// File: Lab 3.pdf
// Author: Joseph Shumway
// Date: 10/4/2021
// Section: 504
// E-mail: joseph.shumway@tamu.edu
// Description:
// The contents of this file contains answers to the problems in part 2

1.How many DFFs are needed to make a 64-bit register?

64 DFFS are needed

2. A bit chip has two inputs: in and load, and one output: out.
Complete the following pseudo code for implementing bit logic in terms of the clock signal (t)

if (load(t-1) == 0)

out(t) = __out(t-1)__

else

out(t) = __in(t-1)__

3. What are the 7th and 12th values in the Fibonacci sequence with 0 and 1 as first and second values?

8 and 89

4. Design a RAM4 chip circuit from 16-bit Register(s). Then, use the RAM4 chips in a modular way to design the RAM8 chip.

The chip I/O for RAM4 are: in[16], load, address[2] and out[16]
The chip I/O for RAM8 are: in[16], load, address[2] and out[16].

```
CHIP RAM4 {
   IN in[16], load, address[2];
   OUT out[16];

   PARTS

   //select register
   DMux4Way(in=load, sel=address, a=a, b=b, c=c, d=d);

   // apply load to the correct register
   Register(in=in, load=a, out=outA);
   Register(in=in, load=b, out=outB);
   Register(in=in, load=c, out=outC);
   Register(in=in, load=d, out=outD);

   // output loaded value
   Mux4Way16(a=outA, b=outB, c=outC, d=outD, sel=address, out=out);
```

}

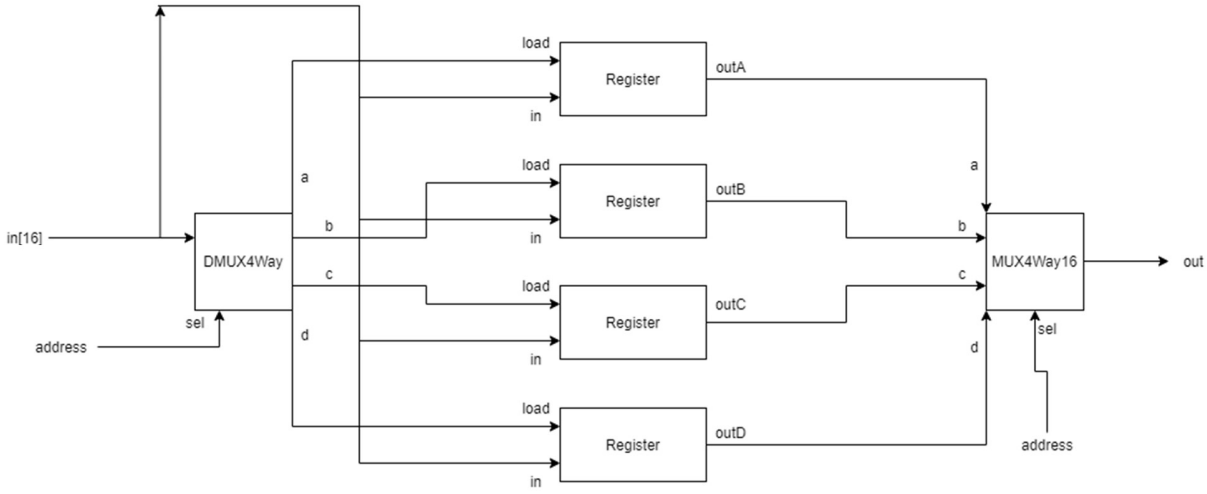
Use other combinational chips with their I/O names, if necessary.

A. How many address bits are required for RAM4?
2

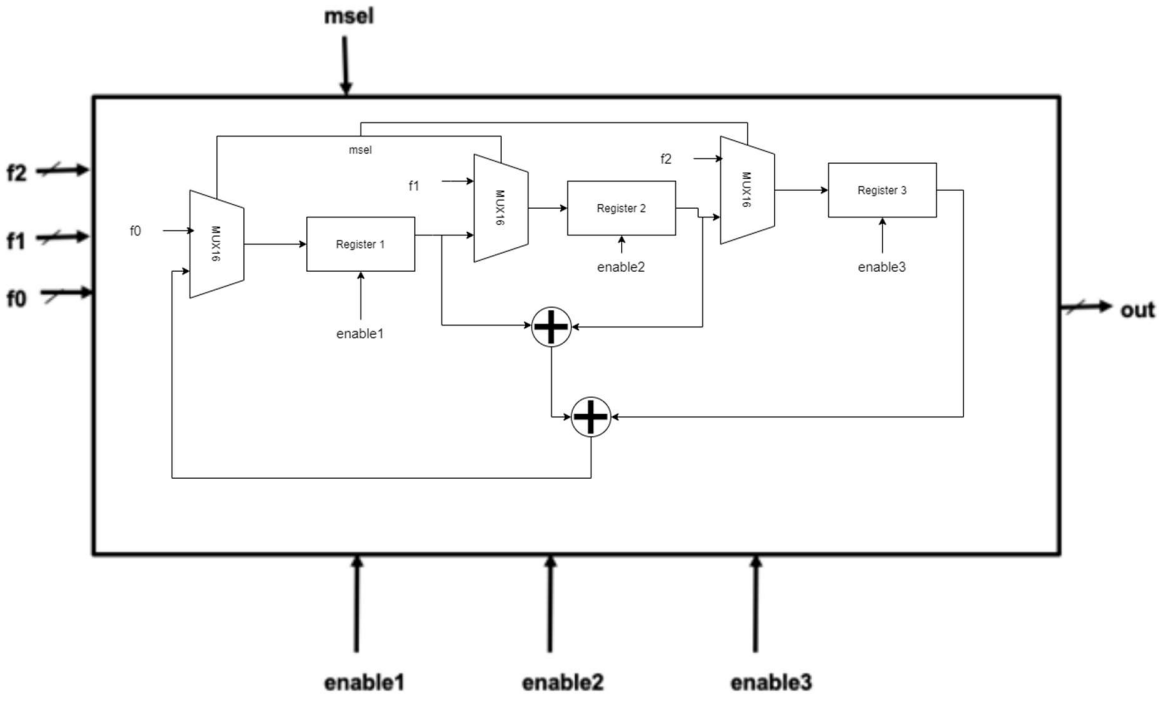
B. How many address bits are required for RAM8?
3

C. Draw the logic circuit diagram of the RAM4 using registers.

D. Draw the logic circuit diagram of the RAM8 using RAM4.



5. Draw the logic circuit diagram (with I/O pins of all components) to implement a new sequence similar to Fibonacci. In this new sequence, the first three terms f0, f1, f2, are provided, and the out is computed by adding the previous three terms.
Report the required number of each of the following chips:
#MUX16 = __3__, #Register= __3__ , #ADD16= __2__



6. Draw the logic circuit diagram of a circuit that outputs (out), the user input (in), with a delay of three clock cycles when the load was high at the time of input. Otherwise, the circuit should increment by one the output from 3 cycles before.

Hint: Use the following logic to implement your circuit:

if (load(t) == 1)

out(t+3) = in(t)

else

out(t+3) = out(t) + 1