Memory Module

* SRAM compilers
* Memory architecture
* Bitcell,
* Edge cells, corner cells, tap cells
* IO
* Rowdec
* Control
* Top level

Bitcell :

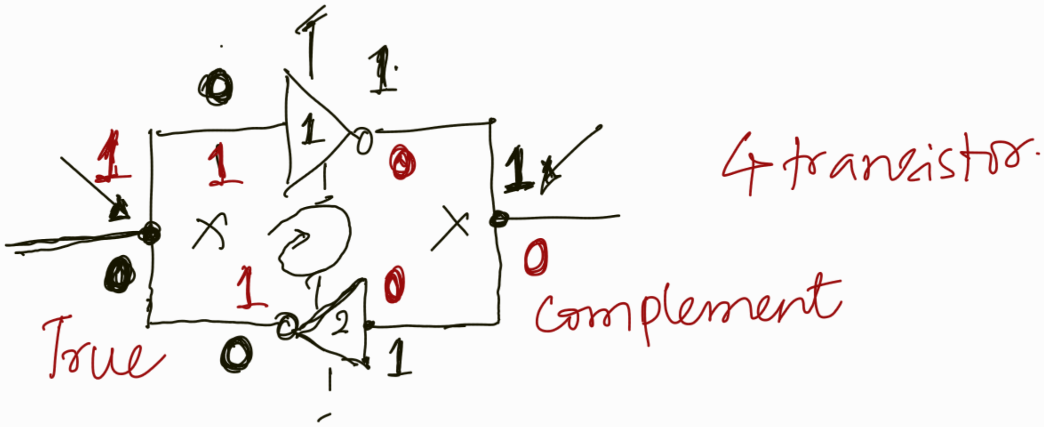
↳ store 1 bit data : 0/1

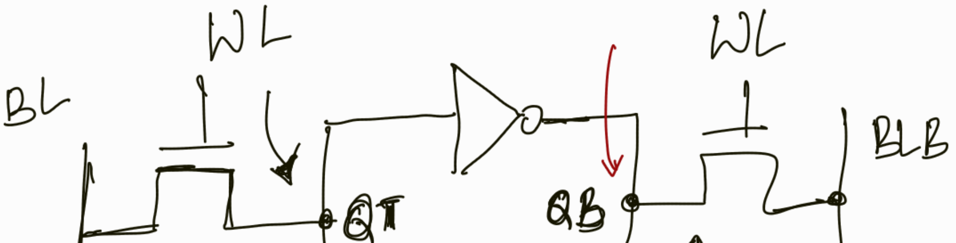
↳ SRAM → 6T bitcell. ⇒ static, volatile

↳ Write / Read

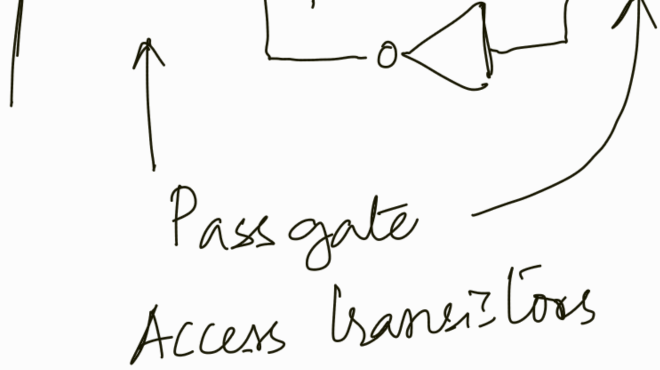
↳ Foundary /Fab, not editable.

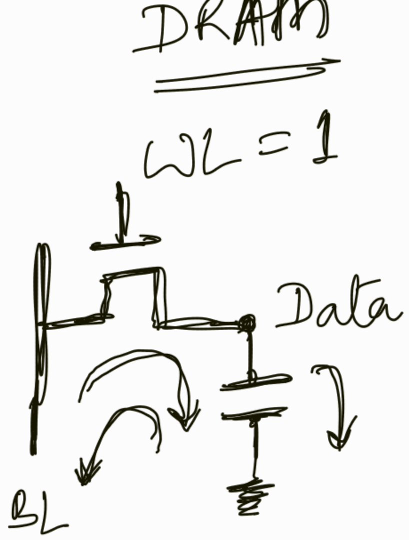
↳ Custom bitcell.

