

Joseph Quismorio
UIN 128008917

1. A 64-bit register requires 64 flip-flops.
2. if (load(t-1) == 0)
 out(t) = out(t+1)
else
 out(t) = out(t)
3. The seventh number in the Fibonacci sequence is 8, while the twelfth number is 89.
4. CHIP RAM4 {
 IN in[16], load, address[2];
 OUT out[16];

PARTS:

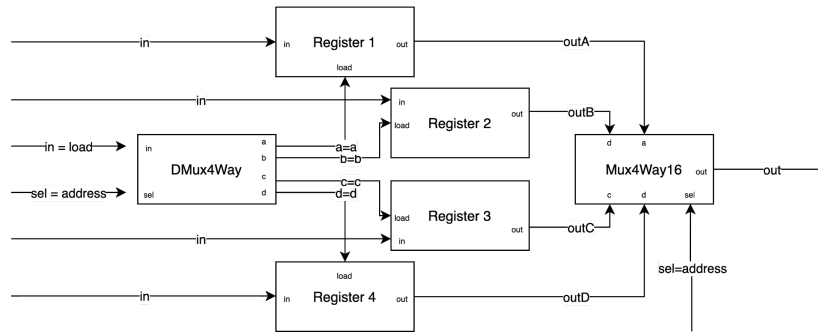
```
DMux4Way (in=load, sel=address, a=a, b=b, c=c, d=d);  
Register (in=in, load=a, out=outA);  
Register (in=in, load=b, out=outB);  
Register (in=in, load=c, out=outC);  
Register (in=in, load=d, out=outD);  
Mux4Way16 (sel=address, a=outA, b=outB, c=outC, d=outD, out=out);  
}
```

```
CHIP RAM8c {  
    IN in[16], load, address[3];  
    OUT out[16];
```

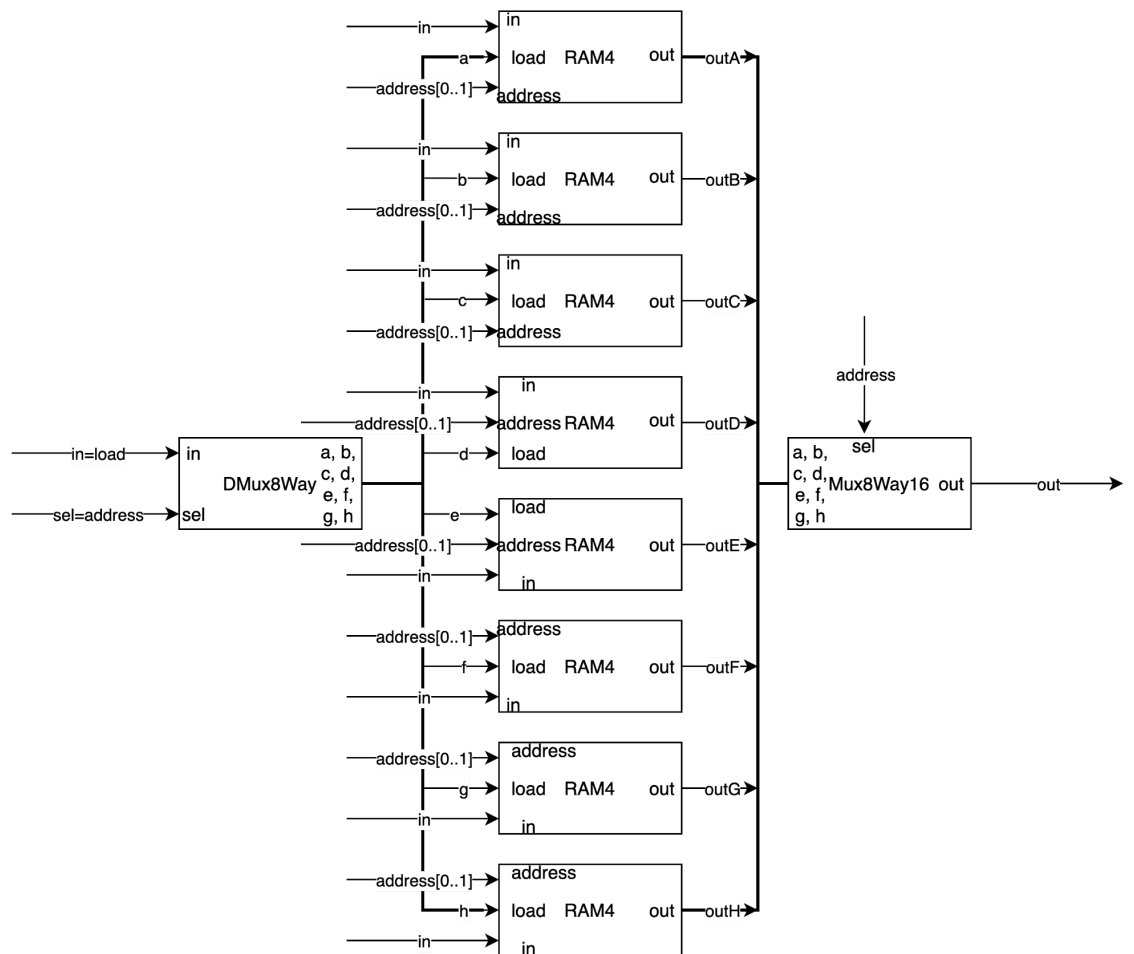
PARTS:

```
DMux8Way (in=load, sel=address, a=a, b=b, c=c, d=d, e=e, f=f, g=g,  
h=h);  
RAM4 (in=in, load=a, address=address[0..1], out=outA);  
RAM4 (in=in, load=b, address=address[0..1], out=outB);  
RAM4 (in=in, load=c, address=address[0..1], out=outC);  
RAM4 (in=in, load=d, address=address[0..1], out=outD);  
RAM4 (in=in, load=e, address=address[0..1], out=outE);  
RAM4 (in=in, load=f, address=address[0..1], out=outF);  
RAM4 (in=in, load=g, address=address[0..1], out=outG);  
RAM4 (in=in, load=h, address=address[0..1], out=outH);  
Mux8Way16 (sel=address, a=outA, b=outB, c=outC, d=outD, e=outE,  
f=outF, g=outG, h=outH, out=out);  
}
```

