Logic Functions

Logic Statement: H = 1 if (A and B and C are 1) or T is 1,
or H is true if all of A, B, and C are true, or T is true,
or the voltage at node H will be high if {the input voltage at node A is high} OR {the input voltage at S is high and the voltages at D and T are high}.

You wish to express under which conditions your burglar alarm goes off (B=1):
If the “Alarm Test” button is pressed (A=1),
OR if the Alarm is Set (S=1) AND { the door is opened (D=1) OR the trunk is opened (T=1)}

Boolean Expression: B = A + S(D + T)
This can be read B=1 if A=1 or S=1 AND (D OR T=1), i.e. B=1 if (A=1) or (S=1 AND (D OR T=1))
or B is true IF (A is true) OR (S is true AND D OR T is true)
or The voltage at node H will be high if (the input voltage at node A is high) OR (the input voltage at S is high) AND the voltages at D and T are high.

You are currently viewing page 1 of 6.
Evaluation of Logical Expressions with "Truth Tables"

The Truth Table completely describes a logic expression. In fact, we will use the Truth Table as the fundamental meaning of a logic expression. Two logic expressions are equal if their truth tables are the same.

Some Important Logical Functions

- **"AND"** $A \cdot B$ (or $A \cdot B \cdot C$)
- **"OR"** $A + B$ (or $A + B + C + \ldots$)
- **"INVERT" or "NOT"** $\overline{A}$ (or $\overline{A}$)
- **"not AND" = NAND** $\overline{A \cdot B}$ (only 0 when $A$ and $B=1$)
- **"not OR" = NOR** $\overline{A + B}$ (only 1 when $A + B=0$)
- **exclusive OR = XOR** $A \oplus B$ (only 1 when $A, B$ differ)
  i.e., $A + B$ except $A \cdot B$

The Important Logical Functions

The most frequent (i.e. important) logical functions are implemented as electronic "building blocks" or "gates". We already know about AND, OR and NOT. What are some others:

- Combination of above: inverted AND = NAND, inverted OR = NOR

And one other basic function is often used: the "EXCLUSIVE OR" ... which logically is "or except not and".

Logic Gates

These are circuits that accomplish a given logic function such as "OR". We will shortly see how such circuits are constructed. Each of the basic logic gates has a unique symbol, and there are several additional logic gates that are regarded as important enough to have their own symbol. The set is: AND, OR, NOT, NAND, NOR, and EXCLUSIVE OR.

Logic Circuits

With a combination of logic gates we can construct any logic function. In these two examples we will find the truth table for the circuit.

Now we complete the truth tables for $X$ and $Y$, and from that for $C$. (Note that $X = C = X + Y$ and finally $C = X + Y$)

### Truth Table for Logic Expression

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

### Some Important Logical Functions

- **"AND"** $A \cdot B$ (or $A \cdot B \cdot C$)
- **"OR"** $A + B$ (or $A + B + C + \ldots$)
- **"INVERT" or "NOT"** $\overline{A}$ (or $\overline{A}$)
- **"not AND" = NAND** $\overline{A \cdot B}$ (only 0 when $A$ and $B=1$)
- **"not OR" = NOR** $\overline{A + B}$ (only 1 when $A + B=0$)
- **exclusive OR = XOR** $A \oplus B$ (only 1 when $A, B$ differ)
  i.e., $A + B$ except $A \cdot B$