Subtractor

(i) Half Subtractor (ii) Full subtractor.

Laws of subtraction.

0 − 0 = 0
1 − 0 = 1
1 − 1 = 1
0 − 1 = 1.

The Half Subtractor:

A logic circuit for the subtraction of B (subtrahend) from A (minuend) where A and B are 1-bit numbers is referred as a half-subtractor. The subtraction is per the laws of subtraction mentioned in the above table, The ~~truble~~ truth table for half subtractor is shown below.

Inputs A	Inputs B	outputs D	outputs B	
0	0	0	0	m0
0	1	1	1	m1
1	0	1	0	m2
1	1	0	0	m3.

(A − B)

~~borr~~

First three column of this truth table represents the truth table of EX-OR gate $D = A\bar{B} + \bar{A}B$.

Difference $D = A \oplus B$.

and Borrow $B = \bar{A}B$, is obtained from I, II and IV column. This is of an AND gate with inputs A complemented.

Obtain the logic circuit using.

① K- Map : for Difference.

	$\bar{A}$	A
$\bar{B}$	m0	m2 1
B	m1 1	m3

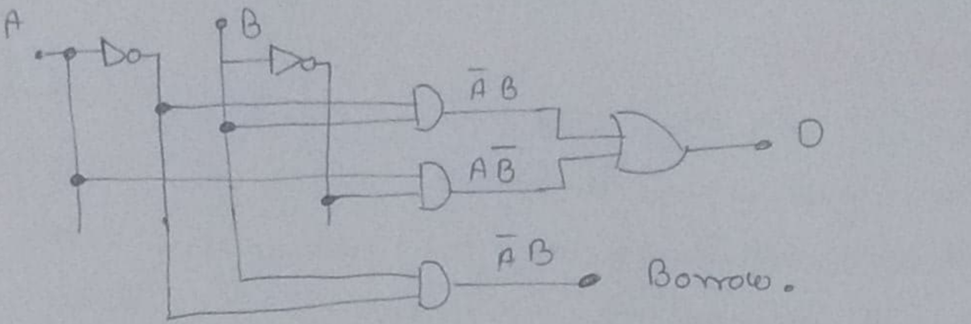
$Y = A\bar{B} + \bar{A}B$

For Borrow:

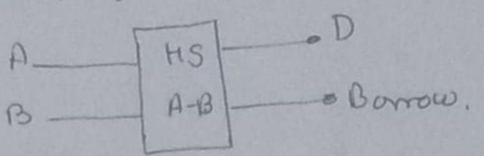
	$\bar{A}$	A
$\bar{B}$	m0	m2
B	m1 1	m3

$Y = \bar{A}B.$

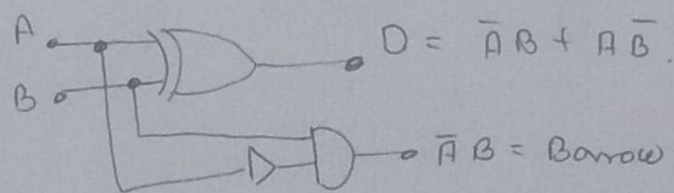
circuit using AND, OR and Inverter gates.



Logic symbol for H.A.



Logic circuit using EX-OR and inverted gate.



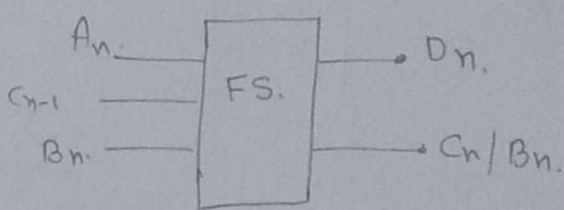
comparing a half-subtractor with half-adder, we find that the expressions for the SUM and Difference outputs are just the same. The expression for the Borrow in the case of half subtractor is similar to what we have for carry in case of the half-adder, the only difference is that in case of Half subtraction for Borrow one of the input is inverted.

Full-subtractor:

Full subtractor is required for performing multiple subtraction, wherein a borrow from the previous bit position may also be there. So a full-subtractor will have three inputs A_n (minuend), B_n (Subtrahend) and C_{n-1} / B_{n-1} (borrow from the previous stage/bit) and two outputs D_n (difference) and B_n (borrow). The truth table is shown below.

Inputs.			outputs.		
A_n	B_n	C_{n-1}	D_n	C_n/B_n	
0	0	0	0	0	m0
0	0	1	1	1	mϕ
0	1	0	1	1	m2
0	1	1	0	1	m3
1	0	0	1	0	m4
1	0	1	0	0	m5
1	1	0	0	0	m6
1	1	1	1	1	m7.

