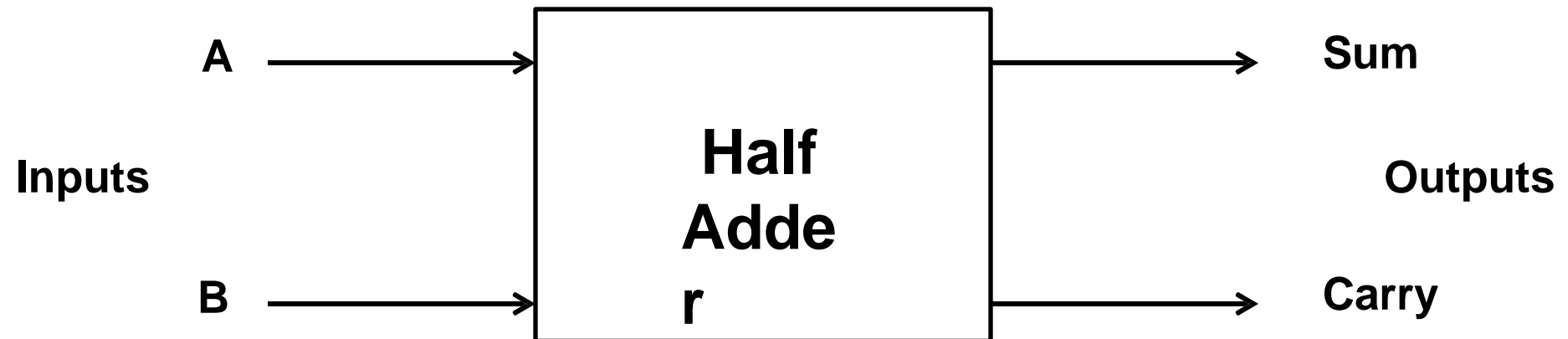


Half Adder

- ✓ Half adder is a combinational logic circuit with two inputs and two outputs.
- ✓ It is a basic building block for addition of two single bit numbers.



Half Adder

Truth Table for Half Adder

Input		Output	
A	B	Sum (S)	Carry (C)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Half Adder

K-map for Sum Output:

		A	
		\bar{A}	A
B	\bar{B} 0	0	1
	B 1	1	0

$$S = \bar{A}B + A\bar{B}$$

$$S = A \oplus B$$

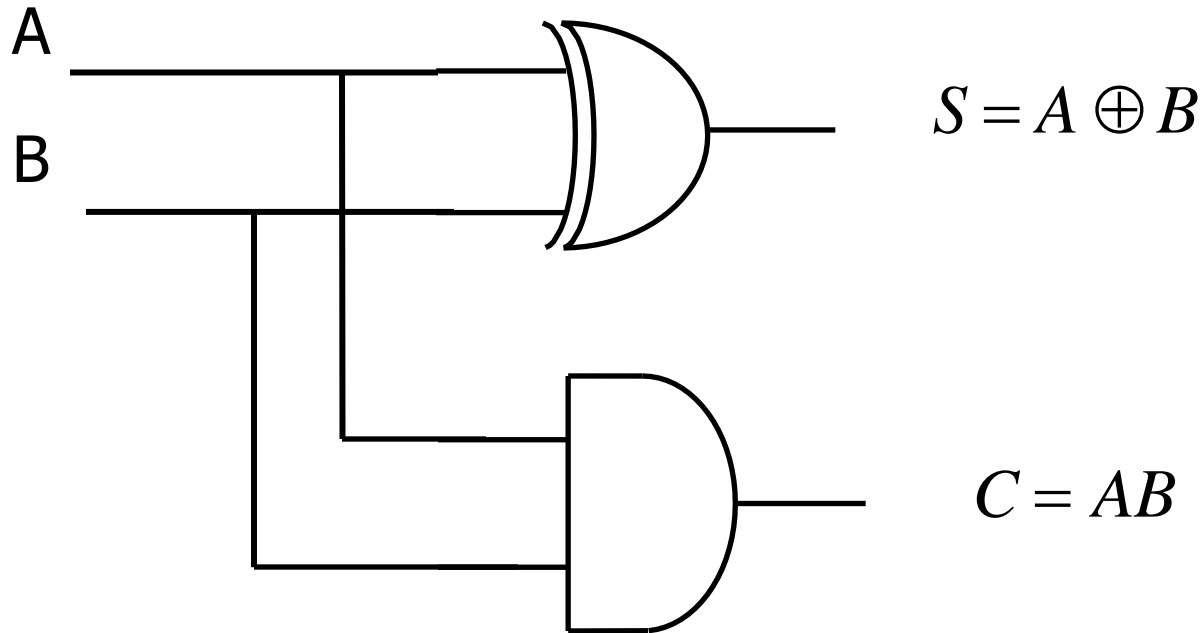
K-map for Carry Output:

		A	
		\bar{A}	A
B	\bar{B} 0	0	0
	B 1	0	1

$$C = AB$$

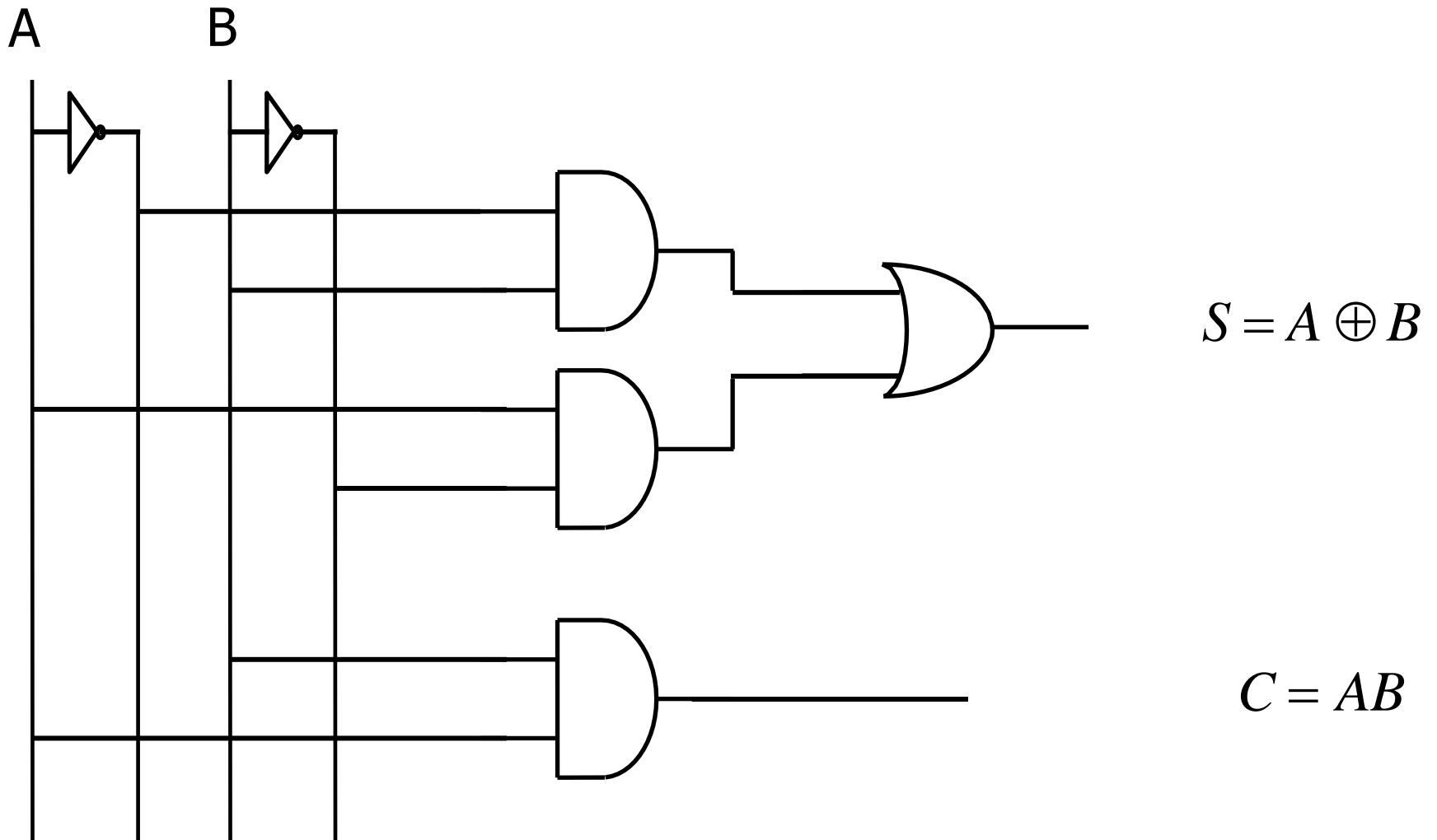
Half Adder

Logic
Diagram:



Half Adder

Logic Diagram using Basic Gates:

