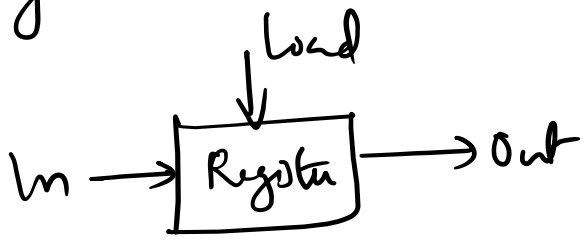
Week 3 – Sequential Logic

Here D flipflop is given.

① 1-bit Register

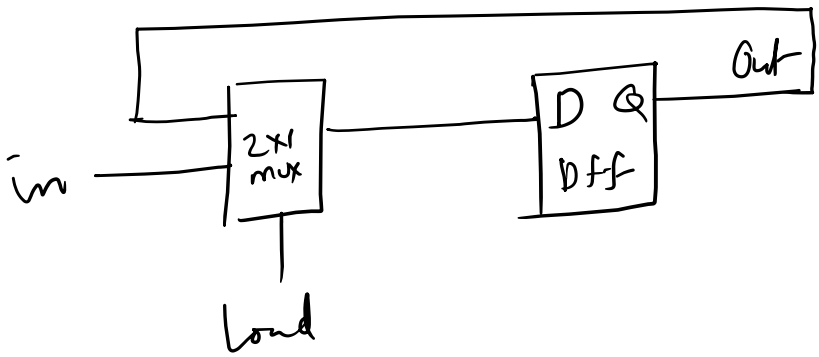


If Load[t] = 1 ⇒ Out[t+1] = in[t]

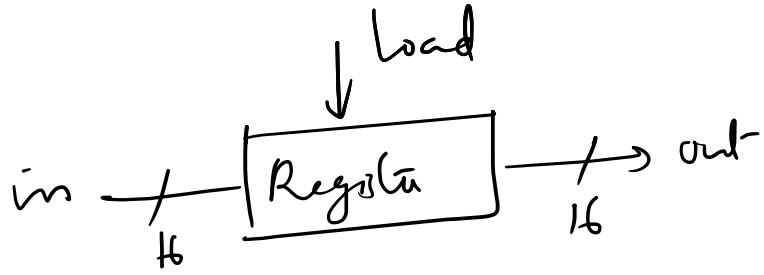
To read probe out. Out shows the value.