Logical Symbol



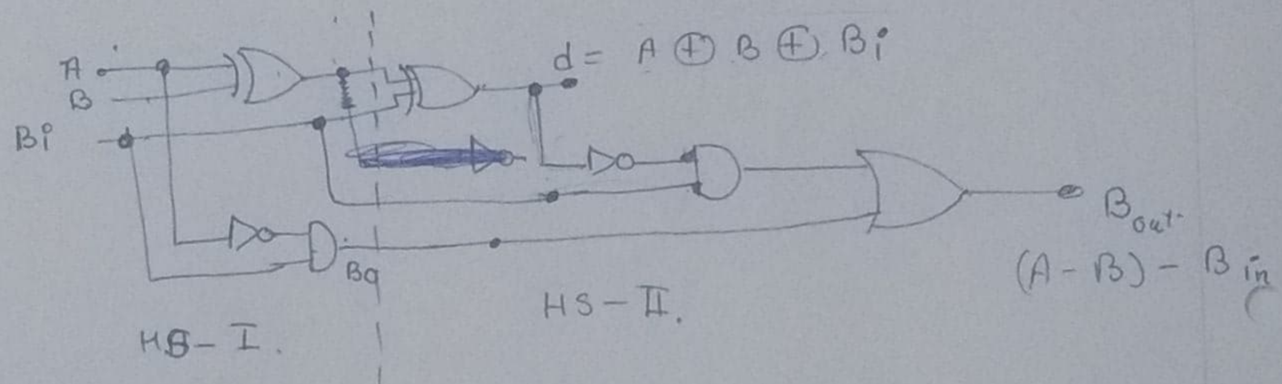
The half-subtractor can be used only LSB subtraction. If there is borrow during the subtraction of the LSB's it affects the subtraction in the next higher ~~bit~~ column; the subtrahend bit is subtracted from the minuend bit, considering the borrow from that column used for the subtraction in the preceding column.

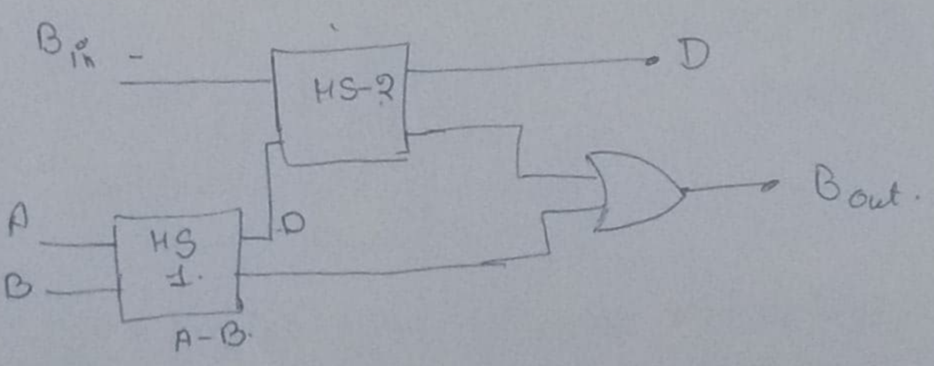
It subtracts one bit (B) from another bit (A), when already there is borrow bi from this column for the subtraction in the preceding column, and outputs the difference bit (d) and the borrow bit (b) required from the next column.

Logic circuits for the Full Subtractor:

(a) using XOR gate, (b) using And, OR and NOT gate.
(c) using universal gates (NAND-NAND, NOR-NOR).
(d) using Two Half adders.

(a) using X-OR gate or using two Half Adders:





Minimisation of circuit using K-Map.

K-Map for difference.

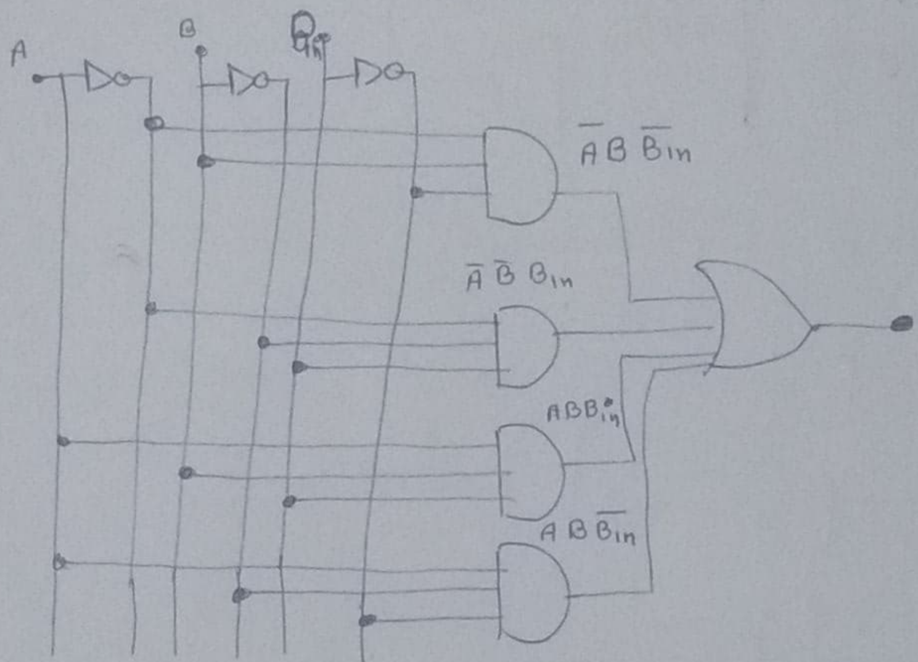
	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
$\bar{B}_{in}$	m0	m2 (1)	m6	m4 (1)
B_{in}	m1 (1)	m3	m7 (1)	m5

K-Map for Borrow.

