Half Adder, Full Adder, Adder/Subtractor

Instructor Name: Engr. Rimsha

Design Examples

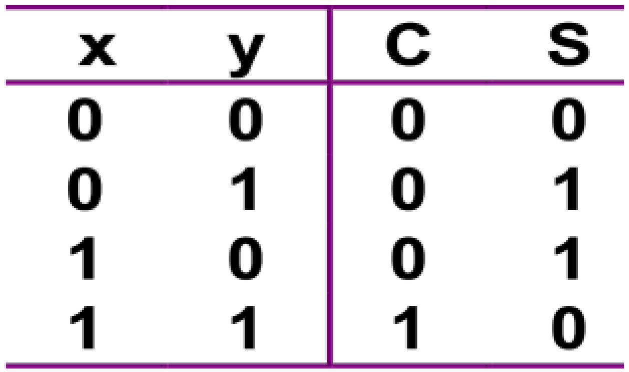
- Binary Adder-Subtractor
  - Half Adder
  - Full Adder
- Carry Lookahead Generator

Binary Adder-Subtractor

- Digital computers perform various arithmetic operations
- The most basic arithmetic operation is the addition of two binary digits
- When both augend and addend are equal to 1, the binary sum consists of two digits (1+1=10). The higher significant bit of this result is called a carry. This carry is added to the next higher order pair of significant bits.
- A combinational circuit that performs the addition of two bits is called a half adder.

Half Adder

- Half adder adds two binary bits so it requires two inputs and two outputs
  - 0+0=0 ; 0+1=1 ; 1+0=1 ; 1+1=10
- The input variable designate the augend and addend bit, the output variables produce the sum (S) and carry (C)
- The truth table for half adder is shown. The C output is 1 only when both inputs are 1. The S output represents the least significant bit of the sum

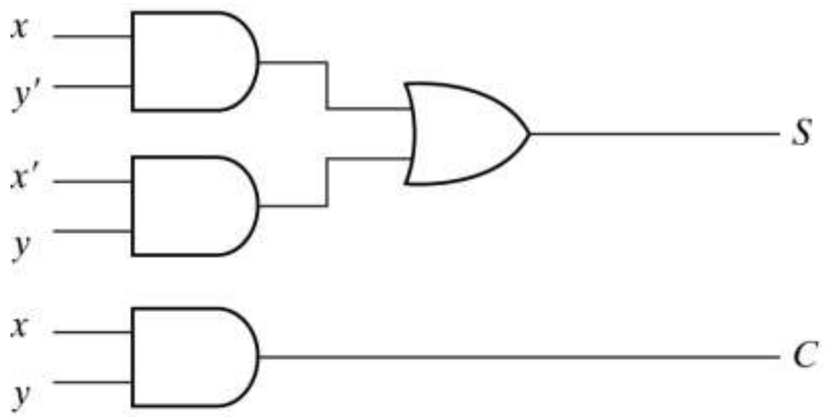


x	y	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

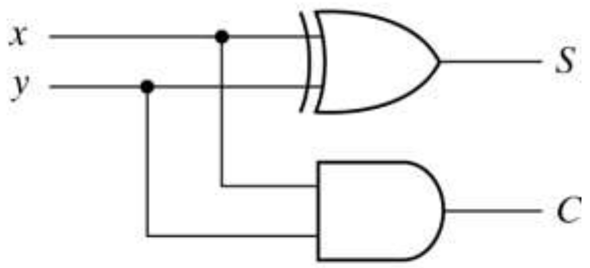
Half Adder Expressions

- The simplified sum of products expressions are:
  - $S = x'y+xy'$
  - $C = xy$
- It can be implemented in sum of products. It can also be implemented with an exclusive-OR and an AND gate
  - $S=x\oplus y$
  - $S = (x+y)(x'+y')$
  - $S' = xy+x'y'$
  - $S' = C+ x'y'$
  - $S = (C+x'y')'$
  - $C = xy$
  - $C = (x'+y')'$