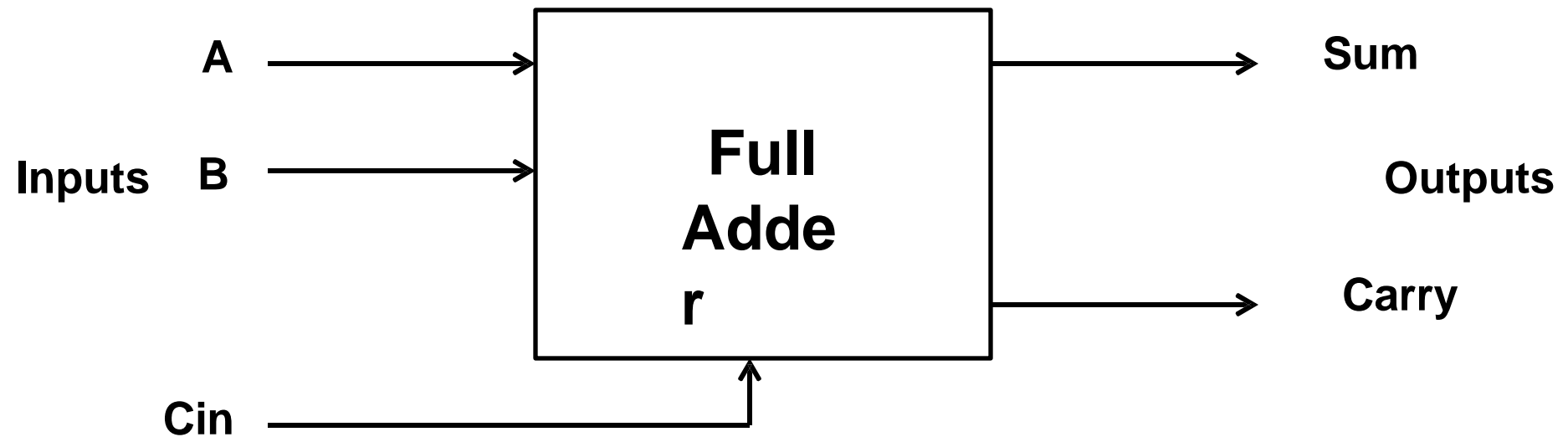


Combinational Logic Circuits

- ✓ **Standard Boolean representation:** Sum of Product (SOP) & Product of Sum (POS), Maxterm and Minterm , Conversion between SOP and POS forms, realization using NAND/NOR gates.
- ✓ **K-map reduction technique for the Boolean expression:** Minimization of Boolean functions up to 4 variables (SOP & POS form)
- ✓ **Design of Airthmetic circuits and code converter using K-map:** Half and Full Adder, Half and Full Subtractor, Gray to Binary and Binary to Gray Code Converter (up to 4 bit).

Full Adder

- ✓ Full adder is a combinational logic circuit with three inputs and two outputs.



Full Adder

Truth Table

Inputs			Outputs	
A	B	Cin	Sum (S)	Carry (C)
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

Full Adder

K-map for Sum Output:

BC A		$\overline{B}\overline{C}$ 00	$\overline{B}C$ 01	BC 11	$B\overline{C}$ 10
\overline{A}	0	0	1	0	1
A	1	1	0	1	0

Diagram illustrating the K-map for the Sum Output (S) of a Full Adder. The K-map is a 2x4 grid with inputs A and BC. The output S is 1 for the following combinations of A and BC: (0, 01), (0, 10), (1, 00), and (1, 11). These combinations are circled in blue. The output S is 0 for the remaining combinations: (0, 00), (0, 11), (1, 01), and (1, 10). Blue arrows point from the circled 1s to the corresponding minterms: $\overline{A}\overline{B}C$, $\overline{A}B\overline{C}$, $A\overline{B}\overline{C}$, and $AB\overline{C}$.

$$S = \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}C + AB\overline{C}$$

$$S = \overline{A}\overline{B}C + A\overline{B}C + \overline{A}B\overline{C} + AB\overline{C}$$

$$S = C(\overline{A}\overline{B} + AB) + \overline{C}(\overline{A}B + A\overline{B})$$

$$\text{Let } \overline{A}B + A\overline{B} = X$$

$$\therefore S = C(\overline{X}) + \overline{C}(X)$$

$$S = C \oplus X$$

$$\text{Let } X = A \oplus B$$

$$\therefore S = C \oplus A \oplus B$$

Full Adder

K-map for Carry Output:

$\begin{array}{c} \text{BC} \\ \text{A} \end{array}$		$\overline{B}\overline{C}$	$\overline{B}C$	BC	$B\overline{C}$
		00	01	11	10
\overline{A}	0	0	0	1	0
A	1	0	1	1	1

