Arithmetic Circuits

The arithmetic circuits are combinational logic circuits, because the output of these circuits only depends upon their present inputs. The arithmetic circuits are used to perform arithmetic functions like addition, subtraction, multiplication, division etc. Half adder, full adder, half subtractor, full subtractor are some examples of arithmetic circuits.

Binary addition:

The rules for binary addition is

Binary addition

A	B	S (sum)	C (Carry)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Examples:

(a) Add (1001) and (1100)

$$\begin{array}{r} 1\,0\,0\,1 \\ +\,1\,1\,0\,0 \\ \hline 1\,0\,1\,0\,1 \\ \downarrow \\ \textit{Carry} \end{array}$$

(b) Add (0101) and (1111)

$$\begin{array}{r} 0101 \\ +1111 \\ \hline 10100 \\ \downarrow \\ \textit{Carry} \end{array}$$

(c) Add (0011010) and (0001100)

$$\begin{array}{r} 0\,0\,1\,1\,0\,1\,0 \\ +\,0\,0\,0\,1\,1\,0\,0 \\ \hline 0\,1\,0\,0\,1\,1\,0 \\ \downarrow \\ \textit{No Carry} \end{array}$$

(d) Add (10001) and (11101)

$$\begin{array}{r} 1\,0\,0\,0\,1 \\ +\,1\,1\,1\,0\,1 \\ \hline 1\,0\,1\,1\,1\,0 \\ \downarrow \\ \textit{Carry} \end{array}$$

Binary subtractions:

Binary subtraction is carried out in two ways:

1. Direct method using rules:

   1 – 1 = 0
   0 – 0 = 0
   1 – 0 = 1
   0 – 1 = 1 (with borrow 1)

Example:

$$\begin{array}{r} 1\,0\,1\,1\,0\,1 \\ -\,1\,0\,1\,1 \\ \hline 1\,0\,0\,0\,1\,0 \end{array}$$

2. Complement method: 1's complement and 2's complement.
   1's complement of binary number is obtained by converting each 0 into 1 and each 1 to 0. For example 1's complement of 1001 is 0110.
   2's complement of binary number is obtained by adding 1 to its 1's complement. Suppose we are asked to find 2's complement of 1011. First compute 1's complement which is equal to 0100. Next, add 1 to it to get 2's complement as 0101. Hence 2's complement of 1011 is 0101.

Binary subtractions using 1's complement:

The following steps are to be followed in binary subtraction using 1's complement:

Step-1: Convert the number to be subtracted to 1's complement form.

Spep-2: Add both the numbers.

Step-3: If final carry is 1, then add it to the result. If there is no carry, result obtained is (-)ve and in 1's complement form.

Example: (a) $(1101)_2 - (0110)_2$

Ans: The 1's complement of (0110) is (1001)

$$\begin{array}{r} \therefore\ 1\,1\,0\,1 \\ +\,1\,0\,0\,1 \\ \hline 1\,0\,1\,1\,0 \\ \downarrow\quad +\ 1 \\ \hline \textit{Carry}\ 0111 \end{array}$$

$\therefore\ (1101)_2 - (0110)_2 = (0111)_2$

(b) $(0110)_2 - (1101)_2$

Ans: The 1's complement of (1101) is (0010)

$$\begin{array}{r} \therefore\ 0\,1\,1\,0 \\ +\,0\,0\,1\,0 \\ \hline 1\,0\,0\,0 \\ \downarrow \\ \textit{No Carry} \end{array}$$

The 1's complement of (1000) is (0111)

$\therefore\ (0110)_2 - (1101)_2 = (0111)_2$

Binary subtractions using 2's complement:

The following steps are to be followed in binary subtraction using 2's complement:
Step-1: Convert the number to be subtracted to 2's complement form.
Spep-2: Add both the numbers.
Step-3: If final carry is generated i.e. 1 then discards the carry and the result is positive. If there is no final carry, result obtained is (-)Ve and in 2's complement form.

