CS 240 SI Worksheet
Valerie Zhao
2/16/17

Bitwise Operations + Sequential Logic (Part 1)

1. Implement a C function that takes in 2 integers (x and y) and **return 1 if x < y, else 0.** Use as few operators as possible, and **only from the following**: **`! ~ & ^ | + << >>`** Numbers are allowed.

   Before you start: what test cases (sample values for x and y) would you use to prove that your implementation works? After implementing the function, simulate it on your test cases.

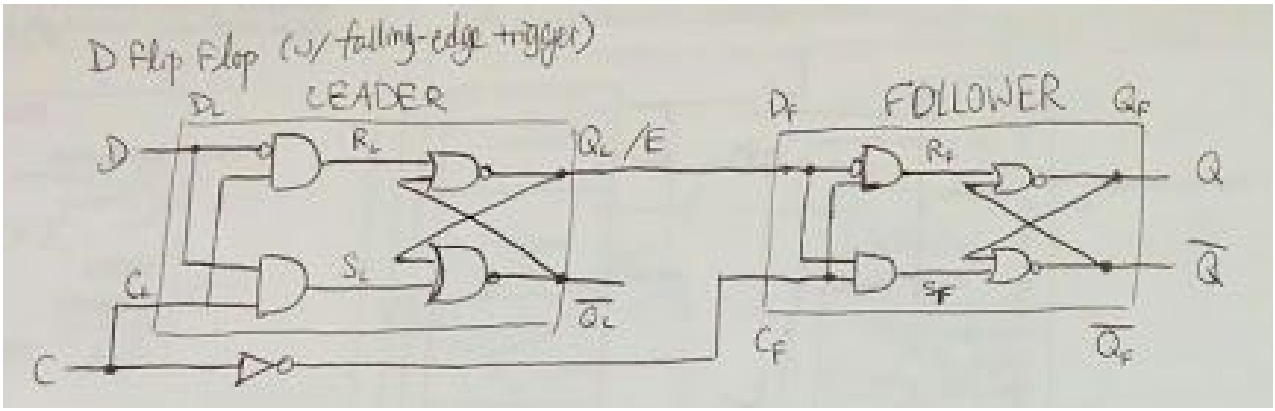
   -x and +y
   +x and -y
   x and y with the same signs
   one or both of the arguments being 0
   etc.

   (**One possible solution--there could be several):**

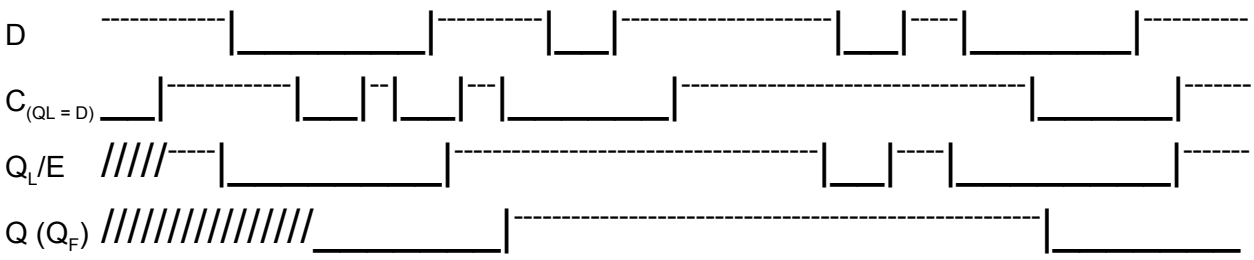
```
int lessThan(int x, int y) {
  int x_neg = (x>>31);
  int y_neg = (y>>31);
  return !((!x_neg & y_neg) | (!(x_neg ^ y_neg) & (y+~x)>>31));
}
```

Cases where <u>x < y is false</u> (which are !-ed in the return statement):
(!x_neg & y_neg): x is nonnegative while y is negative, so x > y
(!(x_neg ^ y_neg) & (y+~x)>>31):
!(x_neg ^ y_neg): same signs
(y+~x)>>31: sign bit of y+(-x-1)
(y-x-1 < 0 therefore y < x+1 or x+1 > y; moreover, because x and y are both ints, this also tests if x != y (because if x==y, the less than relationship does not hold, and y+1 > y through substitution)

2. a. Draw a **D flip-flop** with **falling-edge trigger**, complete with **all of its gates** (including the ones in the leader + follower D latches) and labels:



b. Simulate the behavior of the flip-flop you drew, given the waveforms of D and C:



What are the behaviors of Q_L/E and Q in relation to C?

Q_L/E matches the state of D when C is on/high/1, but doesn't change once C is off/low/0.

Q updates to the state of Q_L/E right after C goes from on to off (falling edge); otherwise doesn't change.

How would the D flip-flop and its waveforms look if it has a **rising-edge** trigger? What would be the behaviors of Q_L/E and Q in relation to C?

The D flip-flop would have a NOT gate between C and the Leader latch instead of the Follower latch.

(Try out the waveforms and see how Q_L/E and Q relate to C!)