Logic Diagram of Half Adder



(a) $S = xy' + x'y$
$C = xy$



(b) $S = x \oplus y$
$C = xy$

Fig. 4-5 Implementation of Half-Adder

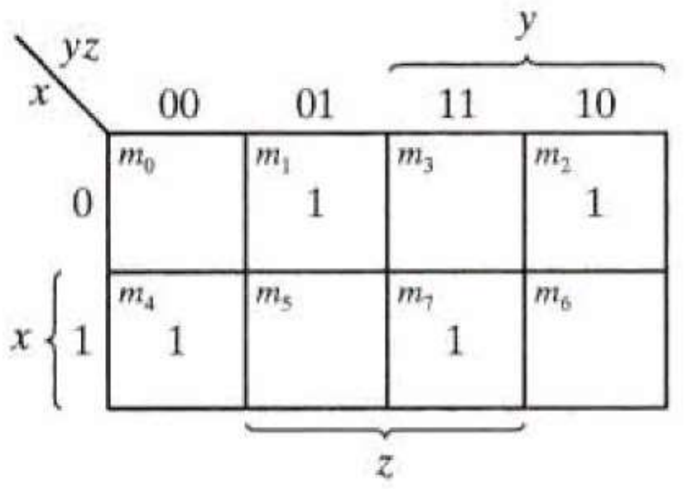
Full Adder

- A full-adder is a combinational circuit that forms the arithmetic sum of three bits (three input bits).
  - Two of the input variables x, y represents the two significant bits to be added
  - The third input z represents the carry bit from the previous lower significant position
  - Two output bits are necessary designated by the symbol S for sum and C for carry

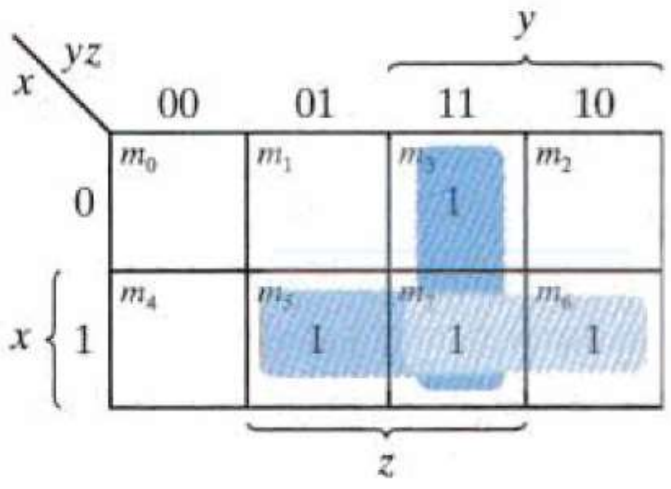
- When all input bits are 0, the output is 0
- The S output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1
- The C output has a carry of 1 if two or three inputs are equal to 1

x	y	z	c	s
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Maps for Full Adder



$S = x'y'z + x'yz' + xy'z' + xyz$



$C = xy + xz + yz$

Simplified Expressions for Full Adder

- The simplified expressions for full adder are
  - S = x' y' z + x' y z' + x y' z' + x y z
  - C = xy + xz + yz

