Implementation of Logic Functions using TG's

TG: \( m \quad in \quad output = in \quad \overline{out} \quad \overline{CLK} \)

1. Identify the control signals
2. Build a truth table
3. Convert the truth table to a multiplexer-style design by creating a path for each row of T.T. to the output.
4. The desired outputs are routed from the input to output.

Example 1

\[ F = AB + ABC + \overline{AC} \]

1. Let \( A \) & \( B \) be control signals.
2. Two-level MUX

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>( F )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>( \overline{C} )</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>( C )</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Note that one can combine TG's with static logic gates to achieve efficient implementations of logic functions.
Example 2 (Prob. 7.4c)

\[ F = \overline{(A + B + C)} + \overline{AB} = \overline{ABC} + \overline{AB} \]

1) Let \( A + B \) be control signals

2) \[
\begin{array}{ccc|c}
AB & C & F \\
\hline
00 & 0 & 1 \\
00 & 1 & 0 \\
01 & 0 & 1 \\
01 & 1 & 0 \\
10 & 0 & 0 \\
10 & 1 & 0 \\
11 & 0 & 0 \\
11 & 1 & 0 \\
\end{array}
\]

3) [Diagrams of CMOS transmission gates and delays]

CMOS Transmission Gate Delays

\[ \text{Delay} \propto RC \]
6) What is $R_{TG}(R)$?
   a) For the propagation of $V_{DD}$
      
      $$R_{TG} = \frac{R_{II}R_p}{2.4R_{pp}} \approx \frac{2R_{pp}}{4.8R_{pp}} \approx R_{pp}$$

      This approximates the NMOS shutting off.

   b) For propagation of a zero.
      
      $$R_{TG} = \frac{R_{II}R_p}{2R_{pp}} = \frac{2R_{pp}}{4.8R_{pp}} \approx \frac{R_{pp}}{4.8R_{pp}}$$

      Approximating the Kead for a PMOS shutting off.

2) What is $C_1$ and $C_2$ ($C_i=C_o$)?
   a) In off state: $C_1 = C_m = C_2 = C_{out} = C_{eff}(W_n + W_p) = 2W_{eff}$
   b) In on state: $C_1 = C_2 = C_{eff}(W_n + W_p) + \frac{1}{2} C_{eff}(W_n + W_p)$
For the above,

Elmore delay = \( r_{\text{inv}} (C_{\text{inv}} + C_{\text{res}}) + (r_{\text{inv}} + R_{\text{g}}) (C_{\text{g}} + C_{\text{out}}) \)

**Example 7.4**

Find the Elmore delay for the following.