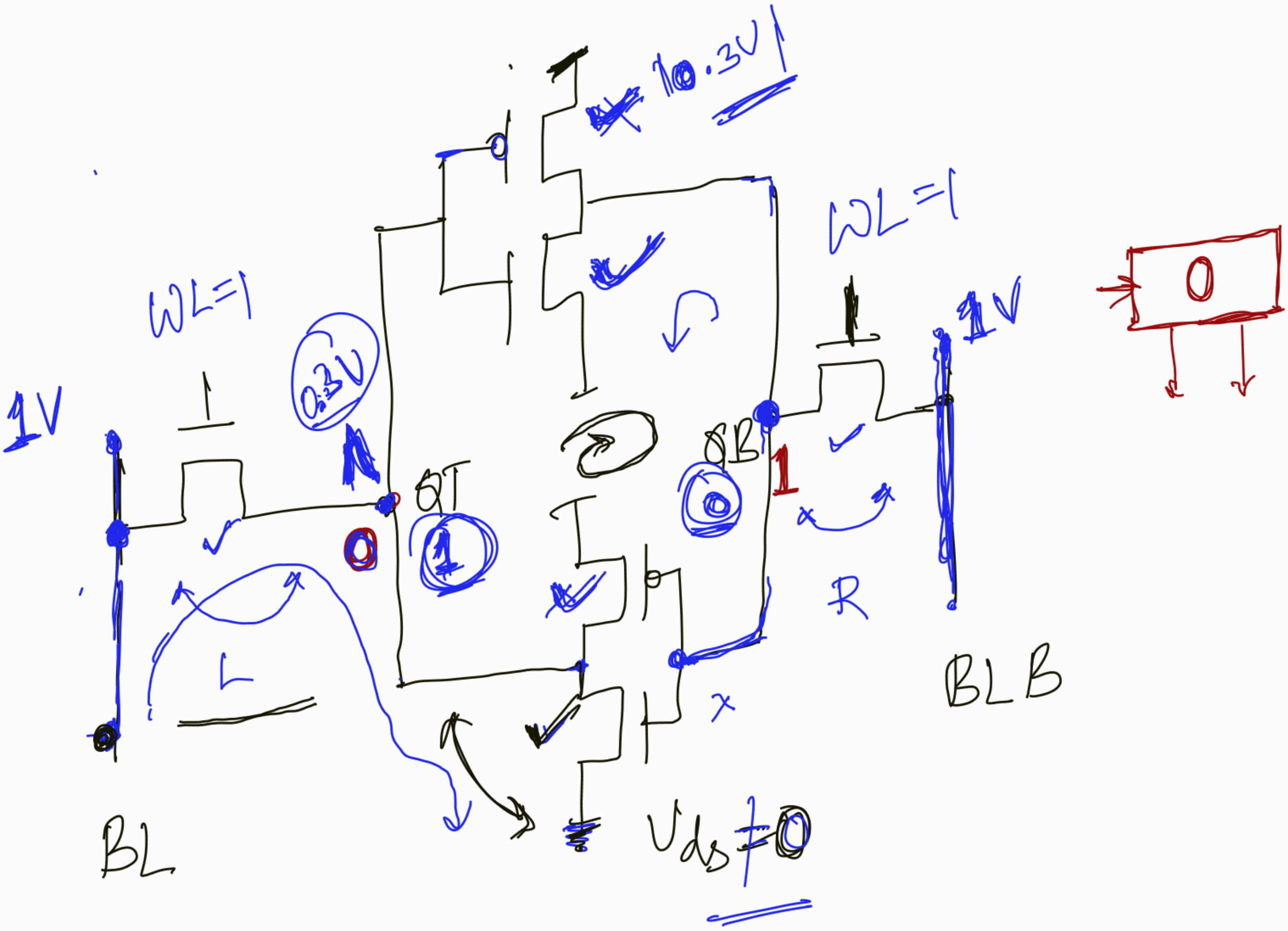




6T bitcell.







