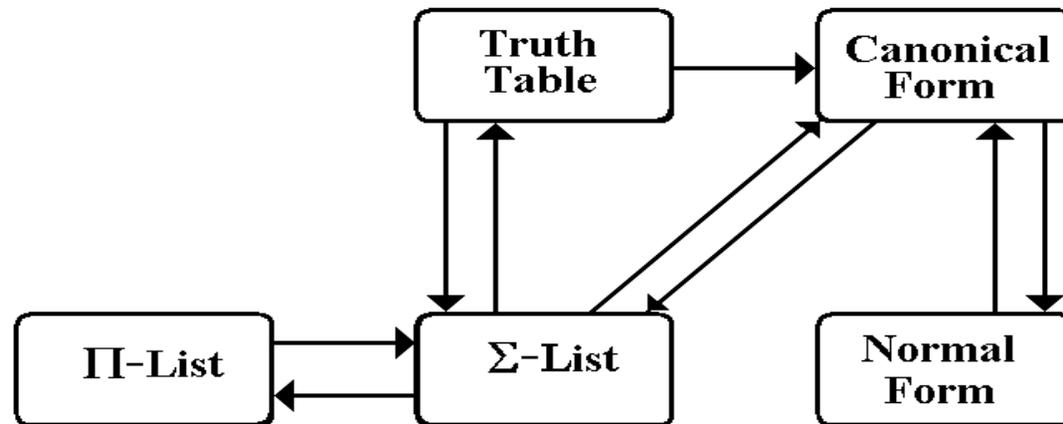
Example Truth Table

Here is a truth table definition of a Boolean function of three Boolean variables.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

We shall discuss this function in both of its Sum of Product and
    Product of Sum representations.

We begin with SOP, the form that most students find easier.

SOP Example: Truth Table to Σ–List

Here is the truth table.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Note that the function has value 1 in rows 3, 5, 6, and 7.

The function is $F2(A, B, C) = \Sigma(3, 5, 6, 7)$.

SOP Example: Σ–List to Truth Table

The function is F2(A, B, C) = Σ(3, 5, 6, 7).

Create a truth table and place logic 1's in rows 3, 5, 6, and 7.

Row	A	B	C	F2
0	0	0	0	
1	0	0	1	
2	0	1	0	
3	0	1	1	1
4	1	0	0	
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Place 0's in the other rows.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

SOP Example: Between Σ–List and Π–List

Any function of N Boolean variables is represented by a truth table having 2^N rows, numbered from 0 through $(2^N - 1)$.

The function is F2(A, B, C) = Σ(3, 5, 6, 7).

There will be 2^3 = 8 rows, numbered 0 through 7, in the truth table for F2.

To translate from one list form to the other list form, just pick the numbers in the range 0 … $(2^N - 1)$ that are not in the source list.

F2(A, B, C) = Σ(3, 5, 6, 7). This is missing 0, 1, 2, and 4. So,

F2(A, B, C) = Π(0, 1, 2, 4)

The translation from the Π–list to the Σ–list works in the same way.

SOP Example: Truth Table to Canonical Form

To produce the **Sum of Products** representation from a truth table,
  a) Generate a product term for each row where the value of the function is 1.
  b) The variable is complemented if its value in the row is 0, otherwise it is not.

Here again is the truth table.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

The term is $\overline{A} \bullet B \bullet C$

The term is $A \bullet \overline{B} \bullet C$
The term is $A \bullet B \bullet \overline{C}$
The term is $A \bullet B \bullet C$

$$F2(A, B, C) = \overline{A} \bullet B \bullet C + A \bullet \overline{B} \bullet C + A \bullet B \bullet \overline{C} + A \bullet B \bullet C$$

SOP Example: Between Canonical Form and Σ–List

Given $F2(A, B, C) = \overline{A} \bullet B \bullet C + A \bullet \overline{B} \bullet C + A \bullet B \bullet \overline{C} + A \bullet B \bullet C$

Write a 0 beneath each complemented variable and a 1 below each that is not.

Convert the N–bit numbers, assuming unsigned binary.

These are the rows of the truth table that have a 1.

```
F2(A, B, C)  =  A•B•C  +  A•B•C  +  A•B•C  +  A•B•C
                0 1 1      1 0 1      1 1 0      1 1 1
                  3          5          6          7
```

$F2(A, B, C) = \Sigma(3, 5, 6, 7)$

To convert the other way, just reverse the steps.