Example: (a) $(1011)_2 - (0100)_2$ using 2's complement.

Ans: The 1's complement of (0100) is (1011)
Now, the 2's complement,

$$\begin{array}{r} 1\,0\,1\,1 \\ +\quad 1 \\ \hline 1\,1\,0\,0 \end{array}$$

Therefore,

$$\begin{array}{r} 1\,0\,1\,1 \\ +\,1\,1\,0\,0 \\ \hline 1\,0\,1\,1\,1 \end{array}$$

$\downarrow$

Carry

Now, discarding the carry, we get

$$(1011)_2 - (0100)_2 = (0111)_2$$

(b) $(0101)_2 - (1011)_2$

Ans: The 1's complement of (1011) is (0100)
Now, the 2's complement,

$$\begin{array}{r} 0\,1\,0\,0 \\ +\quad 1 \\ \hline 0\,1\,0\,1 \end{array}$$

Therefore,

$$\begin{array}{r} 0\,1\,0\,1 \\ +\,0\,1\,0\,1 \\ \hline 1\,0\,1\,0 \end{array}$$

$\downarrow$

No Carry

Since, there is no carry, so 1's complement of (1010) is (0101)
Now, the 2's complement, so

$$\begin{array}{r} 0\,1\,0\,1 \\ +\quad 1 \\ \hline 0\,1\,1\,0 \end{array}$$

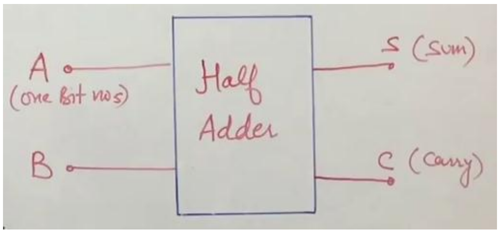
$$(0101)_2 - (1011)_2 = (0110)_2$$

Home work: Subtract the following using 2's complements

1. $(1001)_2 - (1110)_2$
2. $(10111)_2 - (1111)_2$

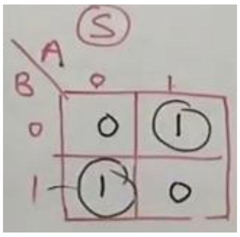
Half adder:

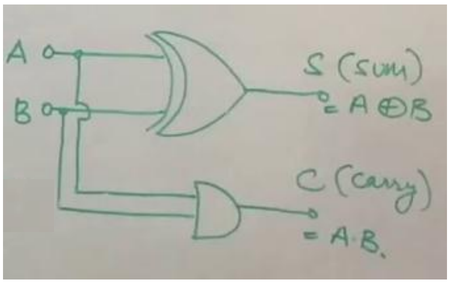
A logic circuit, which is used for adding two single bits binary number, is called half adder. It contains two inputs A and B and two outputs (sum and carry). The block diagram and truth table of half adder is shown below.



Truth Table

A	B	S (sum)	C (Carry)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1





From the truth table, the K-maps of a half adder is shown in figure-2.From the K-maps, the Boolean functions for sum (S) and carry (C) of half adder is

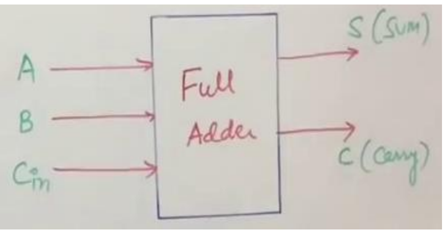
$$S = A\overline{B} + \overline{A}B = A \oplus B$$

and $$C = A.B$$

The logic diagram of a half adder is shown in figure-3.