× Load has 2 possibilities

Load	Q_{n+1}
0	Out
1	in



② 16 bit Register

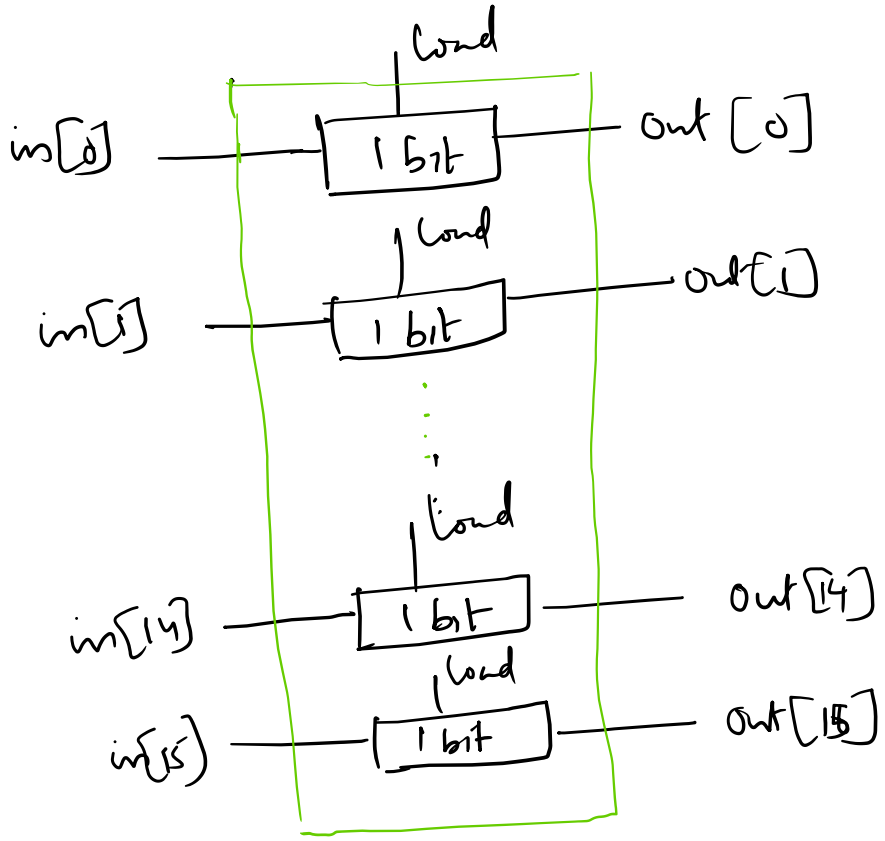


To read: probe out (out shows the value)

Write: set in = V (eg)

set load = 1

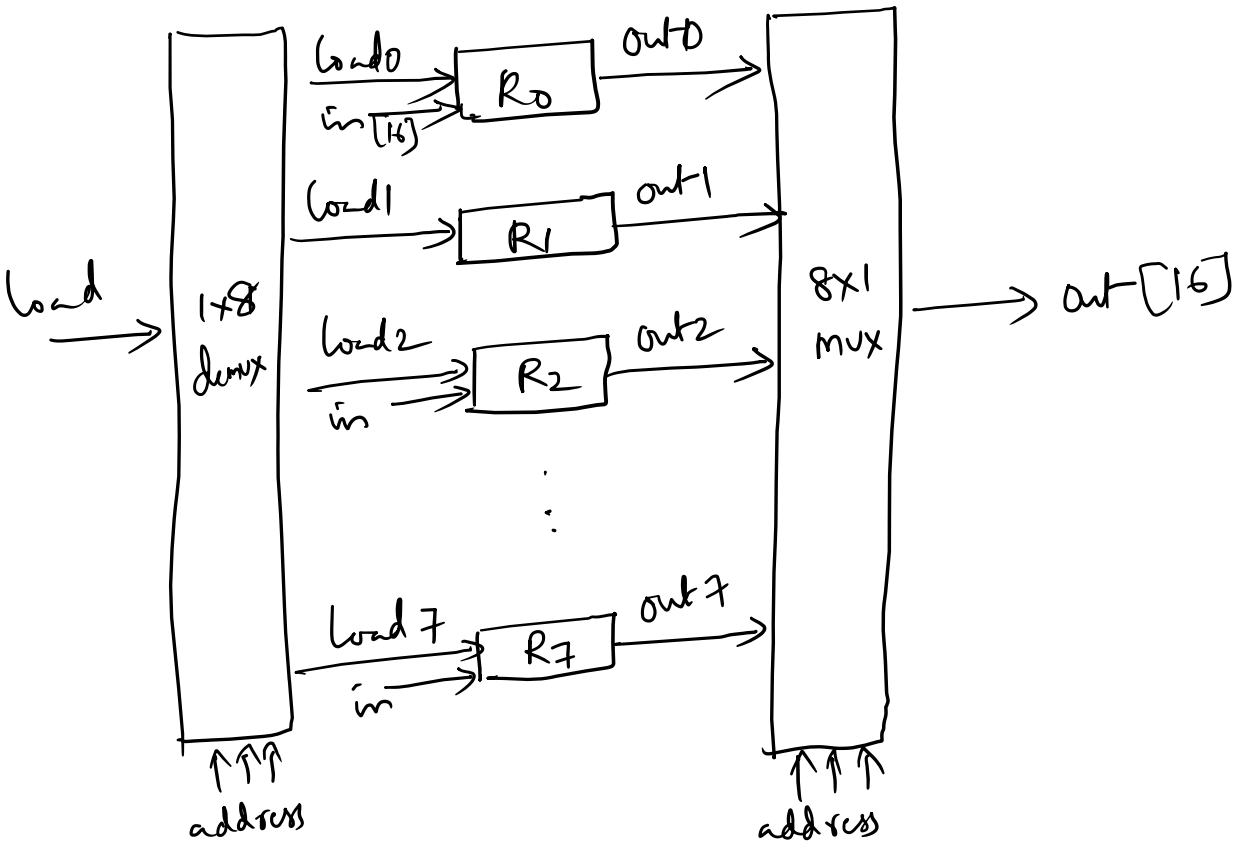
Then on next cycle output becomes V



③ RAM8 (RAM with 8 Registers)

Given: in[16], address, load.