SOP Example: Between Canonical and Normal Forms

Given $F2(A, B, C) = \overline{A} \bullet B \bullet C + A \bullet \overline{B} \bullet C + A \bullet B \bullet \overline{C} + A \bullet B \bullet C$

To convert this to a simpler normal form, we must first make it more complex.

$F2(A, B, C) = \overline{A} \bullet B \bullet C + A \bullet B \bullet C$

$\quad\quad\quad\quad + A \bullet \overline{B} \bullet C + A \bullet B \bullet C$

$\quad\quad\quad\quad + A \bullet B \bullet \overline{C} + A \bullet B \bullet C$

But $\quad \overline{A} \bullet B \bullet C + A \bullet B \bullet C = (\overline{A} + A) \bullet B \bullet C = 1 \bullet B \bullet C = B \bullet C$

$\quad\quad\quad A \bullet \overline{B} \bullet C + A \bullet B \bullet C = A \bullet (\overline{B} + B) \bullet C = A \bullet 1 \bullet C = A \bullet C$

$\quad\quad\quad A \bullet B \bullet \overline{C} + A \bullet B \bullet C = A \bullet B \bullet (\overline{C} + C) = A \bullet B \bullet 1 = A \bullet B$

So, $F2(A, B, C) = B \bullet C + A \bullet C + A \bullet B$

$\quad\quad\quad\quad\quad\quad\quad = A \bullet B + A \bullet C + B \bullet C \quad\quad$ The more standard notation.

POS Example: Truth Table to Π–List

Here is the truth table.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Note that the function has value 0 in rows 0, 1, 2, and 4.

The function is F2(A, B, C) = Π(0, 1, 2, 4).

POS Example: Π–List to Truth Table

The function is F2(A, B, C) = Π(0, 1, 2, 4).

Create a truth table and place logic 0's in rows 0, 1, 2, and 4.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	
4	1	0	0	0
5	1	0	1	
6	1	1	0	
7	1	1	1	

Place 1's in the other rows.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

POS Example: Truth Table to Canonical Form

To produce the **Product of Sums** representation from a truth table,
   a) Generate a product term for each row where the value of the function is 0.
   b) The variable is complemented if its value in the row is 1, otherwise it is not.

Here again is the truth table.

Row	A	B	C	F2
0	0	0	0	0
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

The term is $(A + B + C)$
The term is $(A + B + \overline{C})$
The term is $(A + \overline{B} + C)$

The term is $(\overline{A} + B + C)$

$$F2(A, B, C) = (A + B + C) \bullet (A + B + \overline{C}) \bullet (A + \overline{B} + C) \bullet (\overline{A} + B + C)$$

POS Example: Between Canonical Form and Π–List

$F2(A, B, C) = (A + B + C) \bullet (A + B + \overline{C}) \bullet (A + \overline{B} + C) \bullet (\overline{A} + B + C)$

Write a 1 beneath each complemented variable and a 0 below each that is not.

Convert the N–bit numbers, assuming unsigned binary.

```
F2(A, B, C)  =  (A + B + C)•(A + B + C̄)•(A + B̄ + C)•(Ā + B + C)
                 0   0   0    0   0   1     0   1   0    1   0   0
                     0                1             2            4
```

$F2(A, B, C) = \Pi(0, 1, 2, 4)$

To convert the other way, just reverse the steps.

POS Example: Between Canonical and Normal Forms

$F2(A, B, C) = (A + B + C) \bullet (A + B + \overline{C}) \bullet (A + \overline{B} + C) \bullet (\overline{A} + B + C)$

To convert this to a simpler normal form, we must first make it more complex.

$F2(A, B, C) = (A + B + C) \bullet (A + B + \overline{C})$

$\bullet (A + B + C) \bullet (A + \overline{B} + C)$

$\bullet (A + B + C) \bullet (\overline{A} + B + C)$

But $(A + B + C) \bullet (A + B + \overline{C}) = (A + B)$

$(A + B + C) \bullet (A + \overline{B} + C) = (A + C)$

$(A + B + C) \bullet (\overline{A} + B + C) = (B + C)$

So, $F2(A, B, C) = (A + B) \bullet (A + C) \bullet (B + C)$