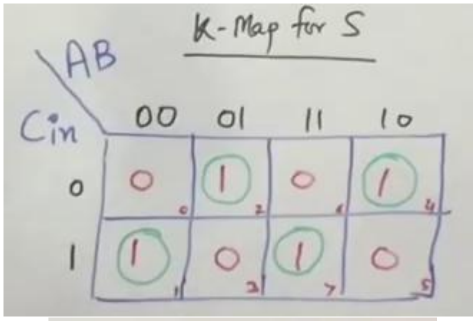
Full adder:

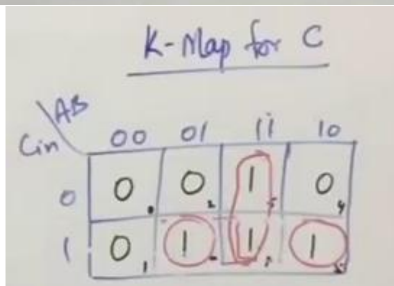
A logic circuit, which is used for adding three single bits binary number is called full adder. It contains three inputs A, B and C_{in} and two outputs (sum and carry). The block diagram and truth table of full adder is shown below.



Truth Table

A	B	C_{in}	S (sum)	C (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





From the truth table, the K-maps of a full adder for sum and carry is shown in figure-2 & 3.From the K-maps, the Boolean functions for sum (S) and carry (C) of half adder is

$$S = \bar A\bar B C_{in} + \bar A B\bar C_{in} + ABC_{in} + A\bar B\bar C_{in}$$

$$\Rightarrow S = (\bar A\bar B + AB)C_{in} + (\bar A B + A\bar B)\bar C_{in}$$

$$\Rightarrow S = \overline{\overline{(\bar A\bar B + AB)}}\, C_{in} + (A\oplus B)\bar C_{in} \qquad \because \ \bar{\bar A} = A$$

$$\Rightarrow S = \left(\overline{\overline{\bar A\bar B}\bullet\overline{AB}}\right)C_{in} + (A\oplus B)\bar C_{in} \qquad \because \ \overline{A+B} = \bar A.\bar B$$

$$\Rightarrow S = \left(\overline{(\bar{\bar A}+\bar{\bar B})\bullet(\bar A+\bar B)}\right)C_{in} + (A\oplus B)\bar C_{in} \qquad \because \ \overline{AB} = \bar A + \bar B$$

$$\Rightarrow S = \overline{((A+B)\bullet(\bar A+\bar B))}\,C_{in} + (A\oplus B)\bar C_{in}$$

$$\Rightarrow S = \overline{(A\bar A + A\bar B + \bar A B + B\bar B)}\,C_{in} + (A\oplus B)\bar C_{in}$$

$$\Rightarrow S = \overline{(0 + A\bar B + \bar A B + 0)}\,C_{in} + (A\oplus B)\bar C_{in} \qquad \because \ A\bar A = 0$$

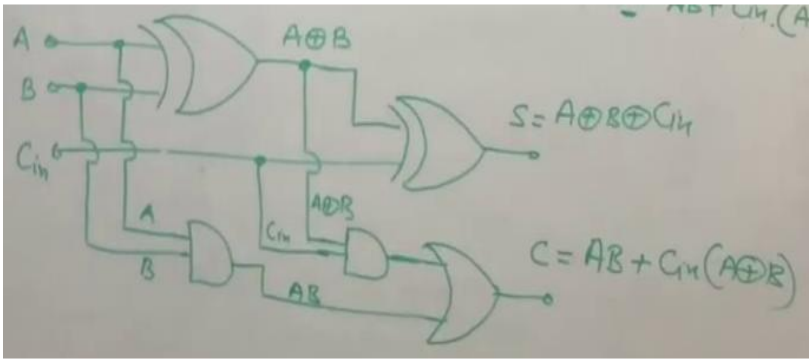
$$\Rightarrow S = \overline{(A\bar B + \bar A B)}\,C_{in} + (A\oplus B)\bar C_{in}$$