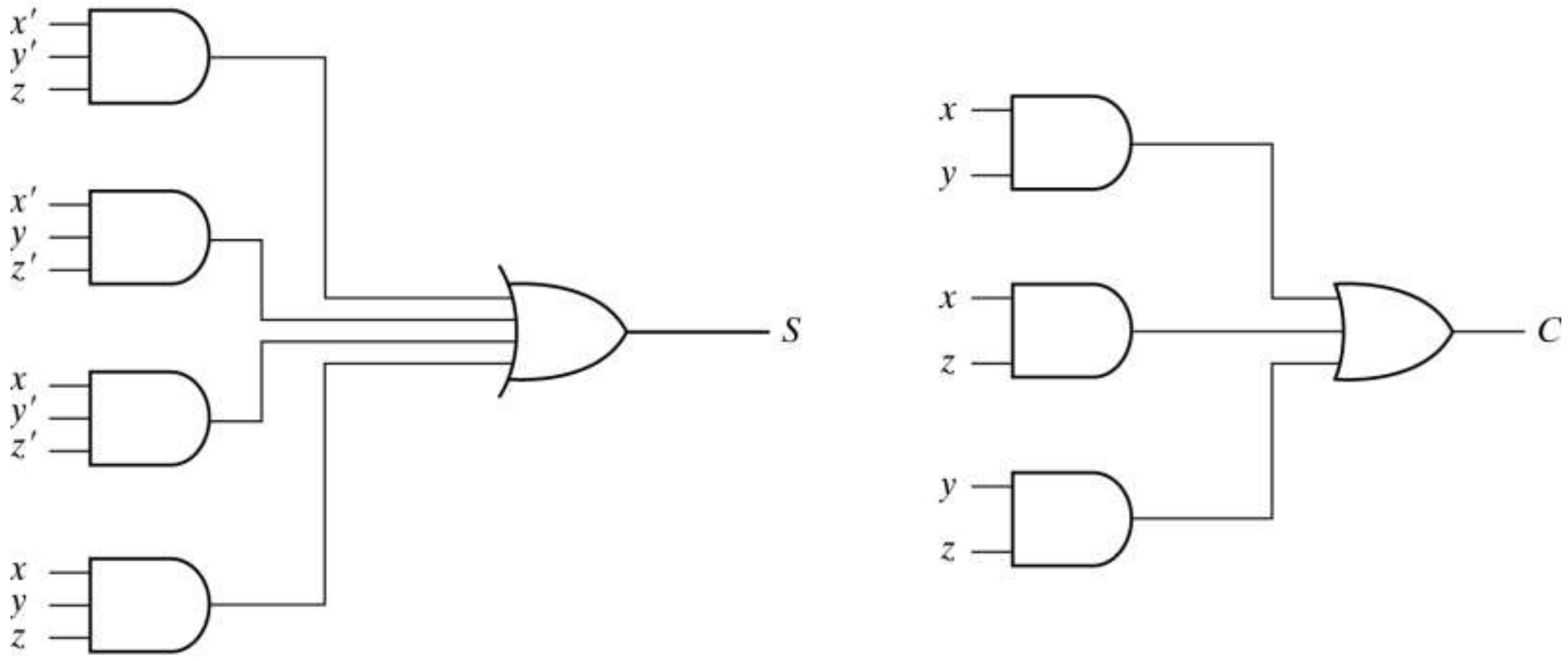


Fig. 4-7 Implementation of Full Adder in Sum of Products

Full Adder with Two Half Adders

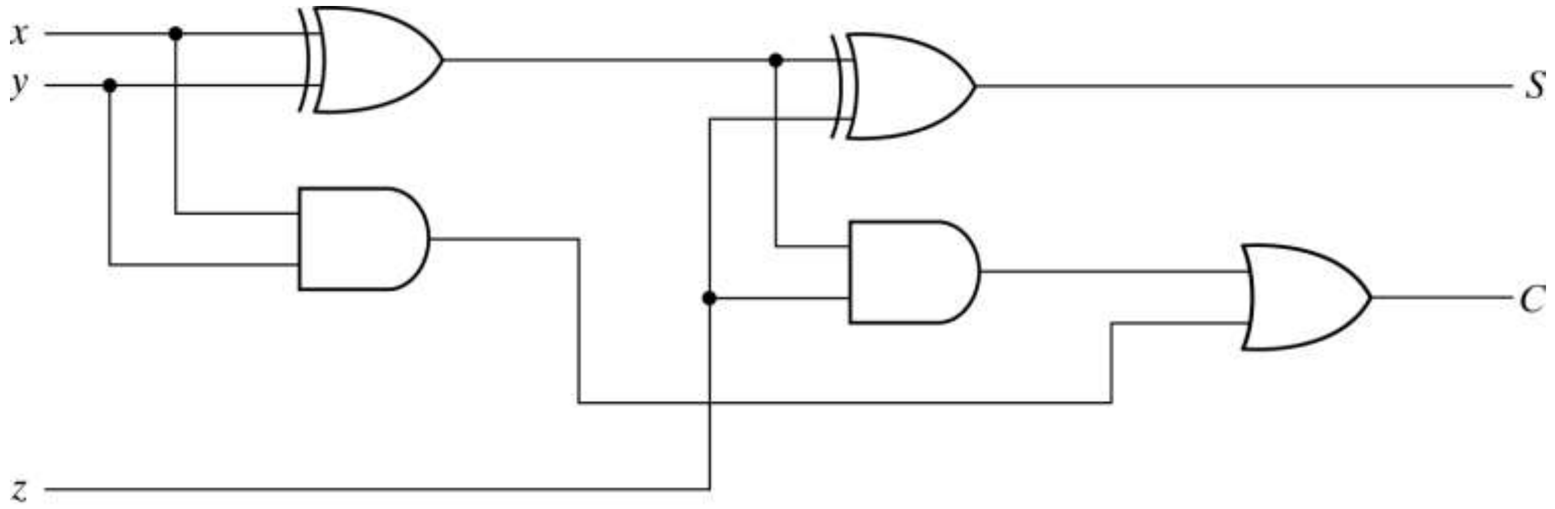
- Full adder can also be implemented with two half adders and an OR gate. The S output from the second half adder is the exclusive-OR of z and the output of the first half adder giving
  - $S = z \oplus (x \oplus y)$