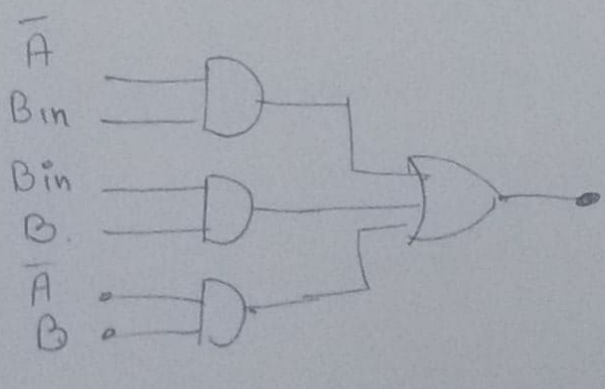
	$\bar{A}\bar{B}$	$\bar{A}B$	AB	$A\bar{B}$
$\bar{B}_{in}$	m0	m2 (1)	m6	m4
B_{in}	m1 (1)	m3 (1)	m7 (1)	m5.

$D = \bar{A}B\bar{B}_{in} + \bar{A}\bar{B}B_{in} + ABB_{in} + A\bar{B}\bar{B}_{in}$

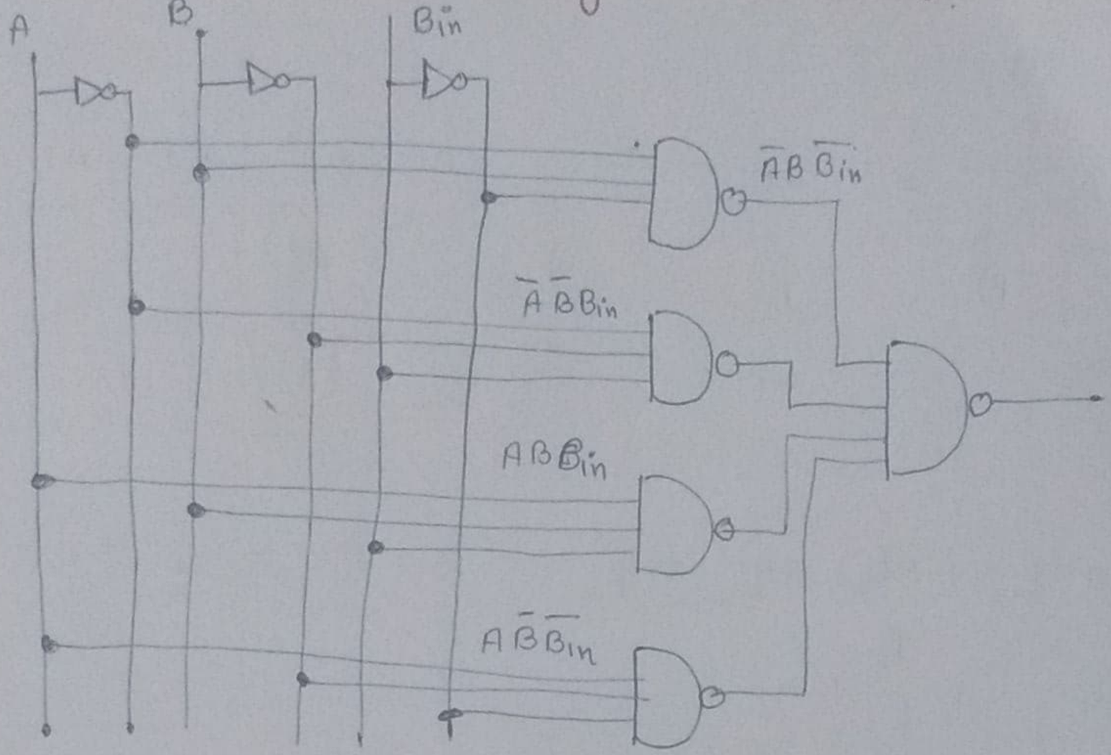
$B = \bar{A}B_{in} + B_{in}B + \bar{A}B.$



For Borrow out $(B_0) = \bar{A}B_{in} + B_{in}B + \bar{A}B$



Minimised circuit using NAND-NAND.



Logic Circuit for Borrow.