$$\Rightarrow S = \overline{(A\oplus B)}\,C_{in} + (A\oplus B)\bar C_{in}$$

$$\Rightarrow S = \bar X\, C_{in} + X\,\bar C_{in} \qquad Let,\ X = A\oplus B$$

$$\Rightarrow S = X\oplus C_{in}$$

$$\Rightarrow S = A\oplus B\oplus C_{in}$$



And the equation for carry is

$$C = AB + \bar A BC_{in} + A\bar B\,C_{in}$$

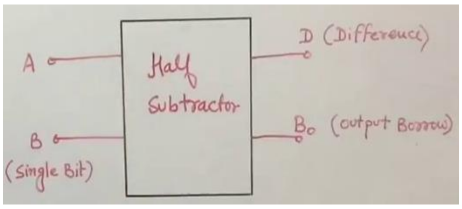
$$\Rightarrow C = AB + C_{in}(\bar A B + A\bar B)$$

$$\Rightarrow C = AB + C_{in}(A\oplus B)$$

The logic diagram of a full adder is shown in figure-4.

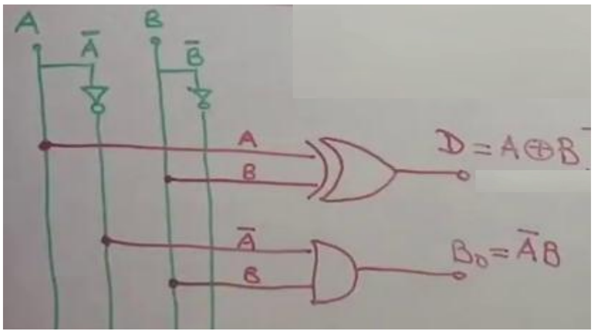
Half Subtractors:

A logic circuit, which is used for subtracting two single bits binary number, is called half subtractor. It contains two inputs A and B and two outputs (Difference and Borrow). The block diagram and truth table of a half subtractor is shown below.



Truth Table

A	B	D (Difference)	B_0 (Borrow)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



From the truth table, the Boolean functions for Difference (D) and Borrow (B_0) of half subtractor is

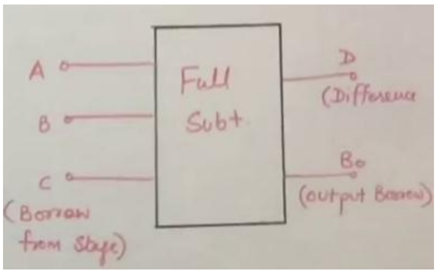
$$D = \overline{A}B + A\overline{B} = A \oplus B$$

And $$B_0 = \overline{A}B$$

The logic diagram of a half subtractor is shown in figure-3.

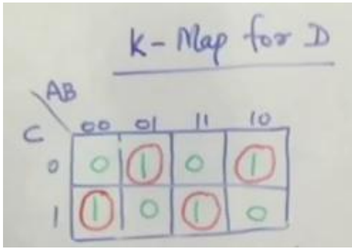
Full Subtractors:

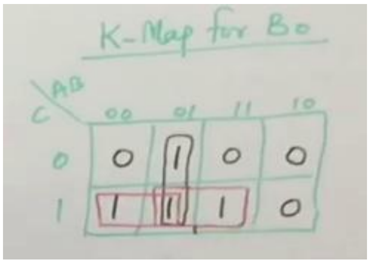
A logic circuit, which is used for subtracting three single bits binary number, is called full subtractor. It contains three inputs A, B and C and two outputs (Difference and Borrow). The block diagram and truth table of a full subtractor is shown below.