Retention / stand by

WL=0, access transistor - OFF

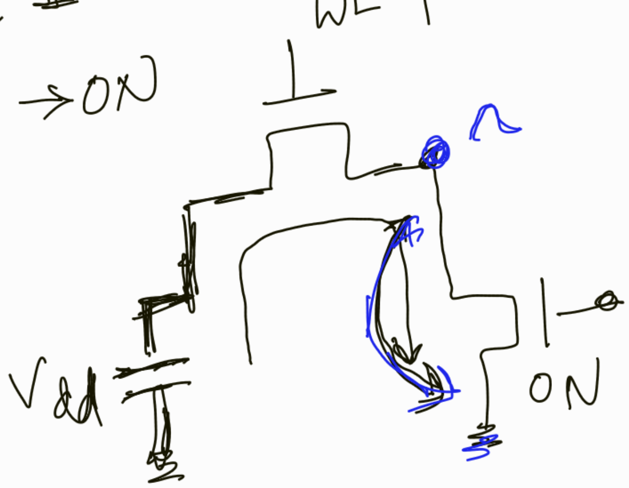
Bitcell will continue to hold the data

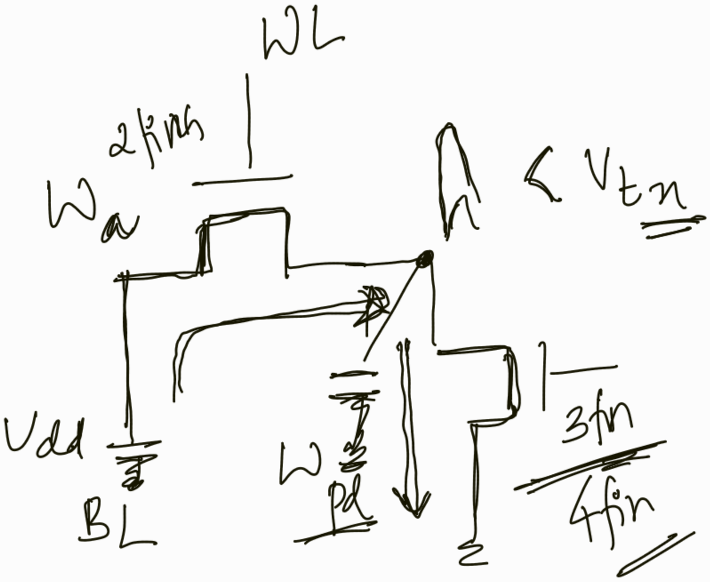
Read:

BL/BLB to Vdd ⇒ logic 1

WL = 1, access transistors → ON

$V_{dsL} = 1V$ $V_{dsR} = 0V$

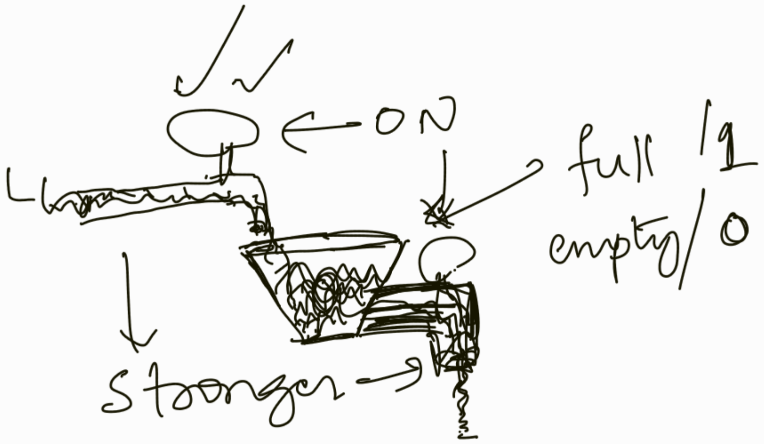




Read Access & Pull down

Pd > Access

$W_{pd} > W_{acc}$

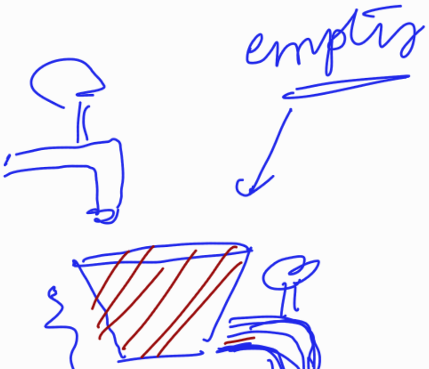


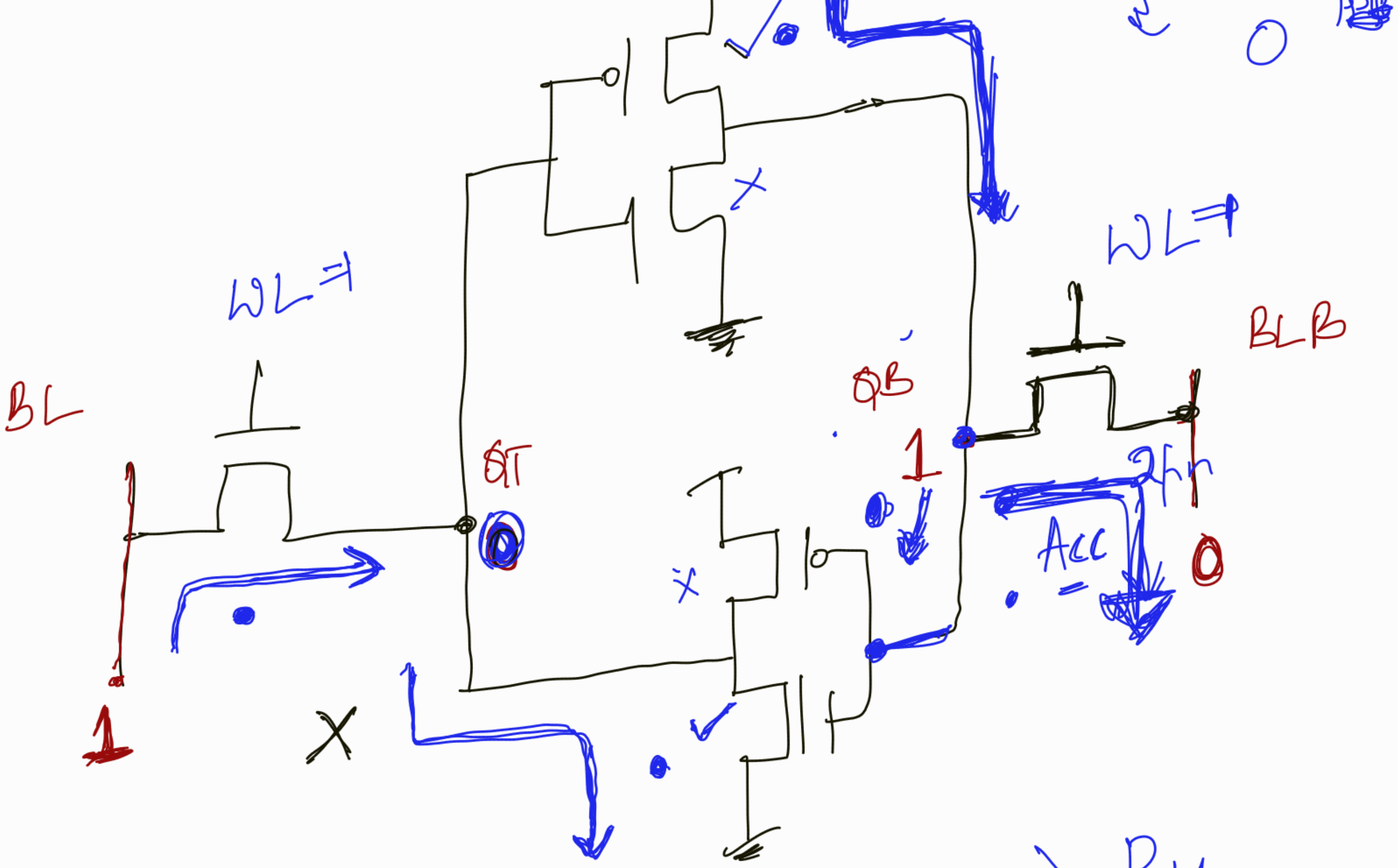
Write operation

BL = data , BLB = $\overline{data}$

WL = 1

If Pu :

