1. Define half adder.

Answer:

The A half adder is a type of adder, an electronic circuit that performs the addition of numbers. The half adder is able to add two single binary digits and provide the output plus a carry value. It has two inputs, called A and B, and two outputs S (sum) and C (carry). The common representation uses a XOR logic gate and an AND logic gate.

1. Draw a truth table for the sum and carry of half adder.

Answer:

| A | B | Sum | Carry |
| --- | --- | --- | --- |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |

1. Write the sop expression from the truth table.

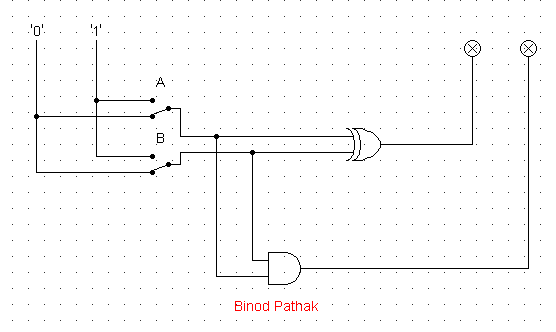
Answer: Sum=A’B+AB’

Carry=A.B

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | Carry in | Sum | Carry |
| 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 |

1. Draw the circuit using logsim.

Answer:



2

* 1. Draw the truth table for the outputs of the full adder.

Answer:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | Carry in | Sum | Carry 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 |

* 1. Write the corresponding sop expression for sum and carry of full adder and simplify the expression

Answer:

Sum=AB’C’+A’BC’+A’B’C+ABC

=A(B’C’-BC)+A’(BC’+B’C)

=A(B⊕C)’+A’(B⊕C)

=A (B⊕ C)

=A ⊕B ⊕C

Carry out=ABC’+AB’C+A’BC+ABC

=ABC’+ABC+AB’C+A’BC

=AB(C’+C)+C(AB’+A’B)

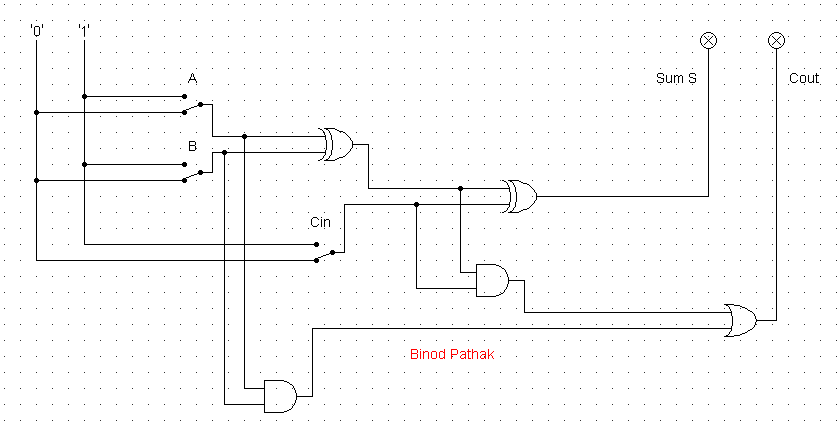
=AB+C(AB’+A’B)

=AB+C(A+B)

=AB+AC+BC

* 1. Draw full adder using two half adder and an OR gate.

Answer:

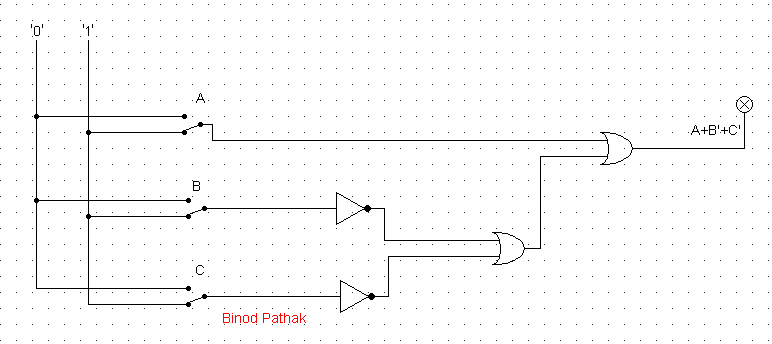


3.Using the three stages of design, construct the circuits for the following input /output values. Here A, B and C are the inputs whereas D, E, F, G, H and I are outputs.