Truth Table

A	B	C	D (Difference)	B_0 (Borrow)
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





From the truth table, the K-maps of a full subtractor for difference and borrow is shown in figure-2 & 3.From the K-maps, the Boolean functions for difference (D) and borrow (B_0) of a full subtractor is

$$D = \overline{A}\overline{B}C + \overline{A}B\overline{C} + ABC + A\overline{B}\overline{C}$$
$$\Rightarrow D = \left(\overline{A}\overline{B} + AB\right)C + \left(\overline{A}B + A\overline{B}\right)\overline{C}$$

$$\Rightarrow D = \left(\overline{\overline{\overline{A}\overline{B} + AB}}\right)C + \left(A \oplus B\right)\overline{C} \qquad \because \overline{\overline{A}} = A$$

$$\Rightarrow D = \overline{\left(\overline{\overline{A}\overline{B}} \bullet \overline{AB}\right)}C + \left(A \oplus B\right)\overline{C} \qquad \because \overline{A + B} = \overline{A}.\overline{B}$$

$$\Rightarrow D = \left(\overline{(\overline{A} + \overline{B}) \bullet (A + B)}\right)C + \left(A \oplus B\right)\overline{C} \qquad \because \overline{AB} = \overline{A} + \overline{B}$$

$$\Rightarrow D = \left(\overline{\overline{A}A + \overline{A}B + \overline{B}A + \overline{B}B}\right)C + \left(A \oplus B\right)\overline{C}$$

$$\Rightarrow D = \overline{\left(0 + \overline{A}B + \overline{B}A + 0\right)}C + \left(A \oplus B\right)\overline{C} \quad \because \quad A\overline{A} = 0$$

$$\Rightarrow D = \overline{\left(A\overline{B} + \overline{A}B\right)}C + \left(A \oplus B\right)\overline{C}$$

$$\Rightarrow D = \overline{\left(A \oplus B\right)}C + \left(A \oplus B\right)\overline{C}$$

$$\Rightarrow D = \overline{X}\,C + X\,\overline{C} \qquad Let,\ X = A \oplus B$$

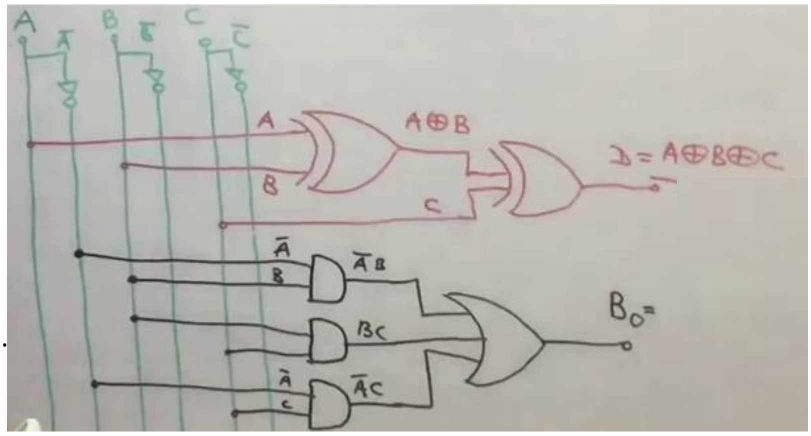
$$\Rightarrow D = X \oplus C$$

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

And the equation for borrow is

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

The logic diagram of a full subtractor is shown in figure-4.



Binary Parallel Adder: (4-bit binary adder)

A full adder is capable of adding two single bit (2 one-bit) numbers and an input carry. In order to add binary numbers with more than one bit, parallel adders are used which employ additional full adders.

A binary parallel adder is a digital circuit that adds two binary numbers in parallel form and produces the arithmetic sum of those numbers in parallel form. It consists of full adders connected in a chain, with the output carry from each full adder connected to the input carry of the next full adder in the chain. The 4-bit parallel adder is constructed by cascading 4 full adder circuits as shown in figure.

Let us consider two 4-bit binary numbers, $A = A_1, A_2, A_3, A_4$ and $B = B_1, B_2, B_3, B_4$. The interconnection of 4 full adder circuits to provide a 4 –bit parallel adder is shown in figure. The carries are connected in a chain through the full adders. The input carry to the adder is C_{in} and the output carry is C_4. The two binary numbers to be added $A = A_1, A_2, A_3, A_4$ and $B = B_1, B_2, B_3, B_4$ are applied to the corresponding inputs of full adders. This parallel adder produces their results as C_4, S_4, S_3, S_2, S_1, where C_4 is the final carry.