    $= z'(xy'+x'y)+z(xy'+x'y)'$

    $=z'(xy'+x'y)+z(xy+x'y')$

    $= xy'z'+x'yz'+xyz+x'y'z$
  - $C = z(xy'+x'y)+xy$

    $= xy'z+x'yz+ xy$



Binary Adder

- A binary adder is a digital circuit that produces the arithmetic sum of two binary numbers.
- A binary adder can be implemented using multiple full adders (FA) connected in cascade with the output carry from each full adder to the input carry of the next full adder in the chain

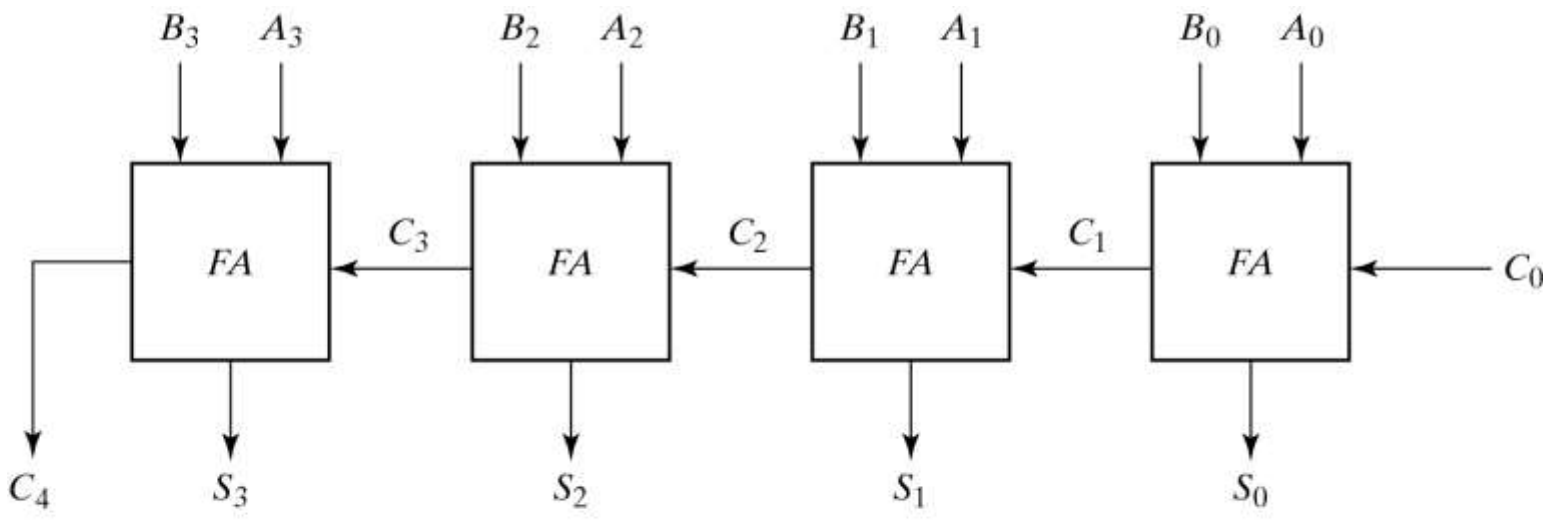
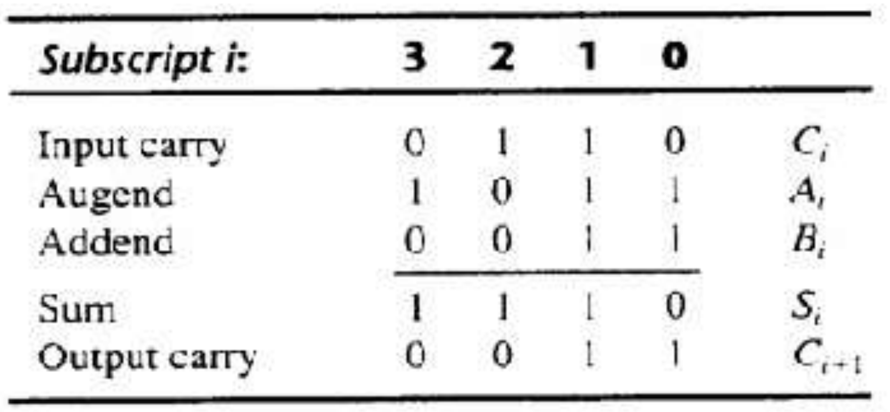


