Addition

Lecture 20
Section C.5

Robb T. Koether

Hampden-Sydney College

Wed, Oct 16, 2019

Outline

Binary Addition

8-Bit Addition

```
 0010 1110
+0100 1001
----------
 0111 0111
```

- We apply the grade-school addition algorithm to the binary number system.

Binary Addition

One-Bit Addition

$$
\begin{aligned}
0 + 0 &= 0, \\
0 + 1 &= 1, \\
1 + 0 &= 1, \\
1 + 1 &= 10.
\end{aligned}
$$

- The rules are very simple.
- Note that in the last case we carry a 1.

One-Bit Addition

One-Bit Addition

a	b	s	c
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

- Adding two bits produces a sum bit s and a carry bit c.
- Do you recognize the sum and carry as logical functions?

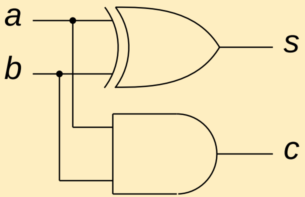
Outline

The Half Adder

- A half adder is an electronic circuit that receives two bits as input, adds them, and produces the sum and carry bits.
- The sum bit is $s = a$ XOR b.
- The carry-out bit is $c = a$ AND b.

The Half Adder

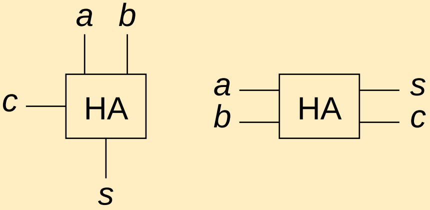
The Half Adder



- A half adder is an electronic circuit that receives two bits as input, adds them, and produces the sum and carry bits.
- The sum bit is $s = a$ XOR b.
- The carry-out bit is $c = a$ AND b.

The Half Adder

The Half Adder



- We will represent a half adder as shown above.

Outline

Multi-column Addition

8-Bit Addition

```
00001 000 (carry)
 0010 1110
+0100 1001
----------
 0111 0111
```

- Adding more than single bits requires more than a half adder.
- In addition to the two bits in each column, there is the carry bit from the previous column.
- Now each column has a carry-in c_{in} and a carry-out c_{out}.

Multi-column Addition

8-Bit Addition

```
00001 0000(carry)
 0010 1110
+0100 1001
----------
 0111 0111
```

- Adding more than single bits requires more than a half adder.
- In addition to the two bits in each column, there is the carry bit from the previous column.
- Now each column has a carry-in c_{in} and a carry-out c_{out}.

Multi-column Addition

One-Bit Addition with Carry-in

a	b	c_{in}	s	c_{out}
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Binary Addition

One-Bit Addition

$$
\begin{aligned}
s &= a\overline{b}\overline{c} + \overline{a}b\overline{c} + \overline{a}\overline{b}c + abc \\
&= (a\overline{b} + \overline{a}b)\overline{c} + (\overline{a}\overline{b} + ab)c \\
&= (a\overline{b} + \overline{a}b)\overline{c} + \overline{(a\overline{b} + \overline{a}b)}c \\
&= (a \oplus b)\overline{c} + \overline{(a \oplus b)}c \\
&= (a \oplus b) \oplus c
\end{aligned}
$$

- The table allows us to write a Boolean expression for s.
- "$\oplus$" is the symbol for XOR.

Binary Addition

One-Bit Addition

$$
\begin{aligned}
s &= a\overline{b}\overline{c} + \overline{a}b\overline{c} + \overline{a}\overline{b}c + abc \\
&= (a\overline{b} + \overline{a}b)\overline{c} + (\overline{a}\overline{b} + ab)c \\
&= (a\overline{b} + \overline{a}b)\overline{c} + \overline{(a\overline{b} + \overline{a}b)}c \\
&= (a \oplus b)\overline{c} + \overline{(a \oplus b)}c \\
&= (a \oplus b) \oplus c \\
&= s_{(a,b)} \oplus c.
\end{aligned}
$$

- The table allows us to write a Boolean expression for s.
- "$\oplus$" is the symbol for XOR.

