SUMMARY OF COMBINATIONAL LOGIC

I. Logic Variables

Logic variables take on only two states. The two states are represented by a 1 (logic one) or a 0 (logic zero), although TRUE and FALSE, ON and OFF, HIGH and LOW, are also names given to the two states. The states are exclusive. That is:

If \( A \neq 0 \), then \( A = 1 \)
If \( A \neq 1 \), then \( A = 0 \)

II. Three Basic Boolean Operations

A. "OR"

Expression: \( F = A + B \)
Read: "\( F \) is equal to \( A \) or \( B \)"

Meaning: \( F \) is true (1) if either \( A \) or \( B \) is true.

Truth Table: | \( F \) | \( A \) | \( B \) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Logic Symbol: 

B. "AND"

Expression: \( F = A \cdot B = AB \)
Read: "\( F \) is equal to \( A \) and \( B \)"

Meaning: \( F \) is true (1) if \( A \) and \( B \) are true.

Truth Table: | \( F \) | \( A \) | \( B \) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Logic Symbol: 

C. "NOT"

Expression: \( F = \overline{A} \)
Read: "\( F \) is equal to not \( A \)"

Meaning: \( F \) is true (1) if \( A \) is not true.

Truth Table: | \( F \) | \( A \) |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Logic Symbol: 

III. Derived Logic Operations

A. "NOR"

Expression: \( F = \overline{A + B} \)
Read: "\( F \) is equal to \( A \) nor \( B \)"

Meaning: Combined OR and NOT operations. \( F \) is true (1) if the quantity \( A + B \) is not true.

Truth Table: | \( F \) | \( A \) | \( B \) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Logic Symbol: 

B. "NAND"

Expression: \( F = \overline{AB} \)
Meaning: Combined AND and NOT operations. \( F \) is true (1) is the quantity \( AB \) is not true.

Truth Table: | \( F \) | \( A \) | \( B \) |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Logic Symbol: 

IV. Basic Theorems

With the basic logic operations it is possible to deduce a set of basic theorems.

\[
\begin{align*}
1 + A &= 1 \\
0 + A &= A \\
A + A &= A \\
A + A &= 1 \\
A &= A \\
A + B &= B + A \\
A + (B + C) &= A + B + C \\
A(B + C) &= (AB)C \\
A(B + C) &= AB + AC \\
(A + B)(A + C) &= A + BC \\
\end{align*}
\]

V. DeMorgan's Theorem's

\[
\begin{align*}
\overline{A + B} &= \overline{A} \cdot \overline{B} \\
\overline{AB} &= \overline{A} + \overline{B} \\
\end{align*}
\]

Once expressions or logic symbol diagrams are written for a logic system, they can be manipulated (simplified) using the above rules.