Midterm Exam

(February 14th @ 7:30 pm)

Presentation and clarity are very important! Show your procedure!

Problem 1 (20 pts)

- Compute the result of the following operations. The operands are signed fixed-point numbers. The result must be a signed fixed point number. For the division, use $x = 5$ fractional bits.

1.010001 + 1.011	1001.1101 - 1.011101	0.010101 + 01.11111
10.101 × 0.10011	1.011 × 1.0101	10.10010 ÷ 0.101

Problem 2 (10 pts)

- Represent these numbers in Fixed Point Arithmetic (signed numbers). Select the minimum number of bits in each case.
  - ✓ -16.375
  - ✓ 32.3125

- Complete the table for the following fixed point formats (signed numbers): (6 pts.)

Integer bits	Fractional Bits	FX Format	Range	Resolution
6	3			
8	5			

Problem 3 (40 pts)

- Calculate the result (provide the 32-bit result) of the following operations with 32-bit floating point numbers. Truncate the results when required. When doing fixed-point division, use 4 fractional bits. Show your procedure.

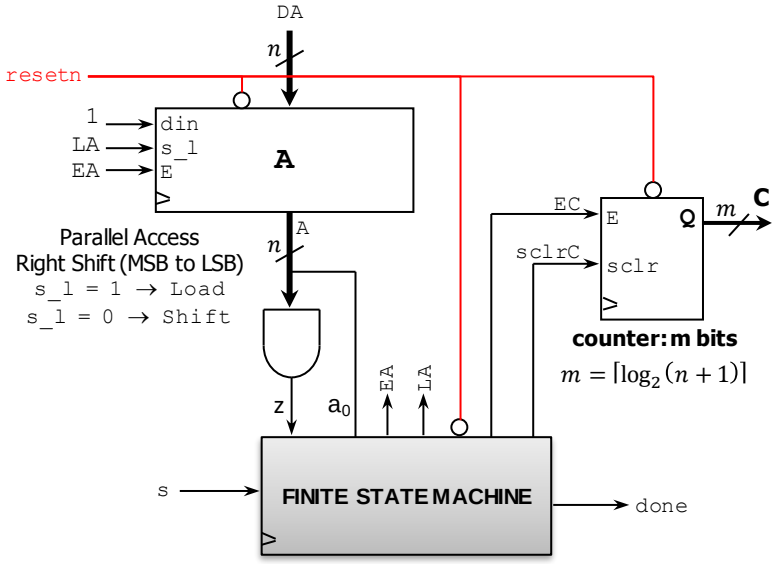
✓ C1500000 + 436A0000	✓ D0A90000 - CF480000	✓ 80400000 × 7AB80000	✓ FBB80000 ÷ 49400000

Problem 4 (30 pts)

- "Counting 0's" Circuit: It counts the number of bits in register A that has the value of '0'.
  The digital system is depicted below: FSM + Datapath. Example: For $n = 8$: if $A = 00110110$, then $C = 0100$.
  - ✓ m-bit counter: $sclr$. If $E = sclr = 1$, the count is initialized to zero. If $E = 1, sclr = 0$, the count is increased by 1.
  - ✓ Parallel access shift register: If $E = 1$: $s\_l = 1 \rightarrow$ Load, $s\_l = 0 \rightarrow$ Shift.

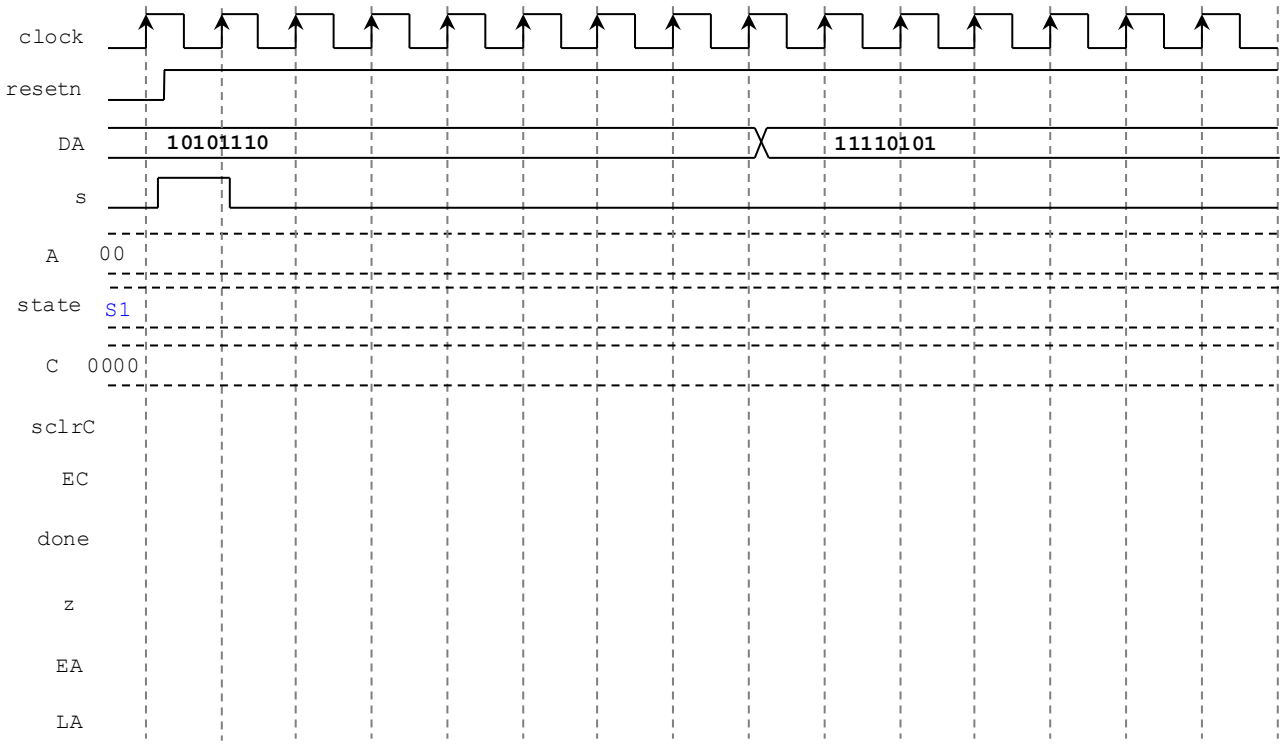
- Sketch the Finite State Machine diagram (in ASM form) given the algorithm (for $n = 8$, $m = 4$). (18 pts.)
  - ✓ The process begins when s is asserted, at this moment we capture DA on register A. Then the process starts by shifting A one bit at a time. The process is concluded when $A = 2^n - 1$. The signal $done$ is asserted when we finish counting.
  - ✓ Note: If $A = 2^n - 1 \rightarrow z = 1$, else $z = 0$. As A is being shifted, each time $a_0 = 0$, we need to increase the count C.

- Complete the timing diagram (next page) where $n = 8, m = 4$. (12 pts.)



ALGORITHM

```
C ← 0
while A ≠ 11...1 (2ⁿ-1)
   if a0 = 0 then
      C ← C + 1
   end if
   right shift A
end while
```


clock
resetn
DA
10101110
11110101
s
A
00
state
S1
C
0000
sclrC
EC
done
z
EA
LA