# 3.27 Adders

Many digital circuits, like in a calculator or computer, must add binary numbers. For decimal numbers, adding by hand start and adds each digit, possibly carrying a 1 to the next digit. Adding in binary is identical.

**PARTICIPATION ACTIVITY** 3.27.1: Adding in binary.

Start ☐ 2x speed

```
  1  1
  0  1  1  1
+ 0  1  1  0
-------------
  1  1  0  1
```

(So 7 + 6 = 13)

**PARTICIPATION ACTIVITY** 3.27.2: Adding binary numbers.

1)
```
  0010
+ 0010
----------
```

Check Show answer

2) 0110
+ 0010
---

Check Show answer

3) 0101
+ 0111
---

Check Show answer

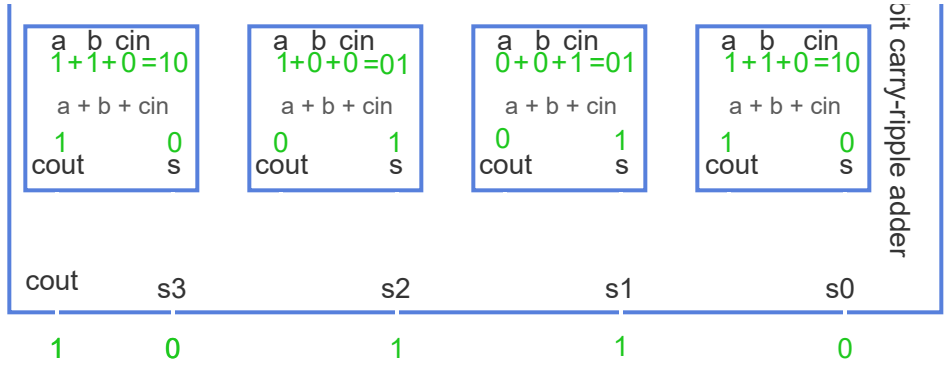
4) 1111
+ 0001
---

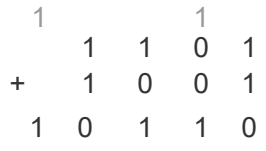
Check Show answer

An ***adder*** computes A + B, where A and B are N-bit numbers, such as 8-bit numbers. A ***carry-ripple adder*** mimics adding by digit's pair of bits and carry-in bit, and generating a sum and carry-out bit.

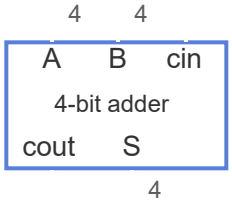
**PARTICIPATION ACTIVITY** 3.27.3: A carry-ripple adder.

Start 2x speed



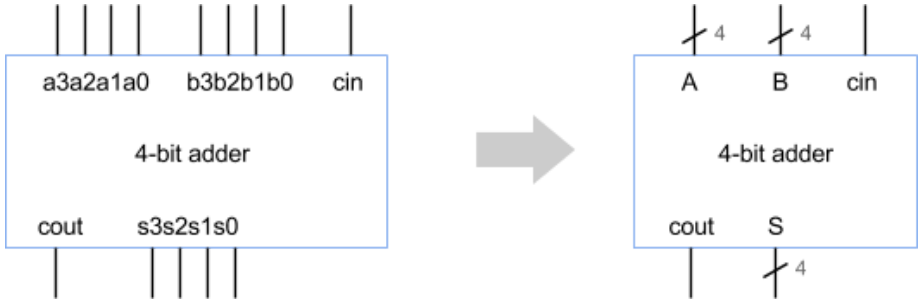




Block symbol




A datapath component is commonly represented as a block symbol. This material generally uses uppercase letters to repre data and lowercase letters to represent single-bit data. A multi-bit wire is drawn as a single wire with a slash (/), as shown b

Figure 3.27.1: Block symbol: 4-bit adder.




**PARTICIPATION ACTIVITY** 3.27.4: Carry-ripple adders.

1) A 12-bit carry-ripple adder adds two ?-bit numbers.
   - 6
   - 12
   - 13

2) Each digit's pair of bits and carry-in bit are added simultaneously.
   - False
   - True

3) A = 1111, B = 0001
   - cout = 0
     s3s2s1s0 = 0000
   - cout = 1
     s3s2s1s0 = 1111
   - cout = 1
     s3s2s1s0 = 0000

A ***full adder*** is a circuit that adds three bits and generates a sum and carry-out. Four full adders were used above. The strar adder" is historical, intended to contrast with a ***half adder***, which is a circuit that adds two bits and generates a sum and ca adder can be designed starting from a truth table.

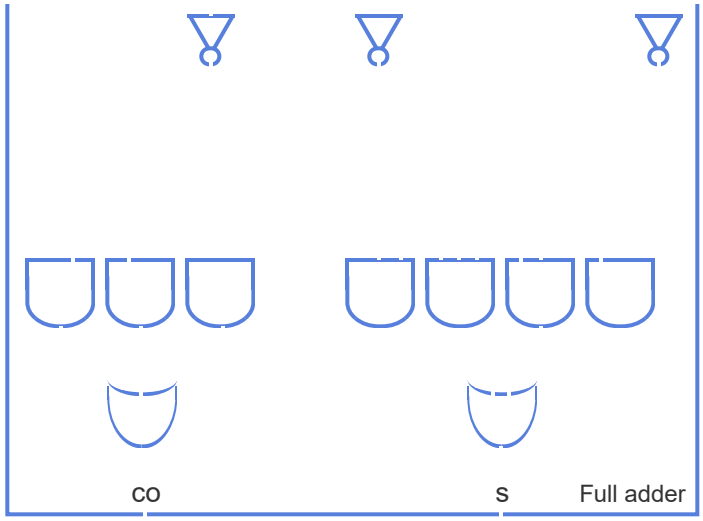
**PARTICIPATION ACTIVITY** 3.27.5: Full adder.