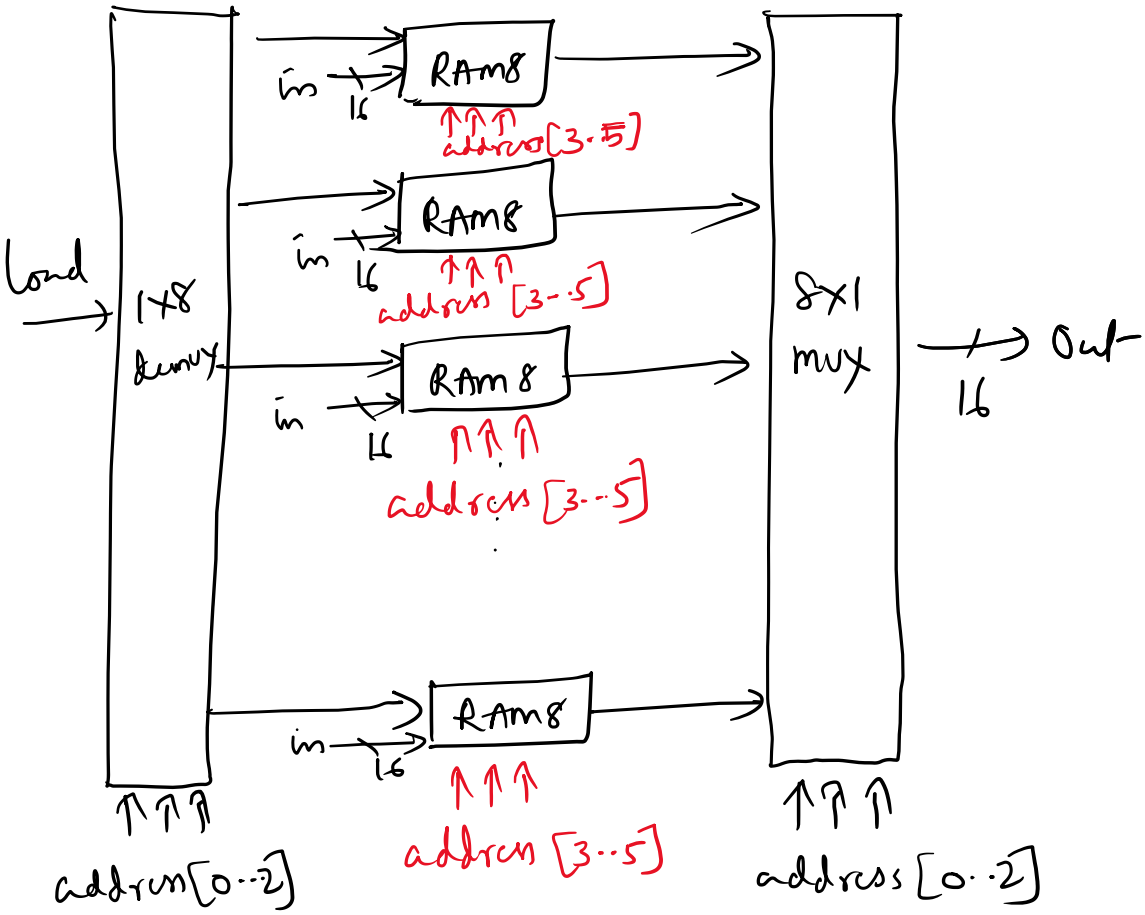
To access 8 registers we need 3 bit address. Once the register is selected, we proceed as in ②. To take output we used the same 3 bit address and select the required register.



④ RAM64

$\frac{64}{8} = 8 \Rightarrow$ 8 RAM8 can be used.

To select 8 RAM stacks we need 3 bit address. Then we use 3 bit address to access the specific register as in ③.