- 2 address bits
- 3 address bits

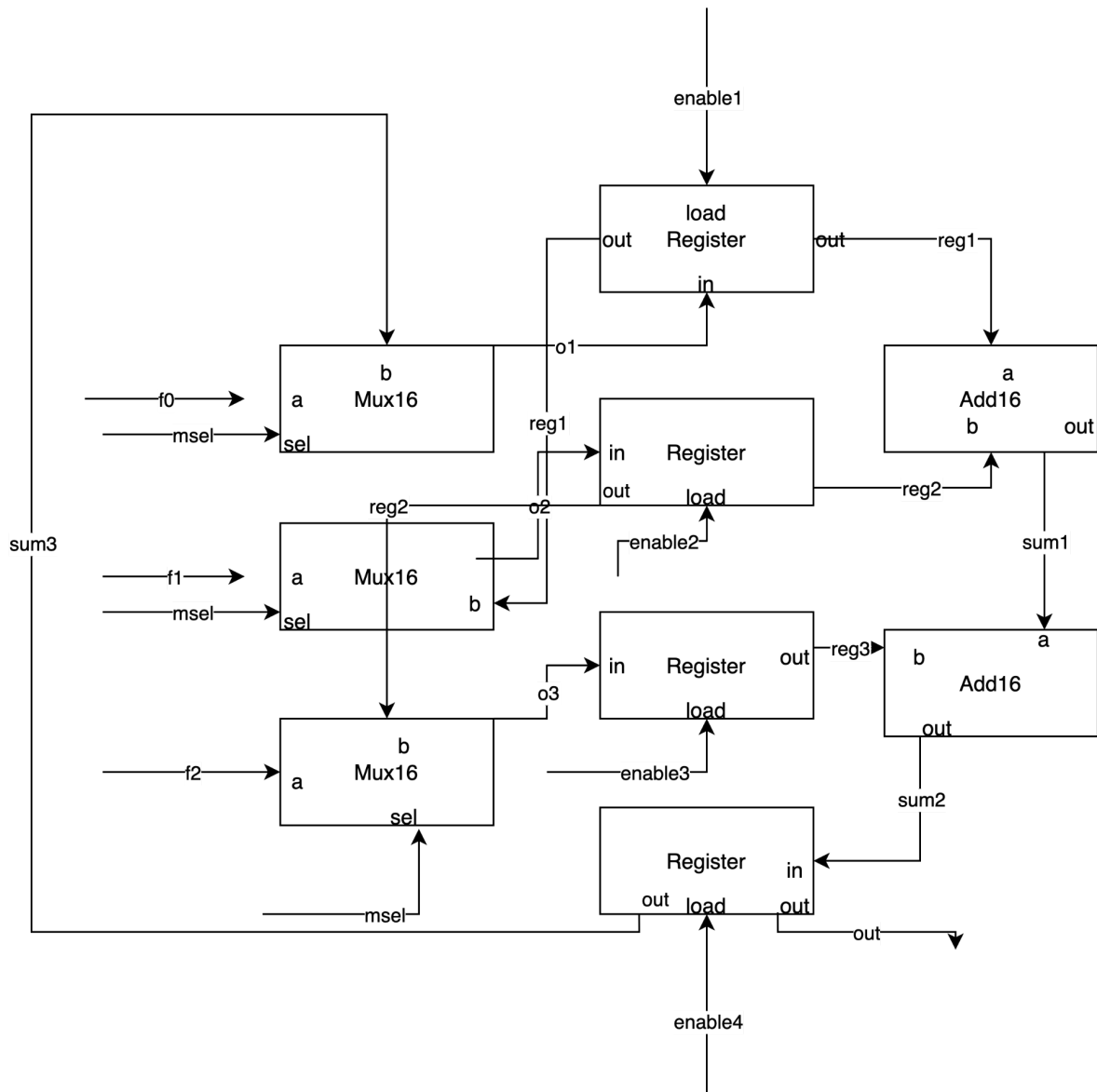


c.



d.

5.



3 Mux16s, 4 registers, 2 add16s

6.