Start ☐ 2x speed

ci a b co 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
```

```
   0     1 0     1 1
   0       1       1
 + 0     + 1     + 1
 ------  ------  ------
 0 0     1 0     1 1
```




co = ci'ab + cia'b + ciab' + ciab
co = ab + cia + cib

s = ci'a'b + ci'ab' + cia'b' + ciab

**PARTICIPATION ACTIVITY** 3.27.6: Full adders.

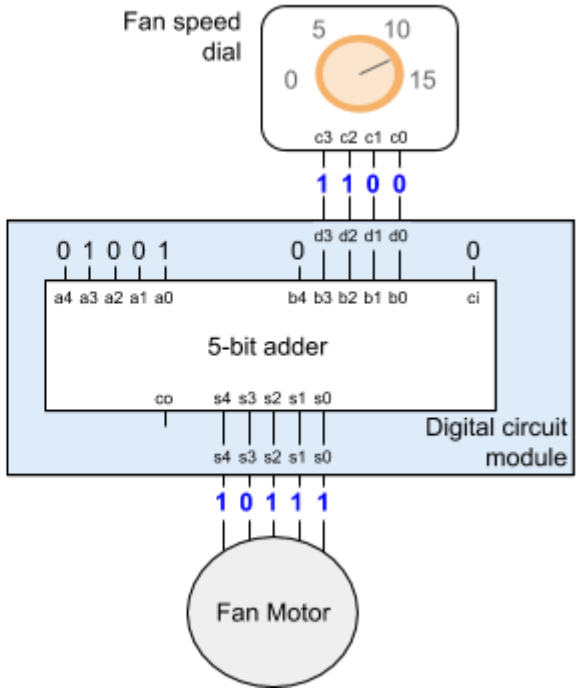
1) A circuit that adds three bits and generates a sum and carry-out is know as a _____.
   - ○ Half adder
   - ○ Full adder

2) How many full adders are needed to create a 7-bit carry-ripple adder?
   - ○ 7
   - ○

14

## Example 3.27.1: Fan speed adjustment using an adder.

An electronic room fan has a digital circuit module. The module's 5-bit output S controls the fan's motor rotation speed: 0 means no rotation, 11111 means fastest rotation. The module's 4-bit input D comes from a dial. When on, the fan's slowest speed is S = 9 (01001). Turning the dial increases D anywhere from 0(0000) to 15 (1111). The output S should be set as: S = 9 + D.




**PARTICIPATION ACTIVITY** 3.27.7: Fan speed adjustment.

Consider the fan speed adjustment example.

1) What is the greatest possible value of

S? Give answer in decimal.

Check Show answer

2) What is the greatest possible value of S? Give answer in binary.

Check Show answer

3) Suppose the fan had an electronic on-off switch. When off, S should output what value? Give answer in decimal.

Check Show answer

An ***incrementer*** adds 1 to a number, which is a common operation. An incrementer can be built by inputting 00..00 to an ad and 1 to the carry-in. Such a circuit is unnecessarily large, because each digit's full adder can add 3 bits (a, b, ci), but b is alv ***adder*** adds two bits (a, b), and is sufficient for an incrementer.

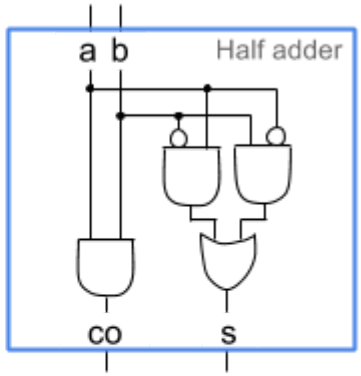
Figure 3.27.2: Half adder truth table equations, and circuit.

| a | b | co | s |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

co = ab

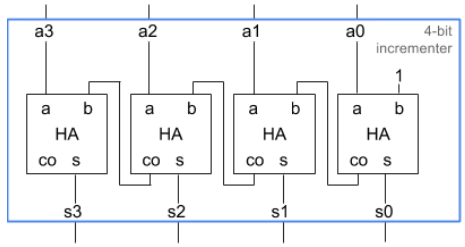
s = a'b + ab'

(a) Half adder truth table

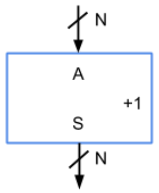



(b) Half adder (HA) circuit

Figure 3.27.3: Incrementer circuit and block diagram.




(a) 4-bit incrementer constructed with half adders




(b) Incrementer block diagram

**PARTICIPATION ACTIVITY** 3.27.8: Half adders.

1) A half adder adds two bits.
   - True
   - False

2) A half adder generates a carry-out if any input is 1.
   - True
   - False

3) A half adder circuit requires fewer gates than a full adder circuit.
   - True
   - False

4) An incrementer adds 1 to input A.
   - True
   - False

5) Full adders can be used to construct an incrementer.

- ○ True
- ○ False

6) A 4-bit carry-ripple adder can be constructed using 4 half adders.

- ○ True
- ○ False

**Provide feedback on this section**