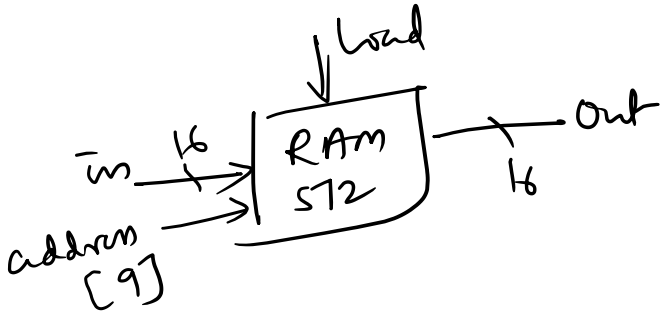


eg: address = 000 001

Select stack 1 — Select R

(5) RAM 512

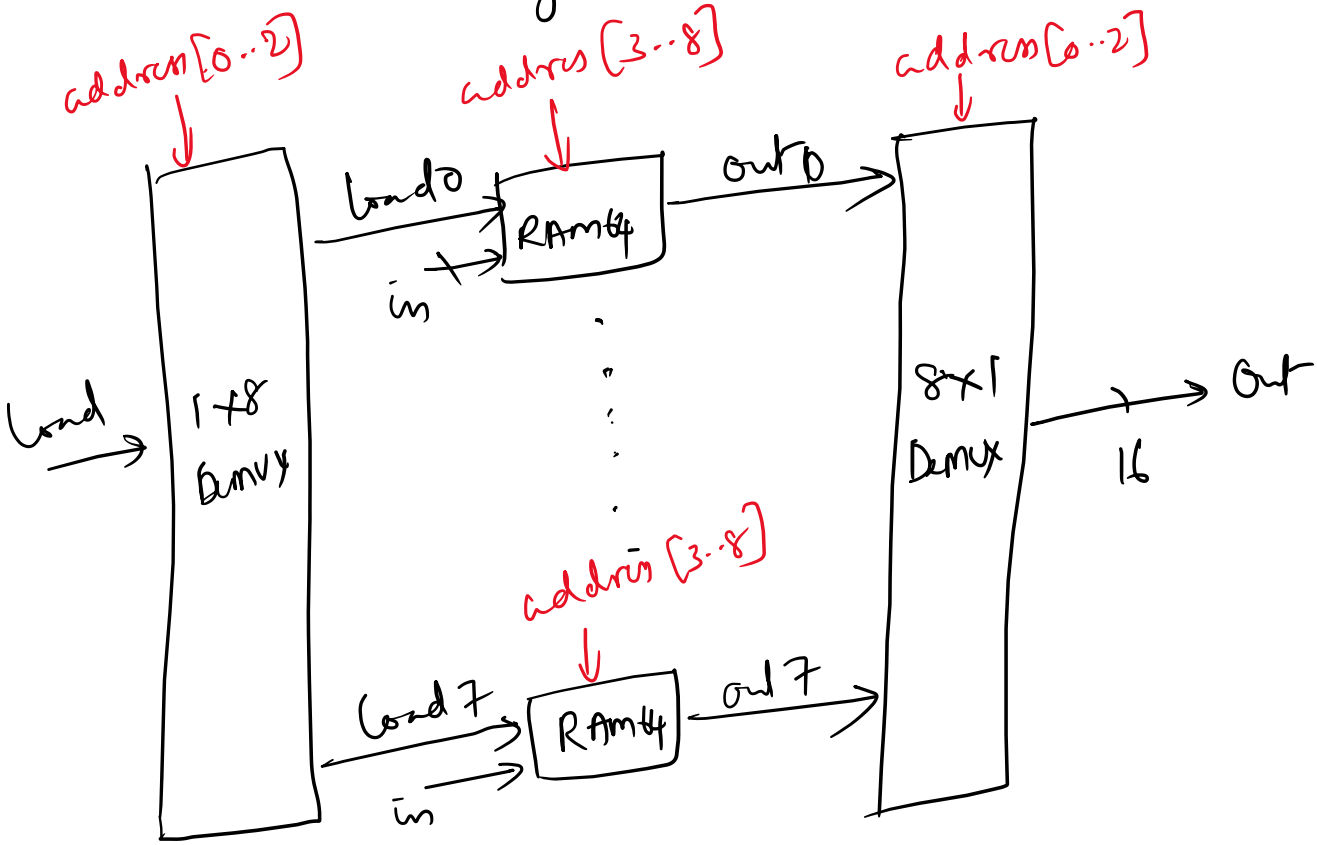
$$\frac{512}{64} = \frac{2^9}{2^6} = 8$$



8 stacks of RAM64 required.

