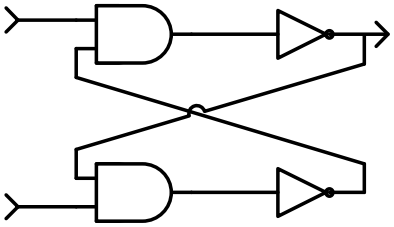
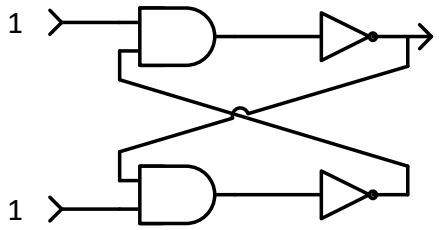
QUIZ 1Answers

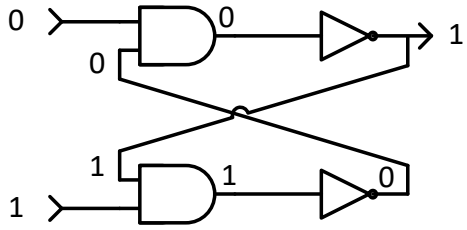
1 Assume that both of the inputs in the following circuit are 1.
Describe what would happen if the upper input were temporarily changed to 0.
Describe what would happen if the lower input were temporarily changed to 0.



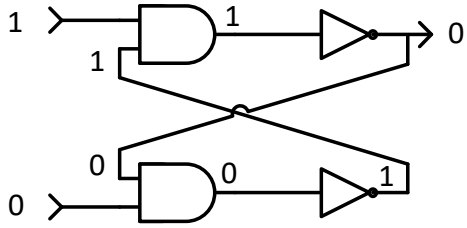
Answer:



Both of the inputs are 1, then the state of the flip-flop is untouched.



If the upper input were temporarily changed to 0, the state of the flip-flop is set to 1. This step is termed Set.



If the lower input were temporarily changed to 0, the state of the flip-flop is set to 0. This step is termed Reset.

Reference: http://en.wikipedia.org/wiki/Flip-flop_(electronics)

2. Suppose a digital camera has a storage capacity of 256MB. How many photographs could be stored in the camera if each consisted of 1024 pixels per row and 1024 pixels per column if each pixel required three bytes of storage?Write down the details of each step.

Answers:

One photo contains 1024 * 1024 pixels. One pixel takes 3 bytes, then one photo takes 1024 * 1024 *3 bytes.

256MB = 256 * 1024 * 1024 bytes

Therefore, the camera can store $\frac{256*1024*1024}{1024*1024*3} \approx 85$ photos.

3. Solve the following problems by translating the values into 2's complement notation (5bit):

5–1

5+12

Verify the result using ten-based notation, write down the details of each step and your discoveries.

Answers:

5- 1

Convert 5- 1 to binary: 00101 – 00001.

Convert 00101 to 2's complement notation: 00101

Convert -00001 to 2's complement notation: 11111 (Complement and plus one, or copy and complement or subtracted by 100000(2^5))

Add 00101 and 11111 = 00100 which is 4

5+12

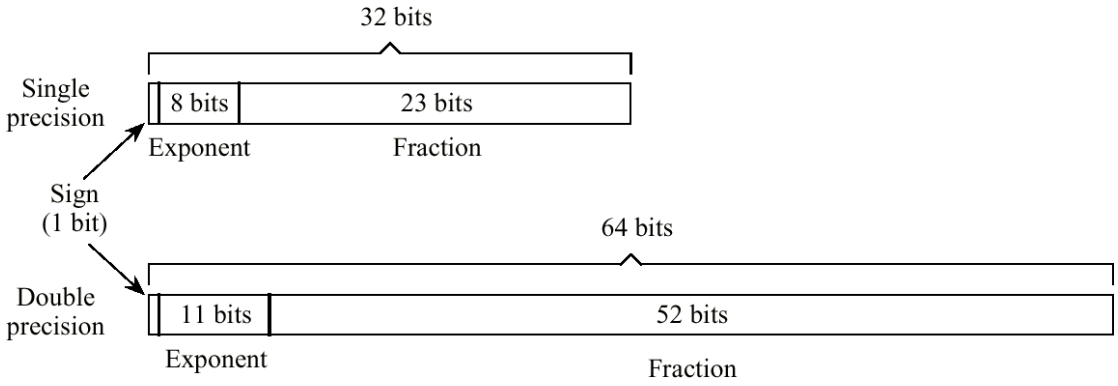
Convert 5+12 to binary: 00101 + 01100.

Convert 00101 to 2's complement notation: 00101

Convert 01100 to 2's complement notation: 01100

Add 00101 and 01100 = 10001 cause overflow! (The erroneous result is –15).

4



The binary numbers of $3\frac{3}{8}$, $-2\frac{1}{2}$ are 11.011 and -10.1 respectively.

Finally, if translating two fractions into IEEE754 single precision notation:

11.011 -> 0(sign bit) .1011 (mantissa) 1 (exponent) ->

0 10000000 10110000000000000000000

-10.1 ->1 (sign bit) .01 (mantissa) 1 (exponent) ->

1 10000000 01000000000000000000000

Reference: an IEEE754 single precision number is calculated as:

$(-1)^{sign\ bit} * (1 + mantissa) * 2^{(\text{exponent}-127)}$