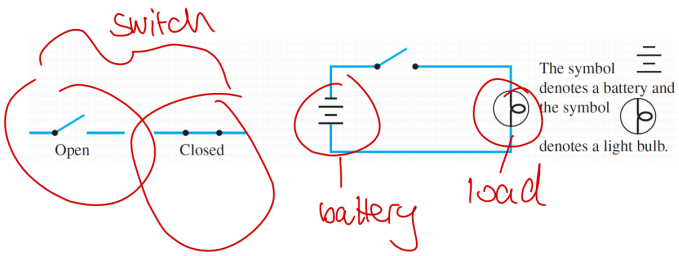
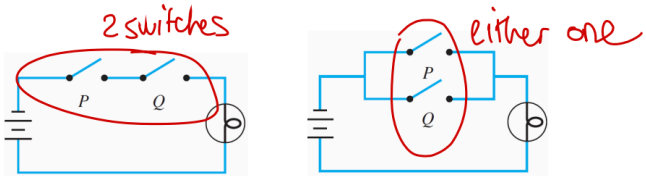
Digital logic circuits

Simple circuit



A simple circuit has three elements: switch, battery and load



(a) Switches "in series" (b) Switches "in parallel"

Switches can either be used "in series" (the circuit will only work when both switches are closed) or "in parallel" (the circuit will work when one of the both switches are closed).

The in series circuit is equivalent to the AND truth table, where closed is equivalent to true and the truth table is only true when both values are true (closed).

The in parallel circuit is equivalent to the OR truth table, where closed is equivalent to true and the truth table is only false when both values are false (open).

Combinatorial circuit

- cominatorial: type of digital circuit whose output only depends on the current input.
- sequential: output depends on both the current input and previous outputs. This means a sequential circuit preserves a memory of the input while a combinational circuit does not (out of scope for this course)

Input			Output
P	*Q*	*R*	*S*
1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	1
0	0	0	0

3 possible inputs

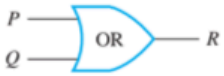
1 possible output

Rules for cominatorial circuits

1. Never combine the input wires.
2. Never feed back the output of a gate into the same gate

- combinatorial equivalence: two circuits can have the same output but built completely differently
- circuit minimization: for production of electrical circuit it makes sense minimizing more complicated circuits to simplified ones so that they cost lesser in production.

Logic gates

<table>
<tr><th>Type of Gate</th><th>Symbolic Representation</th><th>Action</th></tr>
<tr><td>NOT</td><td> 
</td><td>
<table>
<tr><th>Input</th><th>Output</th></tr>
<tr><th>P</th><th>R</th></tr>
<tr><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td></tr>
</table>
</td></tr>
<tr><td>AND</td><td> 
</td><td>
<table>
<tr><th colspan="2">Input</th><th>Output</th></tr>
<tr><th>P</th><th>Q</th><th>R</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table>
</td></tr>
<tr><td>OR</td><td> 
</td><td>
<table>
<tr><th colspan="2">Input</th><th>Output</th></tr>
<tr><th>P</th><th>Q</th><th>R</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table>
</td></tr>
</table>

Disjunctive Normal Form (DNF)

Also called "Or of And's"

$$G(p, q) \equiv (p \wedge q) \vee (\neg p \wedge q)$$

Conjunctive Normal Form (CNF)

Also called "And of Or's"

$$G(p, q) \equiv (p \vee q) \wedge (\neg p \vee q)$$

Multiplexer

S	Function
0	AND
1	OR