$\Rightarrow 2^3$, i.e 3 bit address for selecting stack

$64 = 2^6$, i.e 6 bit address for selecting the Register in 64 RAM.

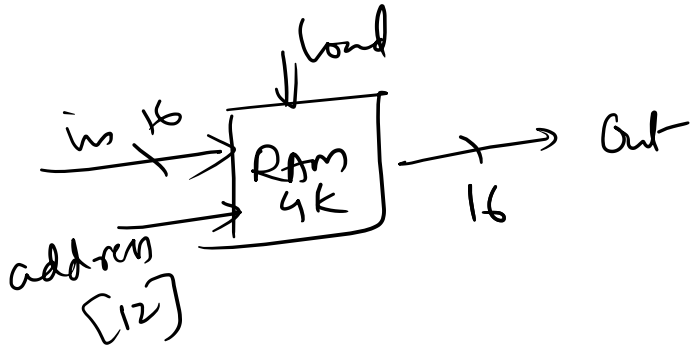


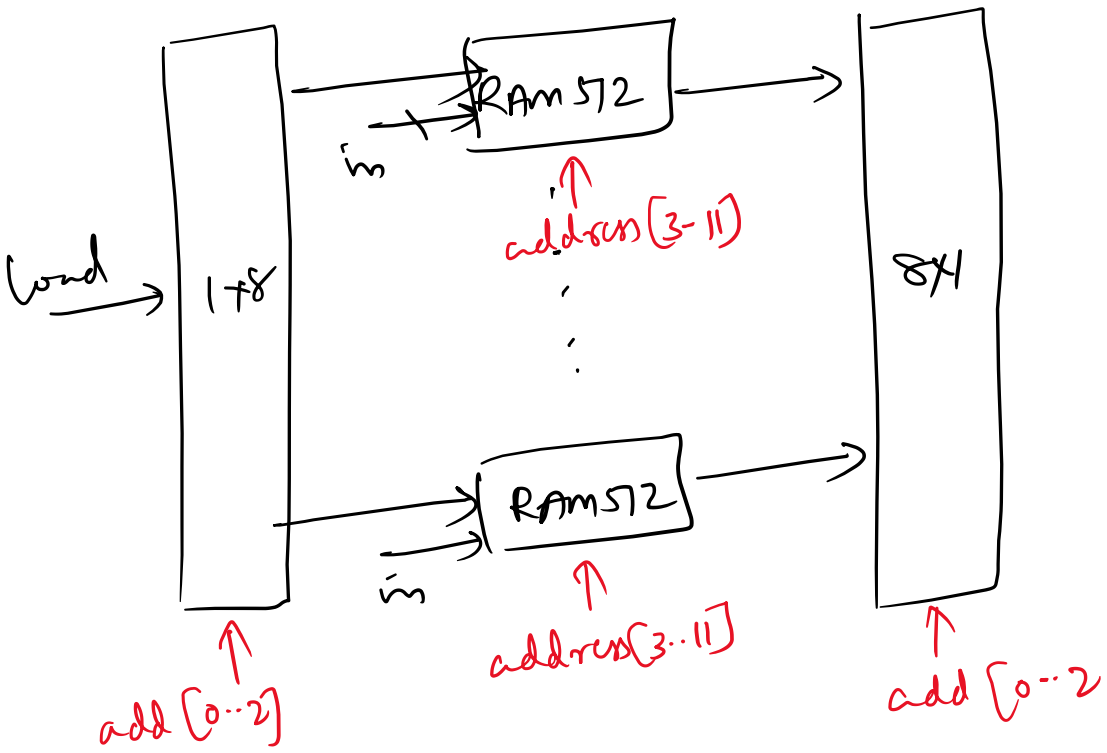
⑥ RAM 4K

$$\frac{4K}{512} = \frac{4 \times 1024}{512} = 8$$