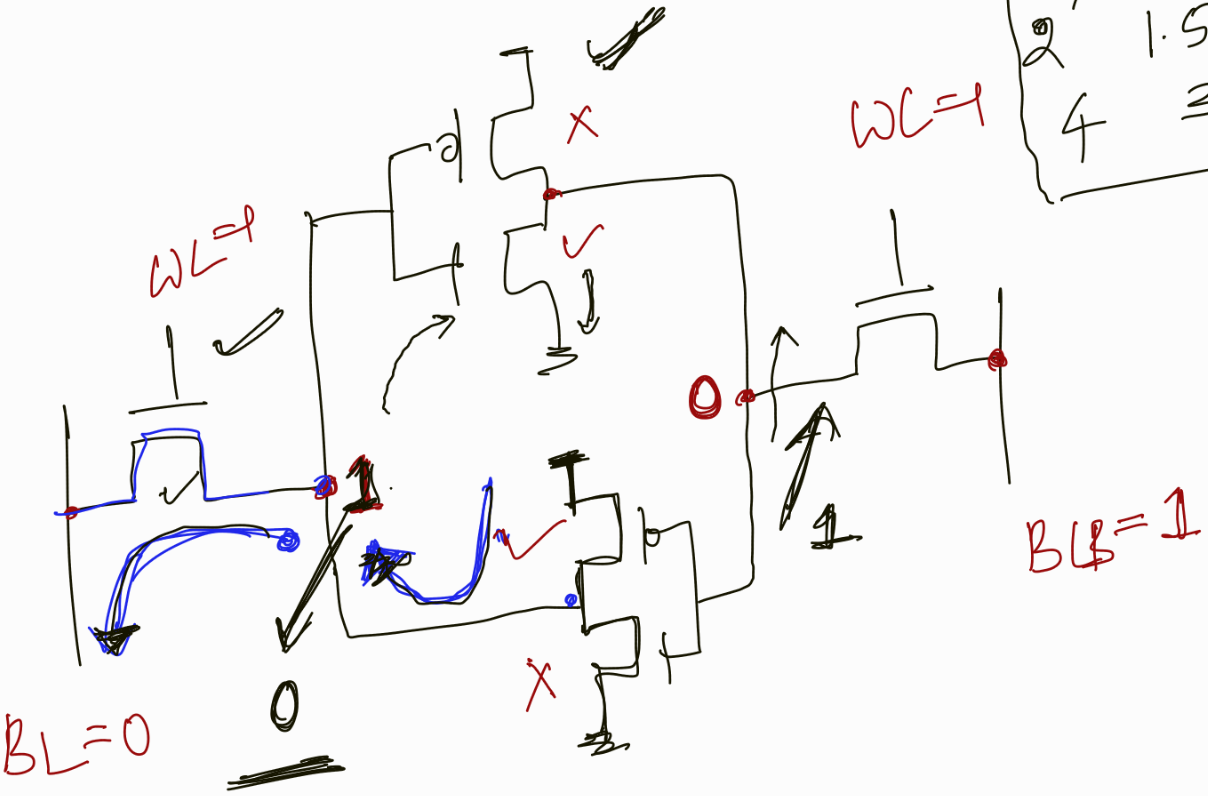




Write requirement : Acc > Pu

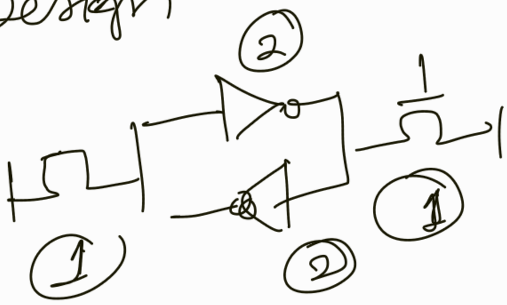
write gets initiated/triggered where the fight betn pull up & access,

Pd > Acc > Pu		
2	1.5	1
4	3	2



Memory Design

1. Bitcell



2