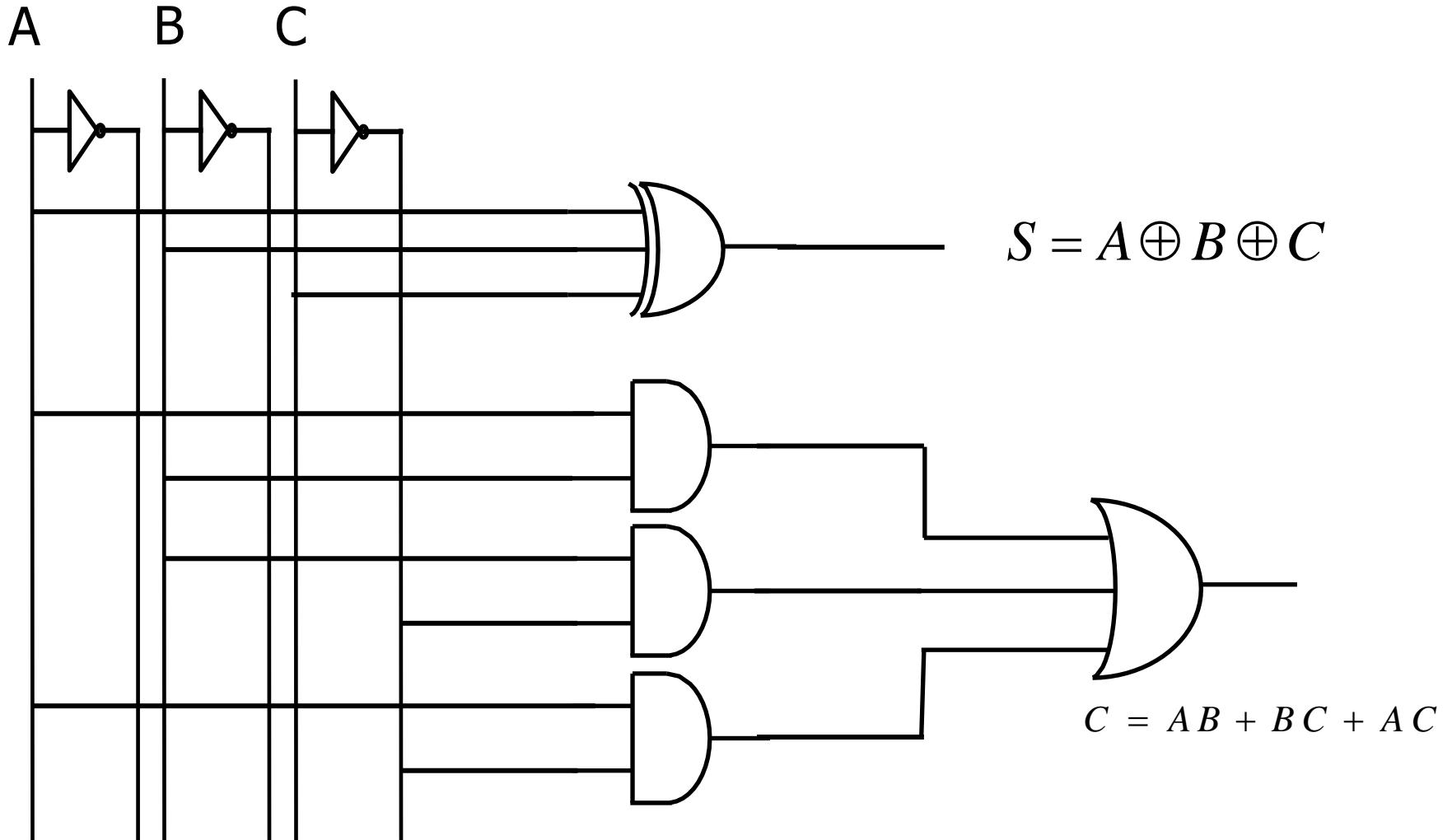
Diagram illustrating the K-map for Carry Output (C) of a Full Adder. The K-map is a 2x4 grid with inputs A and BC. The output C is 1 for the following combinations: (A=0, BC=11), (A=1, BC=01), (A=1, BC=11), and (A=1, BC=10). The K-map shows three groups of 1s circled in blue, corresponding to the terms in the Boolean expression:

- Group 1: (A=0, BC=11) and (A=1, BC=11) → BC
- Group 2: (A=1, BC=01) and (A=1, BC=11) → AB
- Group 3: (A=1, BC=01) and (A=1, BC=10) → AC