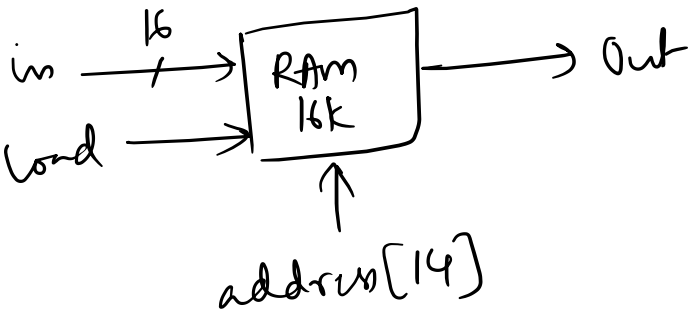
8 stacks $\Rightarrow 2^3 \Rightarrow$ 3 bit address

512 $\Rightarrow 2^9 \Rightarrow$ 9 bit address



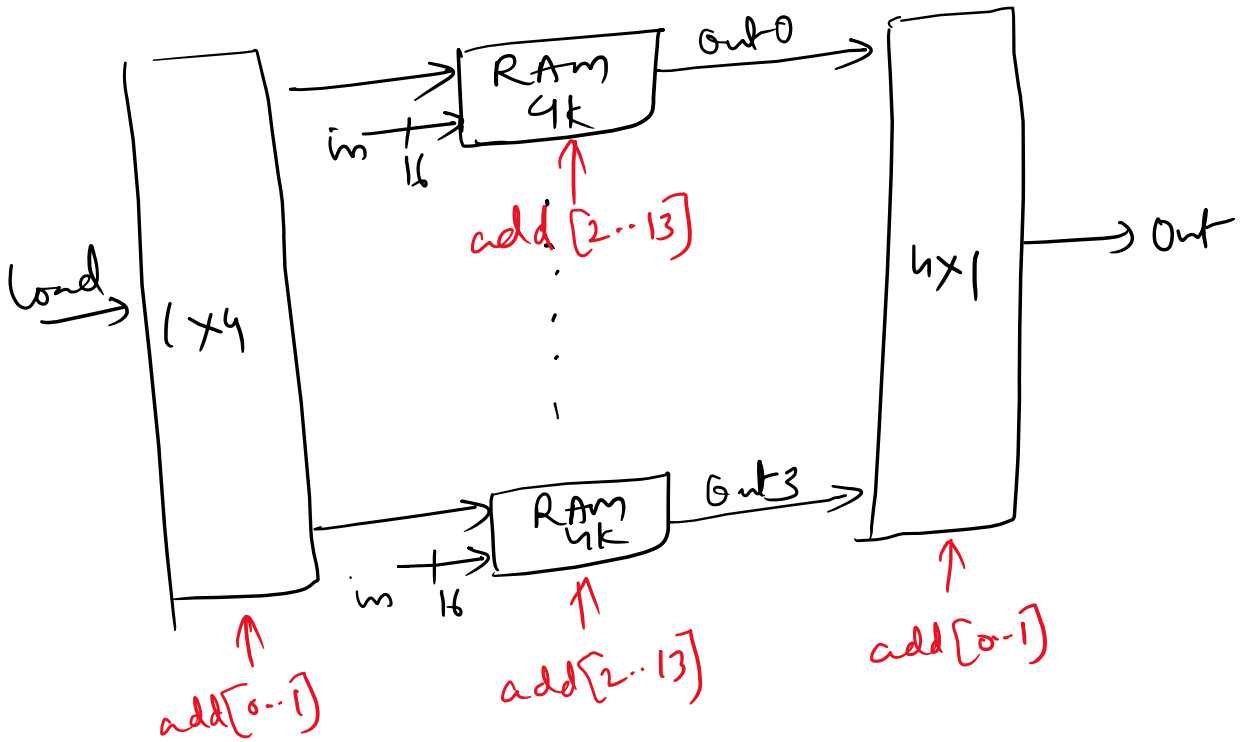


(P) RAM 16K

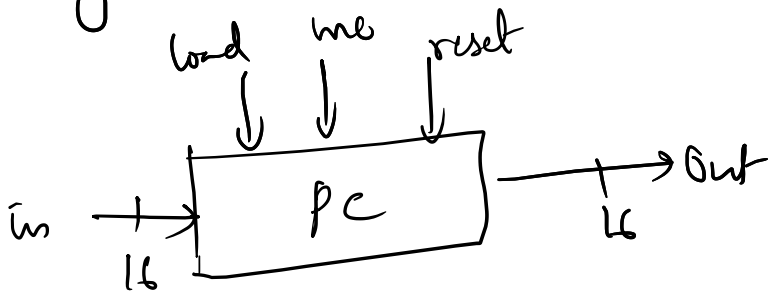


$\frac{16K}{4K} = 4$ ⇒ 4 stacks (2 bit address)