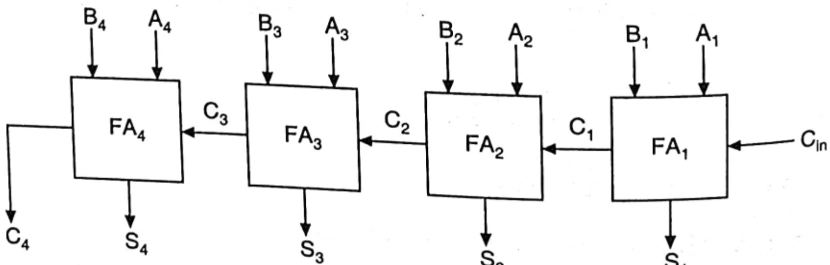


Figure 7.20 Logic diagram of a 4-bit binary parallel adder.

When the 4-bit full adder circuit is enclosed within an IC package, it has 4 terminals for the augend bits, 4 terminals for the addend bits, 4 terminals for the sum bits, and 2 terminals for the input and output carries. An n-bit parallel adder requires n-full adders.

4-bit binary Subtractors:

A binary subtractor is another type of combinational digital arithmetic circuit that produces an output which is the subtraction of two binary numbers.

The subtraction (A-B) can be done by taking the 2's complement of B and adding it to A. The 2's complement can be obtained by taking the 1's complement and adding 1 to the least significant pair of bits. The 1's complement can be implemented with inverters as shown in figure below.

In order to perform the subtraction of binary numbers with more than one bit, parallel subtractors can be used. This type of parallel subtractor can be designed in a number of ways, including combination of half and full subtractors, all full subtractors, all full adders with subtrahend complement input etc.

The figure-1 shows a 4-bit parallel binary subtractor formed by connecting four full adders with subtrahend complement input.

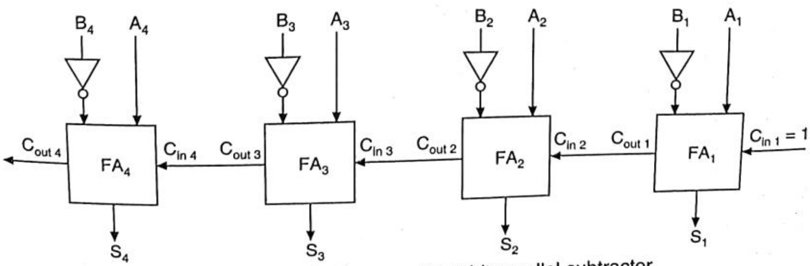
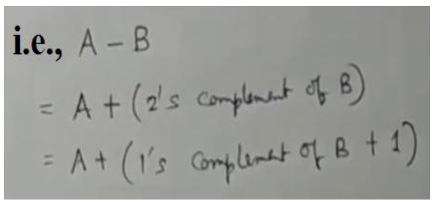


Figure 7.21 Logic diagram of a 4-bit parallel subtractor.



The figure-2 shows a 4-bit parallel binary subtractor formed by connecting one half subtractor and three full subtractors.

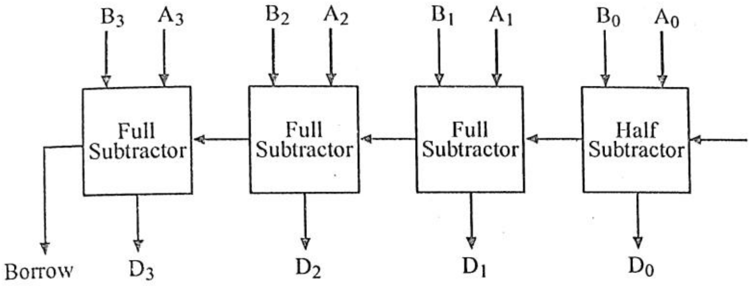


Fig. 6.10. Block diagram of 4-bit full-subtractor