Inverter (NOT gate) using switches

\[
\begin{align*}
5V & \quad X = 0 \text{ (switch open)} \quad 5V \rightarrow \text{HIGH} \rightarrow 1 \\
\frac{1}{2}R & \quad X = 1 \text{ (Switch closed)} \quad 0V \rightarrow \text{LOW} \rightarrow 0 \\
X & \quad \text{With switch open, there is no current} \\
\frac{1}{2} & \quad \text{through R; therefore, } Y = 5V \rightarrow 1 \\
\rightarrow \quad \text{With switch closed, } Y = 0V \rightarrow 0
\end{align*}
\]

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

NAND gate using switches

\[
\begin{align*}
5V & \quad A, B = 0 \text{ (switch open)} \\
\frac{1}{2}R & \quad A, B = 1 \text{ (switch closed)} \\
A & \quad \text{If either } A \text{ or } B \text{ is open, or} \\
\rightarrow & \quad \text{if they are both open, the current} \\
B & \quad \text{through } R \text{ is zero; therefore } Y = 5V = 1. \\
\rightarrow & \quad \text{If } A \text{ and } B \text{ are both closed, } Y = 0V = 0.
\end{align*}
\]

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

How do we make an AND gate?

Start with a NAND gate and follow it with a NOT gate. In other words, take the output of the NAND gate and make that signal control the ON/OFF switching of the switch in the NOT gate.
NOR gate using switches

If either A or B, or both A and B are closed, \( Y = 0 \) \( V = 0 \). If A & B are both open, there is no current through \( R \), therefore \( Y = 5V = 1 \).

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

How do we make an OR gate?
Start with a NOR gate followed by a NOT gate.

We can also implement the logic circuits without using resistors. These circuits use complementary switches.

\[
\begin{array}{c|c|c}
X & \text{switch state} & \overline{X} \\
\hline
0 & \text{open} & 1 \\
1 & \text{closed} & 0 \\
\end{array}
\]

this is a normally open (NO) switch

\[
\begin{array}{c|c|c}
X & \overline{X} & \text{switch state} \\
\hline
0 & 1 & \text{closed} \\
1 & 0 & \text{open} \\
\end{array}
\]

this is a normally closed (NC) switch

With \( X = 0 \), the switch on the left is open while the one on the right is closed (since its controlling variable \( \overline{X} = 1 \)). The situation is reversed when \( X = 1 \).
Inverter using complementary switches

When \( X=0 \), the upper switch is closed and the lower switch is open; therefore, \( Y=5V=1 \). When \( X=1 \), the upper switch is open and the lower switch is closed; therefore, \( Y=0V=0 \). Note that the switches cannot be both closed.

NAND gate using complementary switches

Having either \( A \) or \( B \) or both equal 0 ensures that \( Y=5V=1 \). The only way to have \( Y=0 \) is to have \( A=B=1 \). Convince yourself that this is true

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

AND gate using complementary switches

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
NOR gate using complementary switches

If \( A \) or \( B \) or both are 1, then \( Y=0 \). The only way to have \( Y=1 \) is to have \( A=B=0 \). Convince yourself that this is true.

<table>
<thead>
<tr>
<th>( A )</th>
<th>( B )</th>
<th>( Y )</th>
</tr>
</thead>
<tbody>
<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>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

We can implement the switches using transistors. For a normally open switch, we can use a npn bipolar junction transistor (npn BJT) or an n-channel metal-oxide semiconductor field-effect transistor (n-channel MOSFET).

When a high voltage is applied as \( X \) in the transistors, the terminals \( a \& b \) are connected together. If \( X=0 \), \( a \) and \( b \) are kept open. The terminals of the transistors are identified as follows:

- **npn BJT**: \( a \) (Collector), \( b \) (Emitter)
- **n-channel MOSFET**: \( a \) (Gate), \( b \) (Source)
We can also implement the normally closed switches using transistors. For this purpose, we use the npn BJT and the p-channel MOSFET.

\[
\begin{array}{c|cc}
0 & \bar{X} & X \\
1 & 0 & 1 \\
\end{array}
\]

switch state

\[
\begin{array}{c}
\bar{X} \\
0 \\
1 \\
\end{array}
\]

on (closed)

off (open)

Note that the signal applied as the input to the transistor is simply \( X \). Unlike the p-type devices, \( X = 0 \) results in a closed switch while \( X = 1 \) in an open switch.

We can now go back and construct transistor-based logic gates, instead of the ones with switches.

**NOT gate**

\[
\begin{array}{c}
5V \\
\downarrow \\
\bar{X} \\
\downarrow \\
Y \\
\downarrow \\
\end{array}
\]

using n-type transistors

\[
\begin{array}{c}
5V \\
\downarrow \\
\bar{X} \\
\downarrow \\
Y \\
\downarrow \\
\end{array}
\]

\[
\begin{array}{c|c}
X & Y \\
0 & 1 \\
1 & 0 \\
\end{array}
\]

\[
\begin{array}{c}
5V \\
\downarrow \\
\bar{X} \\
\downarrow \\
Y \\
\downarrow \\
\end{array}
\]

using p-type transistors

\[
\begin{array}{c}
5V \\
\downarrow \\
\bar{X} \\
\downarrow \\
Y \\
\downarrow \\
\end{array}
\]
**NOT gate using complementary switches**

\[
\begin{array}{c|c|c}
X & Y & \text{Truth Table} \\
0 & 1 & 0 \\
1 & 0 & 1 \\
\end{array}
\]

**NAND gate**

\[
\begin{array}{c|c|c|c|c}
A & B & Y & \text{Truth Table} \\
0 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 \\
1 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 & 0 \\
\end{array}
\]

*using n-type transistors*

*using p-type transistors*
NAND gate using complementary switches

\[ \begin{array}{c}
\text{NOR gate} \\
\text{using } n\text{-type transistors} \\
\text{using } p\text{-type transistors}
\end{array} \]
NOR gate with complementary switches

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