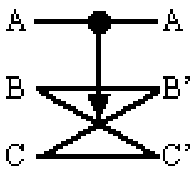
Fredkin Gates

The **Fredkin gate** (also **CSWAP gate**) is a computational circuit suitable for reversible computing, invented by Ed Fredkin. It is *universal*, which means that any logical or arithmetic operation can be constructed entirely of Fredkin gates. The Fredkin gate is the three-bit gate that swaps the last two bits if the first bit is 1.



Definition:

The basic Fredkin gate is a controlled swap gate that maps three inputs (C, I_1, I_2) onto three outputs (C, O_1, O_2). The C input is mapped directly to the C output. If C = 0, no swap is performed; I_1 maps to O_1, and I_2 maps to O_2. Otherwise, the two outputs are swapped so that I_1 maps to O_2, and I_2 maps to O_1. It is easy to see that this circuit is reversible, i.e., "undoes itself" when run backwards. A generalized $n \times n$ Fredkin gate passes its first n-2 inputs unchanged to the corresponding outputs, and swaps its last two outputs if and only if the first n-2 inputs are all 1.

The Fredkin gate is the reversible three-bit gate that swaps the last two bits if the first bit is 1.

Truth table

INPUT			**OUTPUT**		
C	I_1	I_2	C	O_1	O_2
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	1

Matrix form

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

It has the useful property that the numbers of 0s and 1s are conserved throughout, which in the billiard ball model means the same number of balls are output as input. This corresponds nicely to the conservation of mass in physics, and helps to show that the model is not wasteful.

Logic function with XOR and AND gates:

$O_1 = I_1$ XOR S
$O_2 = I_2$ XOR S

with $S = (I_1$ XOR $I_2)$ AND C

It can also be implemented by the following logic:

O_1 = (NOT C AND I_1) OR (C AND I_2) = CI_1+CI_2
O_2 = (C AND I_1) OR (NOT C AND I_2) = CI_1+CI_2
C_{out}= C_{in}

Completeness:

It is easy to see that the Fredkin gate is universal, since it can be used to implement AND and NOT:

if I2 = 0, O2 = C and I1
if I1 = 0 and I2 = 1, O2 = not C