Binary Addition

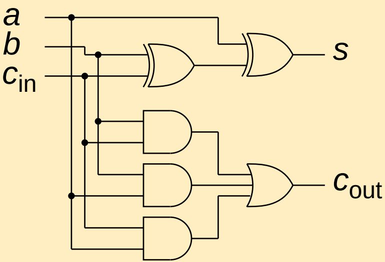
One-Bit Addition

$$
\begin{aligned}
c_{\text{out}} &= ab\overline{c} + a\overline{b}c + \overline{a}bc + abc \\
&= (abc + ab\overline{c}) + (abc + a\overline{b}c) + (abc + \overline{a}bc) \\
&= ab(c + \overline{c}) + ac(b + \overline{b}) + bc(a + \overline{a}) \\
&= ab + ac + bc.
\end{aligned}
$$

- We can write a Boolean expression for c_{out}.

The Full Adder

The Full Adder



- The above circuit implements the Boolean expressions.

Binary Addition

One-Bit Addition

$$
\begin{aligned}
c_{\text{out}} &= ab\overline{c} + a\overline{b}c + \overline{a}bc + abc \\
&= (abc + ab\overline{c}) + (a\overline{b}c + \overline{a}bc) \\
&= ab(c + \overline{c}) + (a\overline{b} + \overline{a}b)c \\
&= ab + (a \oplus b)c
\end{aligned}
$$

- Or we can simplify it this way.

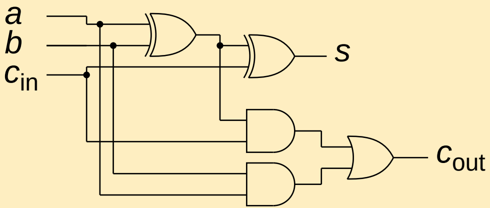
Binary Addition

One-Bit Addition

$$
\begin{aligned}
c_{\text{out}} &= ab\overline{c} + a\overline{b}c + \overline{a}bc + abc \\
&= (abc + ab\overline{c}) + (a\overline{b}c + \overline{a}bc) \\
&= ab(c + \overline{c}) + (a\overline{b} + \overline{a}b)c \\
&= ab + (a \oplus b)c \\
&= c_{(a,b)} + s_{(a,b)}c.
\end{aligned}
$$

- Or we can simplify it this way.

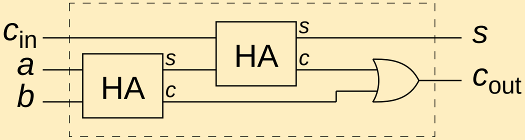
The Full Adder



- The second form uses fewer gates.

The Full Adder

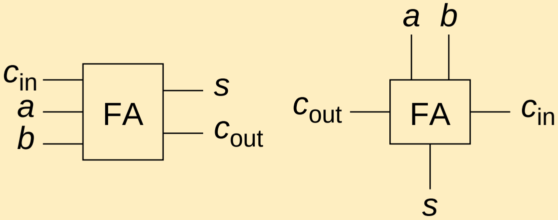
The Full Adder



- The simplified circuit can be implemented with two half adders.
- We call this a full adder.

The Full Adder

The Full Adder

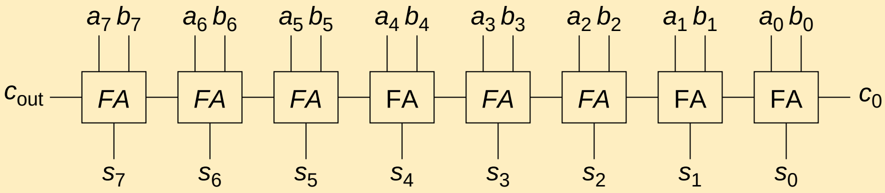


- We can represent it in a variety of ways.

Outline

The Ripple Adder

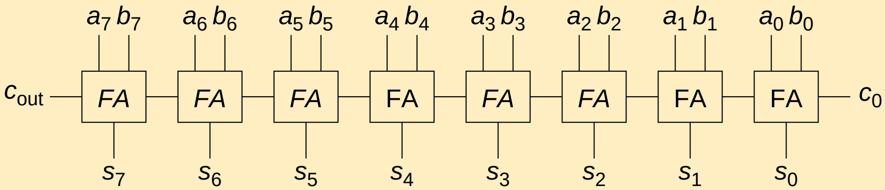
The Ripple Adder



- We can implement multi-column binary addition by using a full adder in each column.
- As one full adder computes its carry-out bit, that bit is used as the carry-in bit of the next full adder.
- This is called a ripple adder.

The Ripple Adder

The Ripple Adder



- How many levels of logic (gates) are required by an 8-bit ripple adder?
- How many levels of logic (gates) are required by a 32-bit ripple adder?
- Can we do better?

Outline

Assignment

Assignment

- Read Section C.5.