*Note: Draw circuit diagram using logsim corresponding to the simplified expression of outputs D, E, F, G, H and I.*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F | G | H | I |
| 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 |

Answer: **D=A+B’+C’**

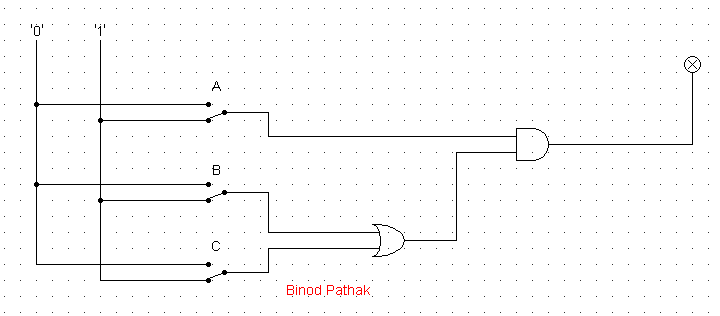


**E = ABC+ABC’+AB’C**

**= AB(C+C’)+AC(B+B’)**

**=AB+AC**

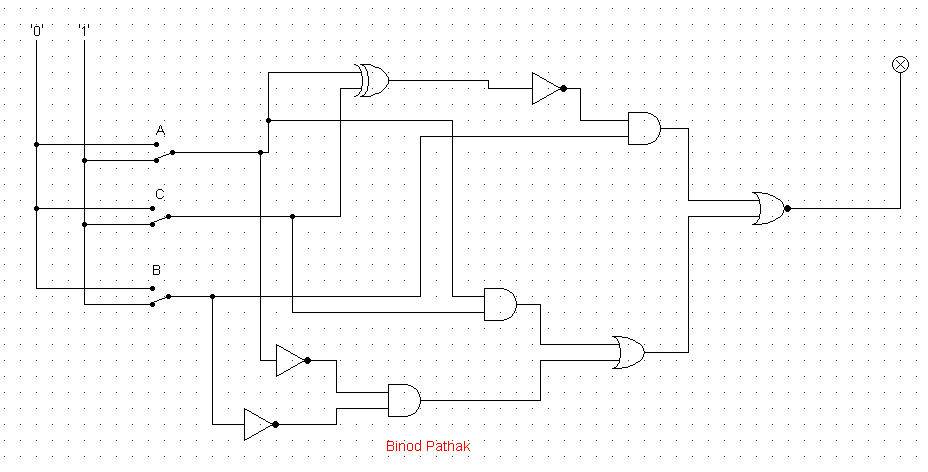
**= A (B +C)**

****

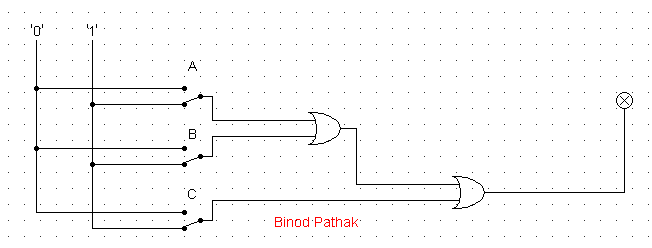
**F=A’B’C’+A’B’C+A’BC’+A’BC’+ABC+AB’C**

**=A’B’(C’+C)+(A’C’+AC)B+AC(B’+B)**

**= A’B’+[(A⊕C)’]B+AC**

****

**G=A+B+C**

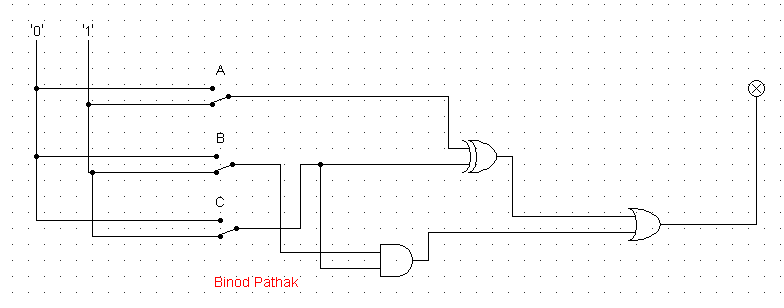
****

**H=A’B’C’+A’BC’+A’BC’+ABC+AB’C+A’BC**

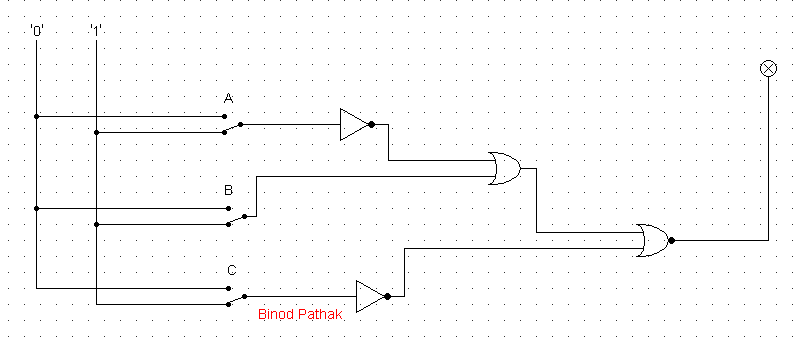
**=A’C’(B’+B)+AC(B+B’)+BC(A+A’)**

**=A’C’+AC+BC**

**= (A ⊕C)’+BC**

****

**I=A’+B+C’**

****