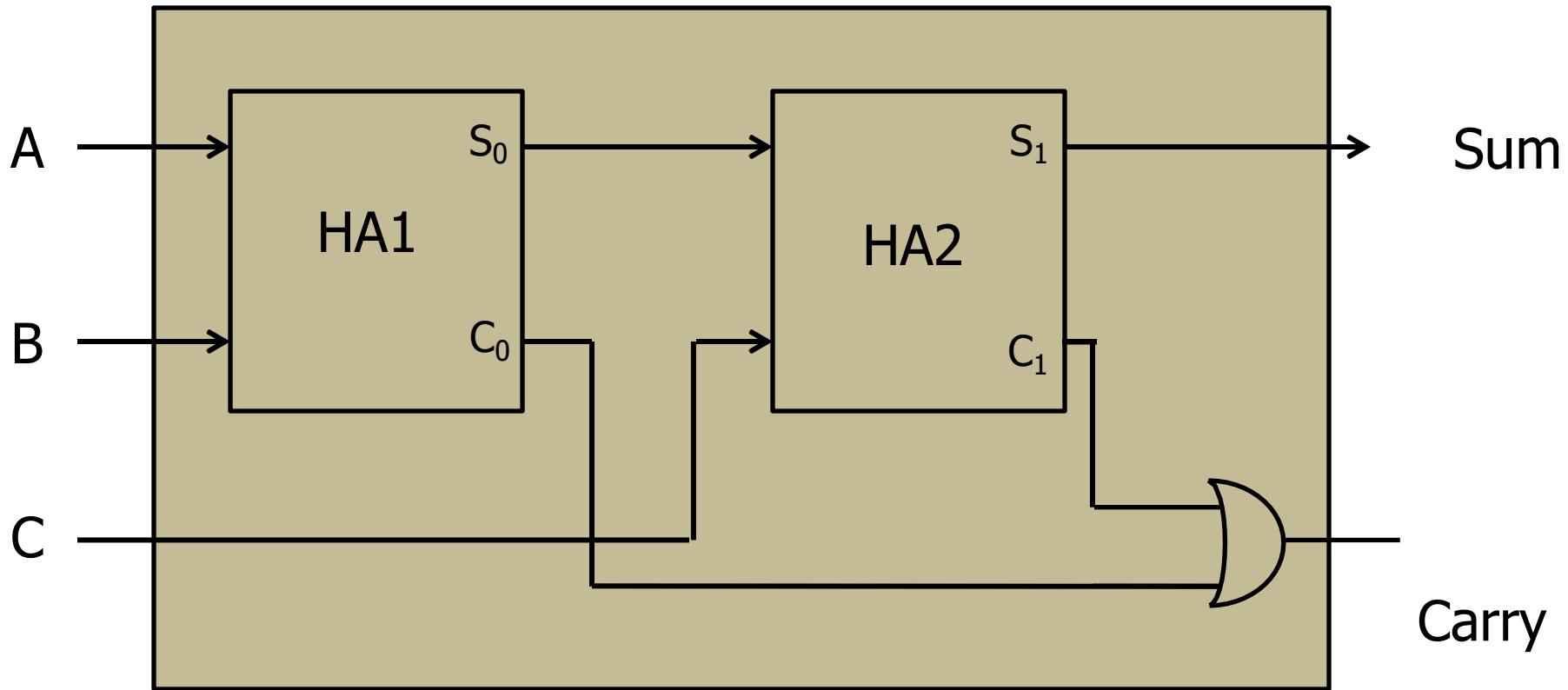
$$C = AB + BC + AC$$

Full Adder

Logic Diagram:



Full Adder using Half Adders

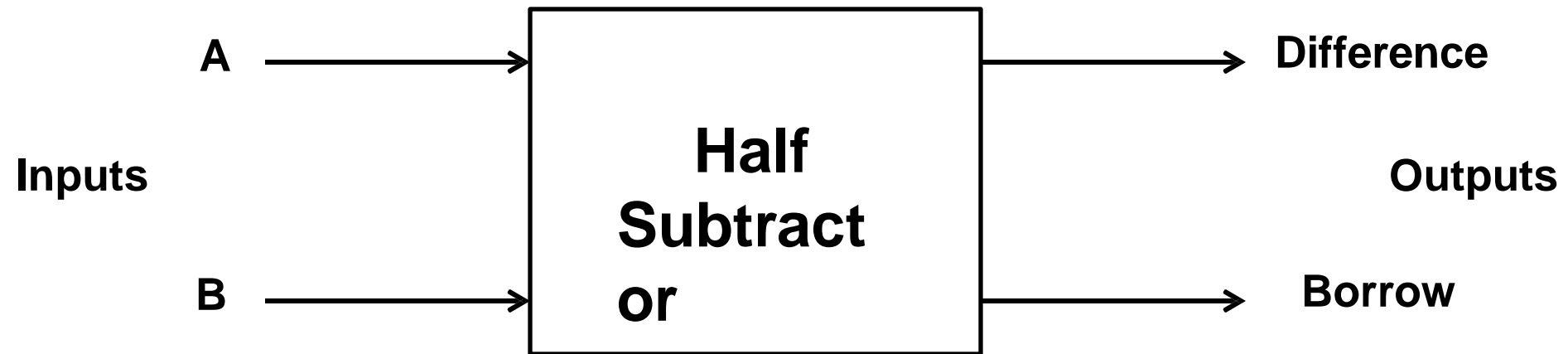


Combinational Logic Circuits

- ✓ **Standard Boolean representation:** Sum of Product (SOP) & Product of Sum (POS), Maxterm and Minterm , Conversion between SOP and POS forms, realization using NAND/NOR gates.
- ✓ **K-map reduction technique for the Boolean expression:** Minimization of Boolean functions up to 4 variables (SOP & POS form)
- ✓ **Design of Airthmetic circuits and code converter using K-map:** Half and Full Adder, **Half Subtractor** and Full Subtractor, Gray to Binary and Binary to Gray Code Converter (up to 4 bit).

Half Subtractor

- ✓ Half subtractor is a combinational logic circuit with two inputs and two outputs.
- ✓ It is a basic building block for subtraction of two single bit numbers.



Half Subtractor

Truth Table

Input		Output	
A	B	Difference (D)	Borrow (B)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Half Subtractor

K-map for Difference Output:

		A	
		\bar{A}	A
B	\bar{B} 0	0	1
	B 1	1	0

$$D = \bar{A}B + A\bar{B}$$

$$D = A \oplus B$$

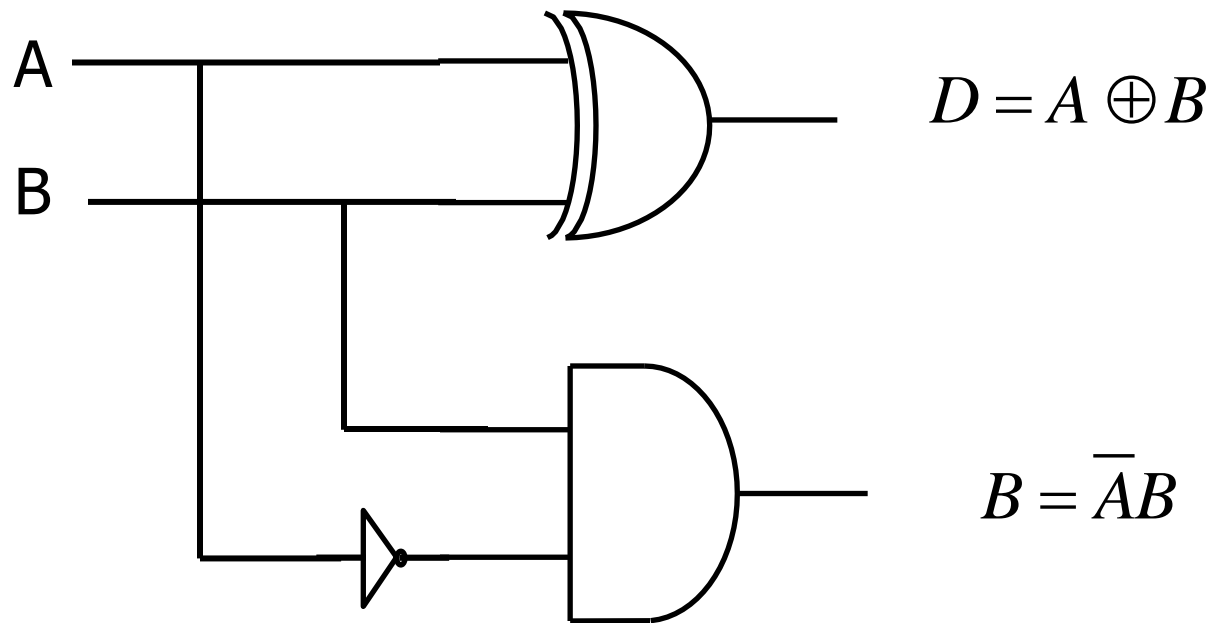
K-map for Borrow Output:

		A	
		\bar{A}	A
B	\bar{B} 0	0	1
	B 1	0	0

$$B = \bar{A}B$$

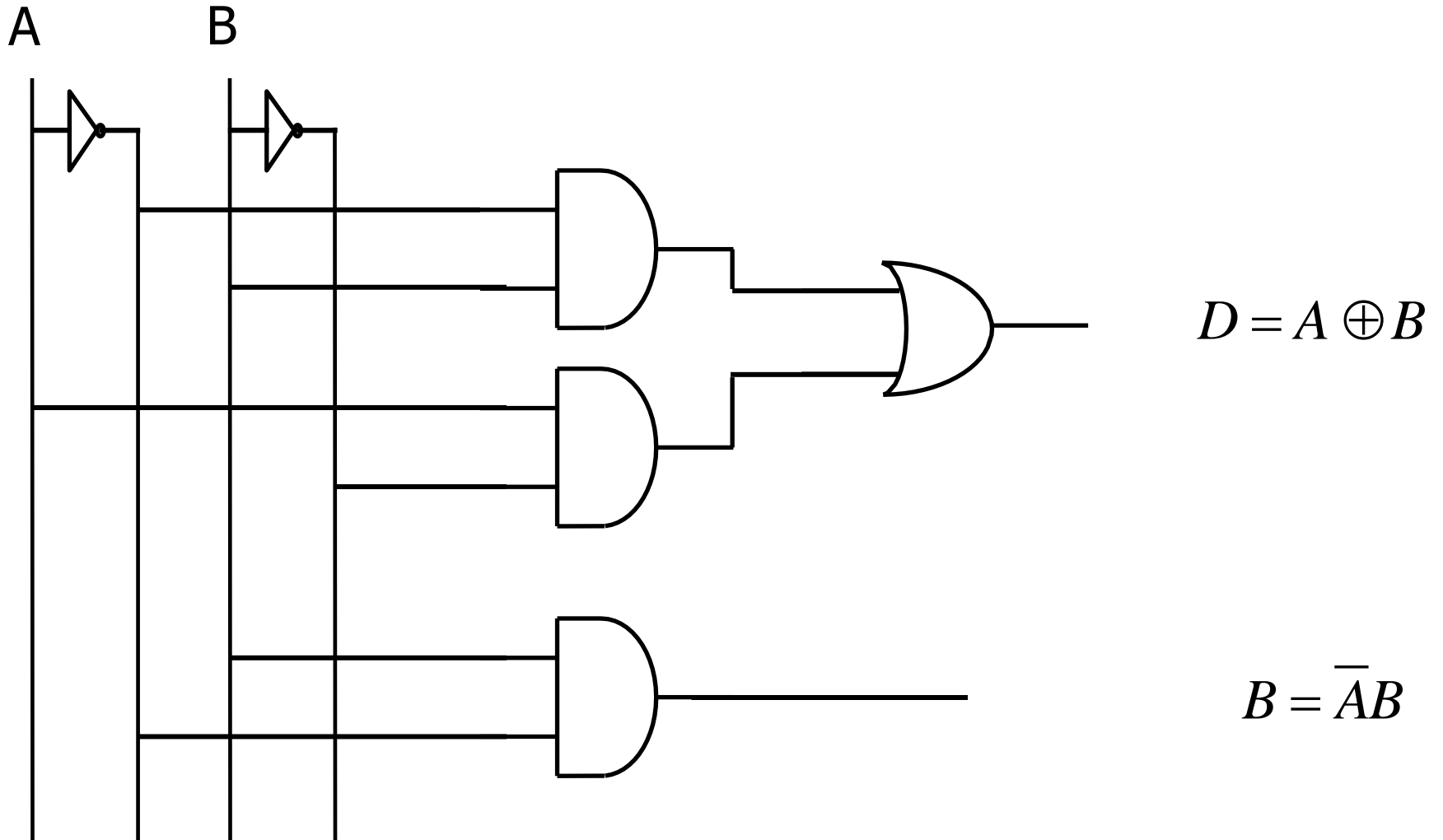
Half Subtractor

Logic Diagram:



Half Subtractor

Logic Diagram using Basic Gates:

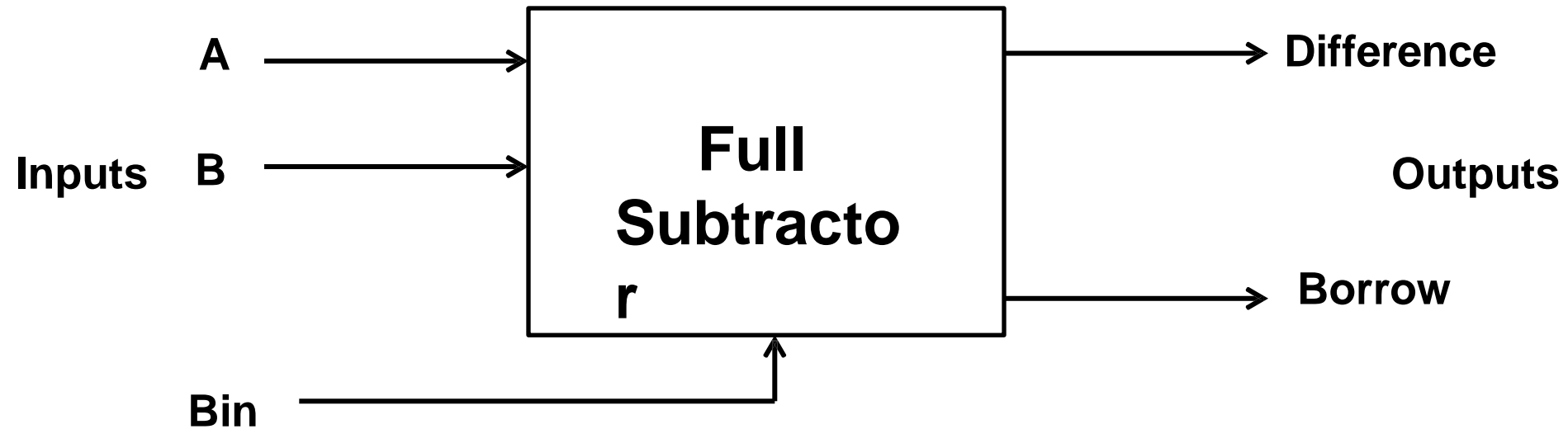


Combinational Logic Circuits

- ✓ **Standard Boolean representation:** Sum of Product (SOP) & Product of Sum (POS), Maxterm and Minterm , Conversion between SOP and POS forms, realization using NAND/NOR gates.
- ✓ **K-map reduction technique for the Boolean expression:** Minimization of Boolean functions up to 4 variables (SOP & POS form)
- ✓ **Design of Airthmetic circuits and code converter using K-map:** Half and Full Adder, Half and **Full Subtractor**, Gray to Binary and Binary to Gray Code Converter (up to 4 bit).

Full Subtractor

- ✓ Full subtractor is a combinational logic circuit with three inputs and two outputs.



Full Subtractor

Truth Table

Inputs			Outputs	
A	B	Bin (C)	Difference (D)	Borrow (B0)
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Full Subtractor

K-map for Difference Output:

<div>B AC</div>		$\overline{B}\overline{C}$	$\overline{B}C$	BC	$B\overline{C}$
		00	01	11	10
\overline{A}	0	0	1	0	1
A	1	1	0	1	0

Diagram illustrating the K-map for the Difference Output (D) of a Full Subtractor. The K-map is a 2x4 grid with inputs A and B, and carry-in C. The output D is 1 for the following combinations: $\overline{A}\overline{B}C$, $\overline{A}B\overline{C}$, $A\overline{B}\overline{C}$, and ABC . These combinations are circled in the K-map, and arrows point to their corresponding minterms: $\overline{A}\overline{B}C$, $\overline{A}B\overline{C}$, ABC , and $A\overline{B}\overline{C}$.

$$D = \overline{A}\overline{B}C + \overline{A}B\overline{C} + ABC + A\overline{B}\overline{C}$$

$$D = \overline{A}\overline{B}C + ABC + \overline{A}B\overline{C} + A\overline{B}\overline{C}$$

$$D = C(\overline{A}\overline{B} + AB) + \overline{C}(\overline{A}B + A\overline{B})$$

$$\text{Let } \overline{A}B + A\overline{B} = X$$

$$\therefore D = C(\overline{X}) + \overline{C}(X)$$

$$D = C \oplus X$$

$$\text{Let } X = A \oplus B$$

$$\therefore D = C \oplus A \oplus B$$

Full Subtractor

K-map for Borrow Output:

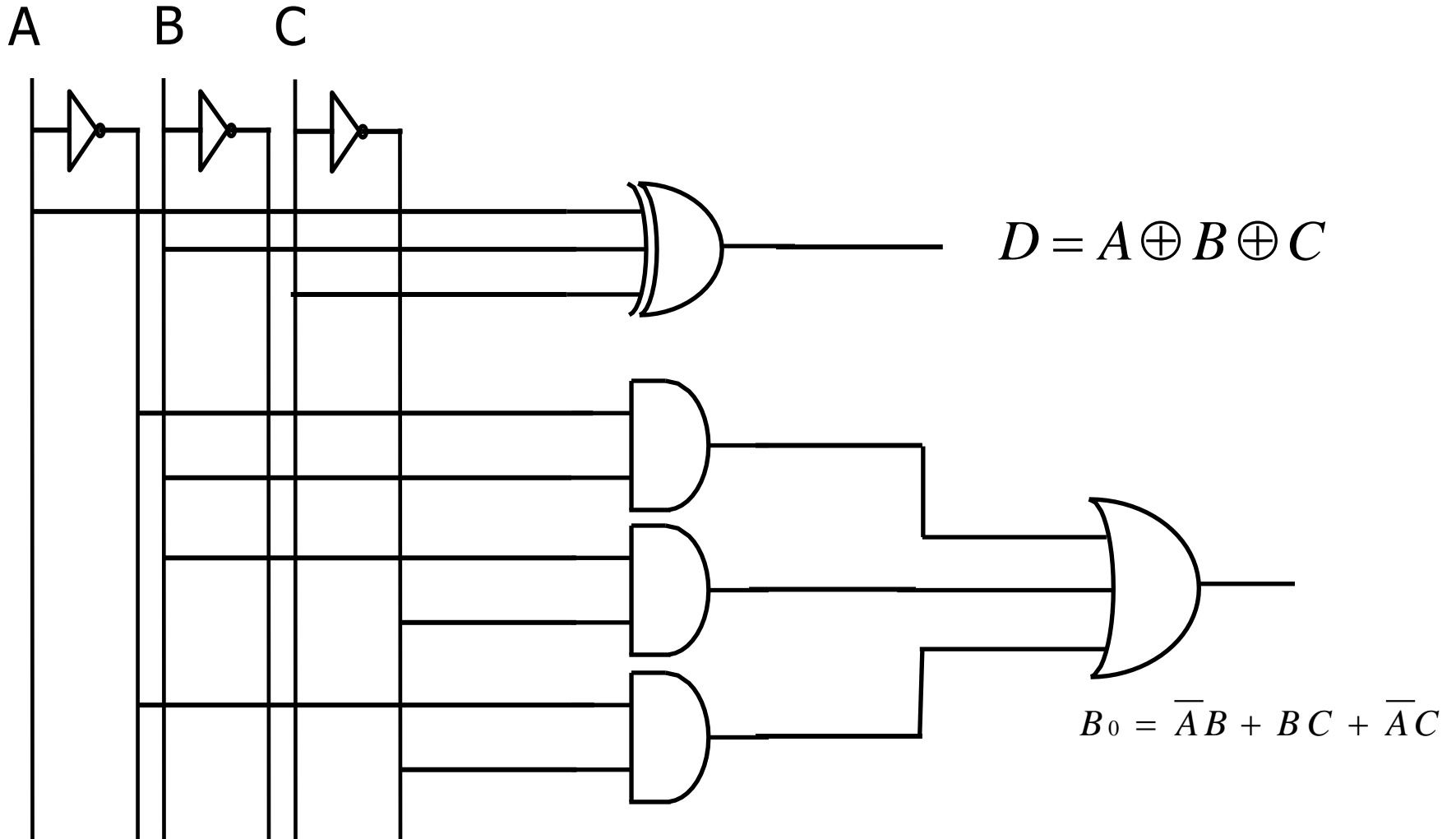
<div>B AC</div>		$\overline{B}\overline{C}$	$\overline{B}C$	BC	$B\overline{C}$
		00	0	11	10
\overline{A}	0	0	1	1	1
A	1	0	0	1	0

Diagram illustrating the K-map for Borrow Output (B_0). The K-map is a 2x4 grid with columns labeled $\overline{B}\overline{C}$, $\overline{B}C$, BC , and $B\overline{C}$, and rows labeled \overline{A} and A . The values in the cells are 0, 1, 1, 1 for $\overline{A}=0$ and 0, 0, 1, 0 for $A=1$. Three groups of 1s are circled: a horizontal group of three 1s in the $\overline{A}=0$ row, a vertical group of two 1s in the BC column, and a diagonal group of two 1s (top-right to bottom-left). Arrows point from these groups to the terms $\overline{A}C$, BC , and $\overline{A}B$ respectively.

$$B_0 = \overline{A}B + BC + \overline{A}C$$

Full Subtractor

Logic Diagram:



Full Subtractor using Half Subtractor