⇒ 4K (12 bit address)



⑧ Program Counter



If reset [t-1] ⇒ out [t] = 0

else if load [t-1] ⇒ out [t] = in [t-1]

else if inc (t-1) then out [t] = out [t-1] + 1

else out (t) = out (t-1)

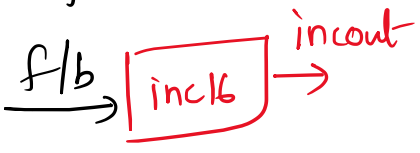
⇒ Priority order : Reset > load > increment

④ So we need a unit to do integer addition. We can used inc16 from project 2.

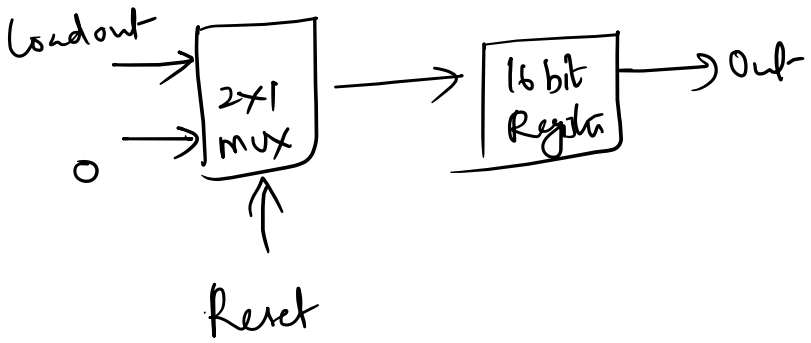
feedback

I/Ps : Out [t-1]

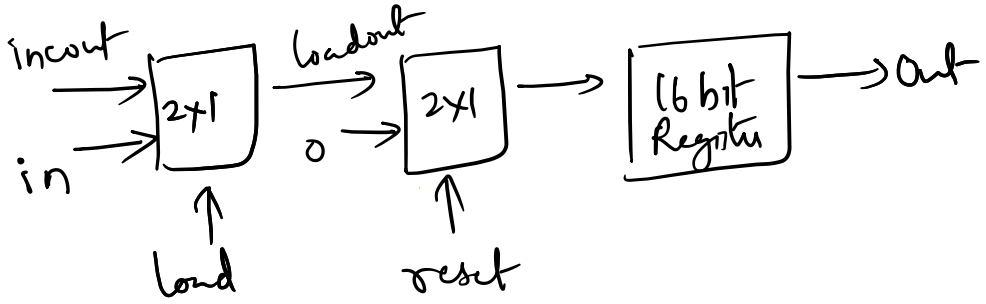
O/p : incout



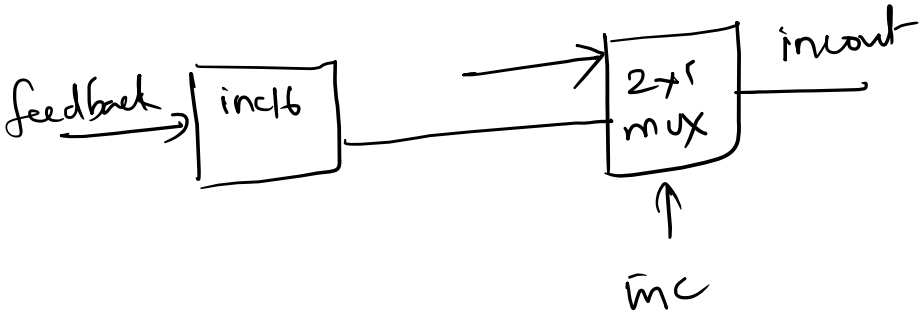
② Check reset



(3) Check load



(4) Check inc.



⑤ Connect together