- The implementation of half adder using ***exclusive–OR*** and an ***AND*** gates is used to show that two half adders can be used to construct a full adder.
- The inputs to the ***XOR*** gate are also the inputs to the ***AND*** gate.

2. Full Adder

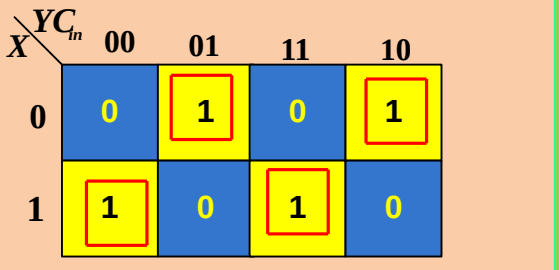
Full Adder is a combinational circuit that performs the addition of three bits (two significant bits and previous carry).

- It consists of ***three inputs and two outputs***, two inputs are the bits to be added, the third input represents the carry form the previous position.
- The full adder is usually a component in a cascade of adders, which add 8, 16, etc, binary numbers.

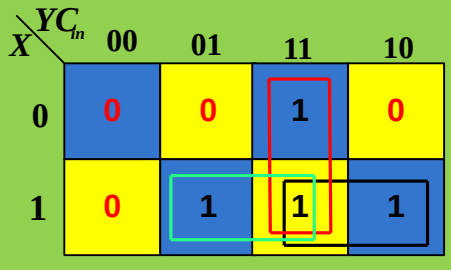
Inputs			Outputs	
X	Y	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Truth table for the full adder

- The S output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1.
- The C_{out} output has a carry 1 if two or three inputs are equal to 1.
- The Karnaugh maps and the simplified expression are shown in the following figures:



$$S = \overline{X}\overline{Y}C_{in} + \overline{X}Y\overline{C_{in}} + X\overline{Y}\overline{C_{in}} + XYC_{in}$$



$$C_{out} = XY + XC_{in} + YC_{in}$$

$$\left.\begin{cases} S = \overline{X}\,\overline{Y}C_{in} + \overline{X}Y\overline{C_{in}} + X\overline{Y}\,\overline{C_{in}} + XYC_{in} \\ C_{out} = XY + XC_{in} + YC_{in} \end{cases}\right\} 1 \quad \textit{(Sum of products)}$$