Fig. 4-9 4-Bit Adder

Binary Adder

- The augend bits of A and the addend bits of B are designated by subscript numbers from right to left, with subscript 0 denoting the least significant bit
- The carries are connected in a chain through the full adders
- The S outputs generate the required sum bits



Subscript i:	**3**	**2**	**1**	**0**	
Input carry	0	1	1	0	C_i
Augend	1	0	1	1	A_i
Addend	0	0	1	1	B_i
Sum	1	1	1	0	S_i
Output carry	0	0	1	1	C_{i+1}

Binary Adder

- Consider the two binary numbers, A= 1011 and B= 0011
- Their sum S= 1110 is formed with four-bit adders
- The bits are added with full adders, starting from the least significant position (subscript 0), to form the sum bit and carry bit
- The input carry C_0 in the least significant position must be 0
- The value of C_{i+1} in a significant position is the output carry of the full adder
- This value is transferred into the input carry of the full adder that adds the bits one higher significant position to the left

Subscript i	3	2	1	0	
Input carry	0	1	1	0	C_i
Augend	1	0	1	1	A_i
Addend	0	0	1	1	B_i
Sum	1	1	1	0	S_i
Output carry	0	0	1	1	C_{i+1}

Binary Adder

- The sum bits are thus generated starting from the rightmost position and are available as soon as the corresponding previous carry bit is generated
- All the carries must be generated for the correct sum bits to appear at the outputs

Subscript i	3	2	1	0	
Input carry	0	1	1	0	C_i
Augend	1	0	1	1	A_i
Addend	0	0	1	1	B_i
Sum	1	1	1	0	S_i
Output carry	0	0	1	1	C_{i+1}

Binary Adder

- A binary adder is a digital circuit that produces the arithmetic sum of two binary numbers.
- A binary adder can be implemented using multiple full adders (FA) connected in cascade with the output carry from each full adder to the input carry of the next full adder in the chain

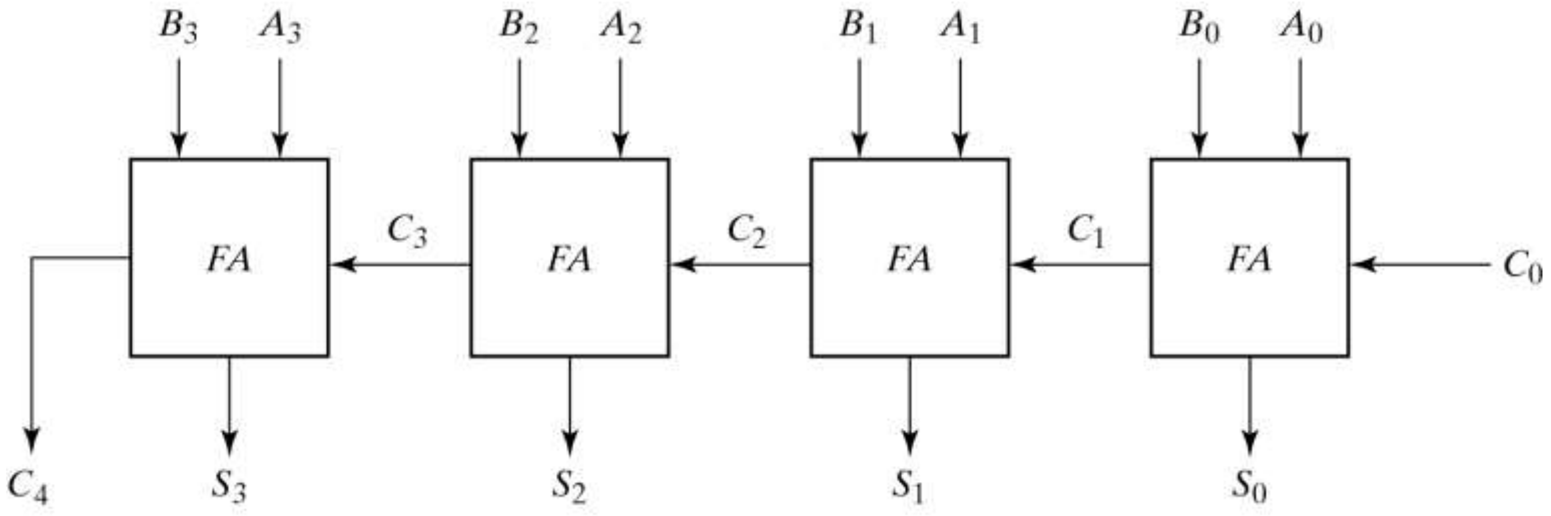


Fig. 4-9 4-Bit Adder

The End