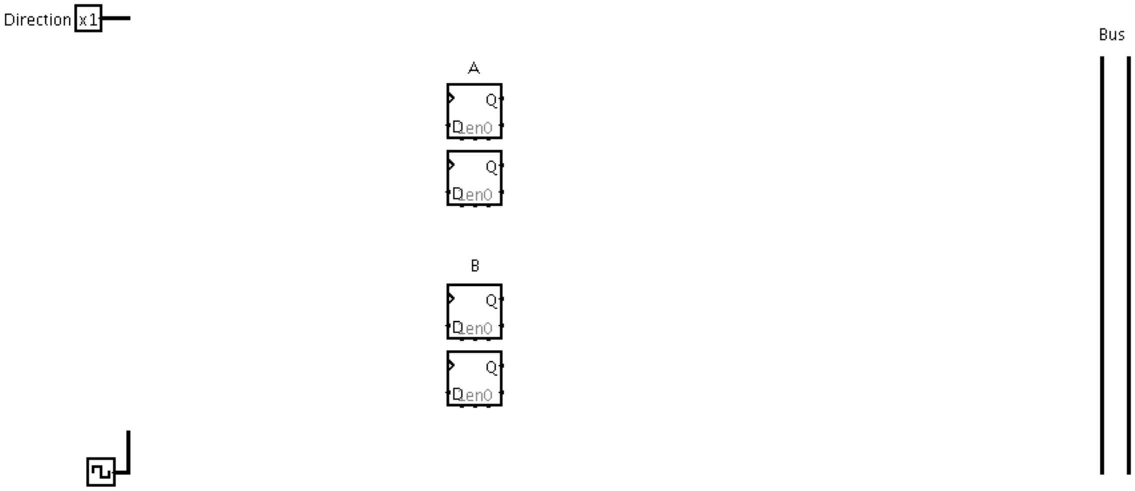
Exam #3

Name: ______________________________

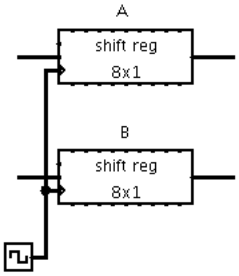
Write all of your responses on the exam paper. Show all of your work.

1. (*20 points*) In the following diagram we have 2 two-bit registers A and B constructed from D Flip-Flops and a two-bit bus to the right. There is a clock and a direction selection bit input. Using AND, OR , and NOT gates, construct the circuitry that will transfer register A to register B if the direction bit is 1 and will transfer register B to register A if the direction bit is 0.

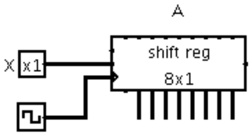


2. (*10 points*) Construct a four-bit shift register using D Flip-Flops.

3. (*20 points*) In the following diagram we have 2 eight-bit shift registers A and B. The register input is on the left and the output is on the right, input and output are both a single bit. Use one full-adder and one D flip-flop to construct a serial adder so that after 8 clock cycles register A contains the sum of registers A and B and register B contains its original value.



4. (*10 points*) In the following diagram we have a single eight-bit shift register A. X is the input stream and the 8 lines coming out of the bottom carry the value of each of the bits in the register. Using AND, OR, and NOT gates, construct a circuit that will recognize the sequence of bits 01100111.



5. (*20 points*) Write a NASM program that will take an informal name (first followed by last) as command-line arguments, and return the person's formal name. Several example runs are below. If the input name is not just first and last then the program should print out an error. You may assume that you have a `functions.asm` file that contains the functions `atoi`, `iprint`, `iprintLF`, `slen`, `sprint`, `sprintLF`, and `quit`, as we did in class.

```
./FormalName Don Spickler
Spickler, Don
./FormalName Jack Frost
Frost, Jack
./FormalName James Earl Jones
Not a valid name.
```

6. (*20 points*) Write a NASM program that will ask the user for a single number n and return the value of $n!$. A couple example runs are below. Recall that $0! = 1$ and $n! = n \cdot (n-1) \cdots 2 \cdot 1$ when $n > 0$. You may assume that you have a `functions.asm` file that contains the functions `atoi`, `iprint`, `iprintLF`, `slen`, `sprint`, `sprintLF`, and `quit`, as we did in class.

```
./prog
Please enter number: 5
5! = 120
./prog
Please enter number: 7
7! = 5040
```

7. (*10 points*) Given the following C++ driver program, create the assembly function named `rem` that will take in two input parameters a and b and return the remainder of a/b. As in class, you are using 64-bit assembly in this exercise.

```
#include <iostream>

long rem(long, long) __asm__("rem");

using namespace std;

int main()
{
    long a, b;

    cout << "Input A: ";
    cin >> a;

    cout << "Input B: ";
    cin >> b;

    cout << "Remainder of A/B = " << rem(a, b) << endl;

    return 0;
}
```