Arithmetic Circuits I

CS207 Lecture 11

James YU

Apr. 29, 2020

南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Arithmetic circuits

- One important aspect of digital design not dealt with in earlier lectures is the design and implementation of arithmetic circuits.
  - Various information-processing jobs are carried out by digital computers.
  - Arithmetic operations are among the basic functions of a digital computer.

Addition

- Addition of two binary digits is the most basic arithmetic operation.
  - $0 + 0 = 0$,
  - $0 + 1 = 1$,
  - $1 + 0 = 1$,
  - $1 + 1 = 10$.
  - The higher significant bit of this result is called the *carry*.
- A combinational circuit that performs the addition of two bits as described above is called a *half-adder*.
- The addition operation involves three bits — the *augend bit*, *addend bit*, and the *carry bit* and produces a sum result as well as carry.
- The combinational circuit performing this type of addition operation is called a *full-adder*.

Half-adder

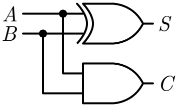
- As described above, a half-adder has two inputs and two outputs.
- Let the input variables augend and addend be designated as A and B, and output functions be designated as S for sum and C for carry.

Input variables		Output variables	
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

- It can be seen that the outputs S and C functions are similar to Exclusive-OR and AND functions, respectively.

Half-adder

- $S = A \oplus B$,
- $C = AB$.



Full-adder

- A combinational circuit of full-adder performs the operation of addition of three bits — the augend, addend, and previous carry X, and produces the outputs sum and carry.

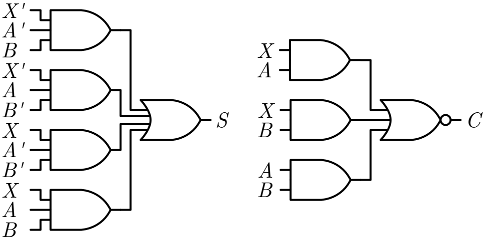
Input variables			Output variables	
X	A	B	S	C
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Full-adder

X \ AB	00	01	11	10
0		1		1
1	1		1	

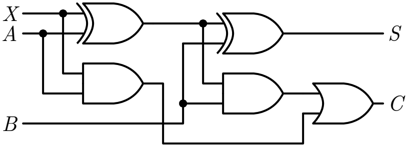
X \ AB	00	01	11	10
0			1	
1		1	1	1

- $S = X'A'B + X'AB' + XA'B' + XAB$,
- $C = AB + BX + AX$.



Full-adder

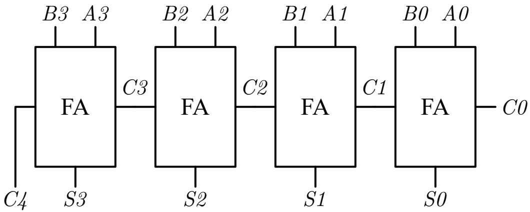
- It can also be implemented with two half adders and one OR gate, as shown below.



$$S = B \oplus (X \oplus A) = B'(XA' + X'A) + B(XA' + X'A)'$$
$$= B'(XA' + X'A) + B(XA + X'A') = XA'B' + X'AB' + XAB + X'A'B.$$
$$C = B(XA' + X'A) + XA = XA'B + X'AB + XA$$

Binary adder

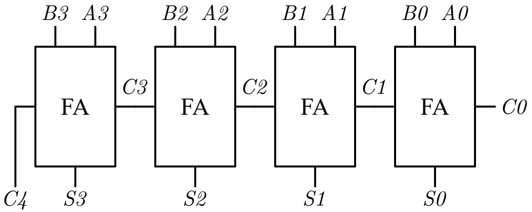
- A binary adder is a digital circuit that produces the arithmetic sum of two binary numbers.
- It can be constructed with full adders connected in cascade, with the output carry from each full adder connected to the input carry of the next full adder in the chain.
- Addition of n-bit numbers requires a chain of n full adders or a chain of one-half adder and $n-1$ full adders.
  - Below shows the interconnection of four full-adder (FA) circuits to provide a four-bit binary ripple carry adder.



Binary adder

- $1011 + 0011 = 1110.$

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}

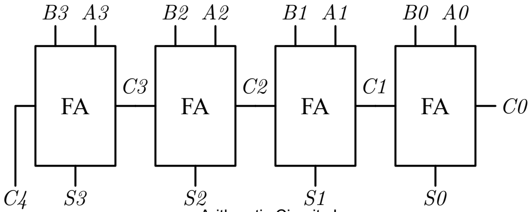


Binary adder

- The four-bit adder is a typical example of a standard component.
- It can be used in many applications involving arithmetic operations.
- Observe that the design of this circuit by the classical method would require a truth table with $2^9 = 512$ entries, since there are nine inputs to the circuit.
- By using an iterative method of cascading a standard function, it is possible to obtain a simple and straightforward implementation.

Carry propagation

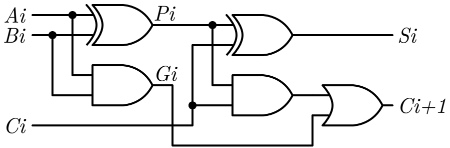
- The addition of two binary numbers in parallel implies that all the bits of the augend and addend are available for computation at the same time.
- As in any combinational circuit, the signal must propagate through the gates before the correct output sum is available in the output terminals.
  - The total propagation time is equal to the propagation delay of a typical gate, times the number of gate levels in the circuit.
  - In this regard, consider output S_3. Inputs A_3 and B_3 are available as soon as input signals are applied to the adder.
  - However, input carry C_3 does not settle to its final value until C_2 is available from the previous stage.



Carry propagation

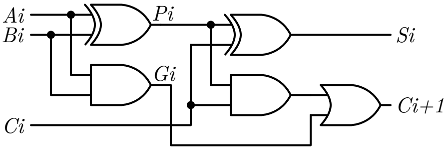
- The carry propagation time is an important attribute of the adder because it limits the speed with which two numbers are added.
  - Since all other arithmetic operations are implemented by successive additions, the time consumed during the addition process is critical.
- A solution is to increase the complexity of the equipment in such a way that the carry delay time is reduced.
- The most widely used technique employs the principle of *carry lookahead logic*.

Carry propagation



- Consider this full-adder circuit:
  - $P_i = A_i \oplus B_i$,
  - $G_i = A_i B_i$.
- The output sum and carry can respectively be expressed as
  - $S_i = P_i \oplus C_i$,
  - $C_{i+1} = G_i + P_i C_i$.
- G_i is called a *carry generator*. P_i is called a *carry propagator*.

Carry propagation



- We now write the Boolean functions for the carry outputs of each stage and substitute the value of each C_i from the previous equations.

$$
\begin{aligned}
C_0 &= \text{input carry}, \\
C_1 &= G_0 + P_0C_0, \\
C_2 &= G_1 + P_1C_1 = G_1 + P_1G_0 + P_1P_0C_0, \\
C_3 &= G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0C_0.
\end{aligned}
$$

Carry propagation