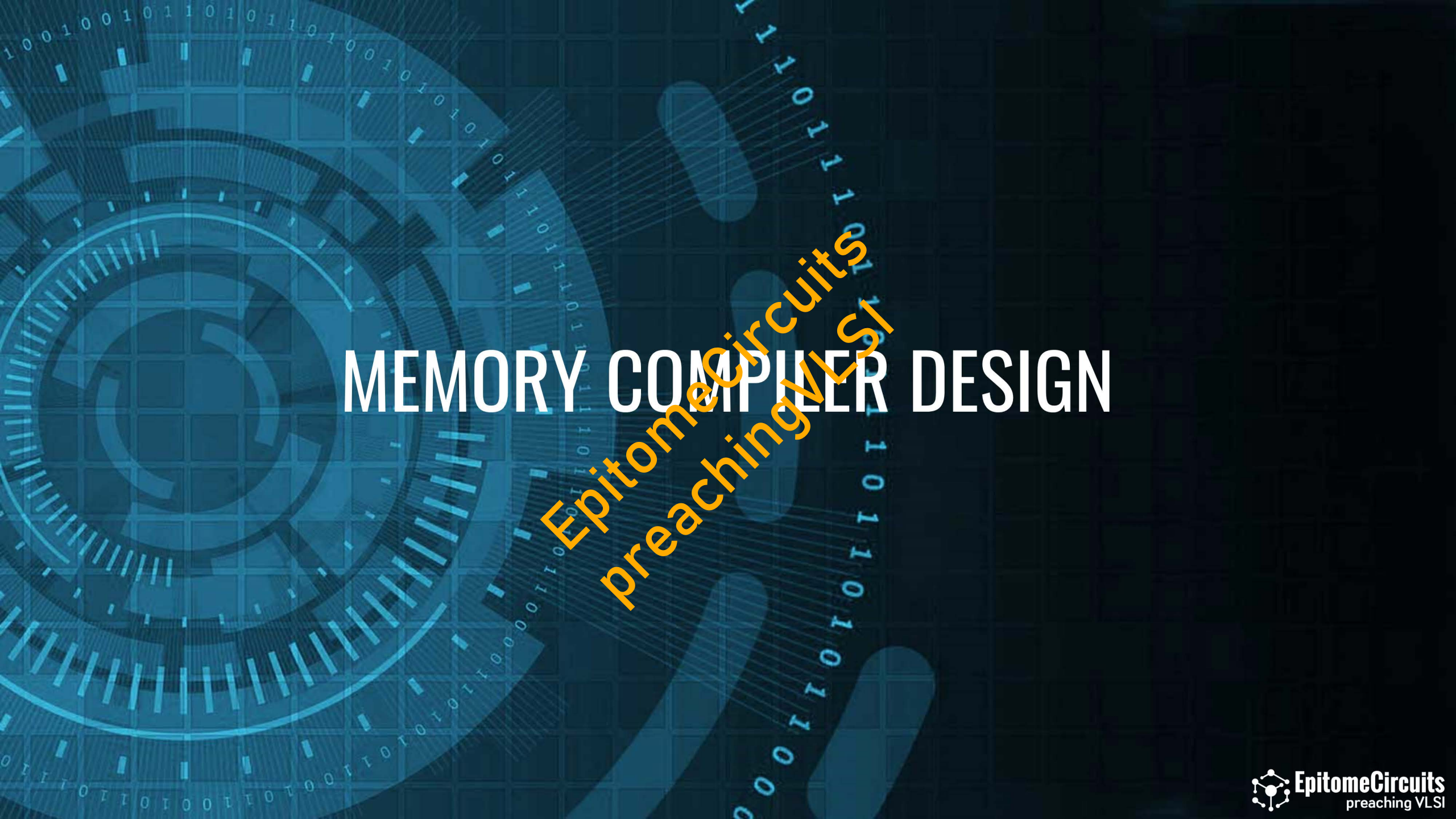
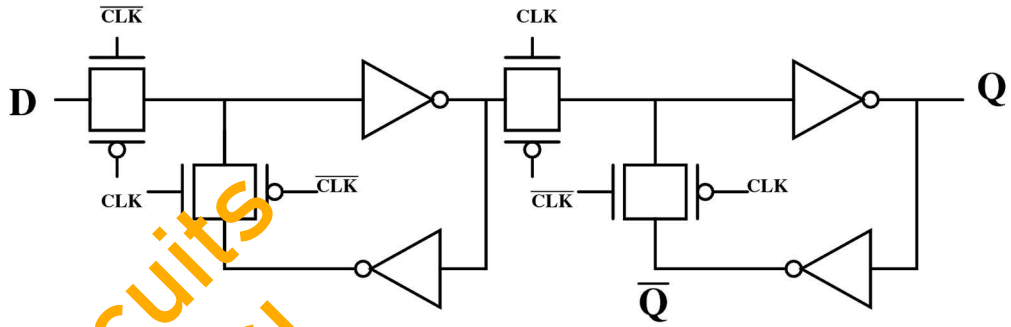


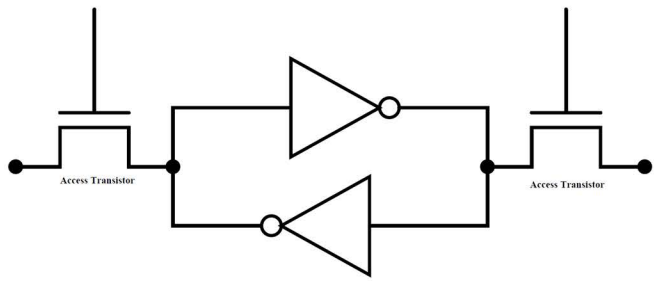
Why Bitcell? Why not D-Flip-Flop?

- D flip-flop requires a minimum of 16 transistors to store 1 bit!!!
- Also D flip flop uses clock
- As a result Area, Circuit Complexity and hence cost increases
- Therefore there is a need to go for different circuit which requires less transistors to store a bit & hence occupies less area thus decreasing the circuit complexity
- How about just using a single latch???

- Then how to access data stored???



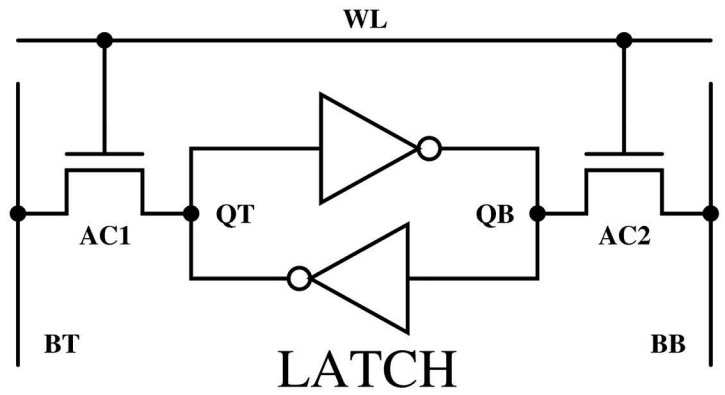
- Master Latch: 2 inverters + 2pass transistors
- Slave Latch: 2 inverters + 2pass transistors
- Inverter: 1 pMOS + 1nMOS → 2 MOS
- Pass Transistors: 1 pMOS + 1nMOS → 2 MOS
- DFF: (2x2 + 2x2)x2 = 16

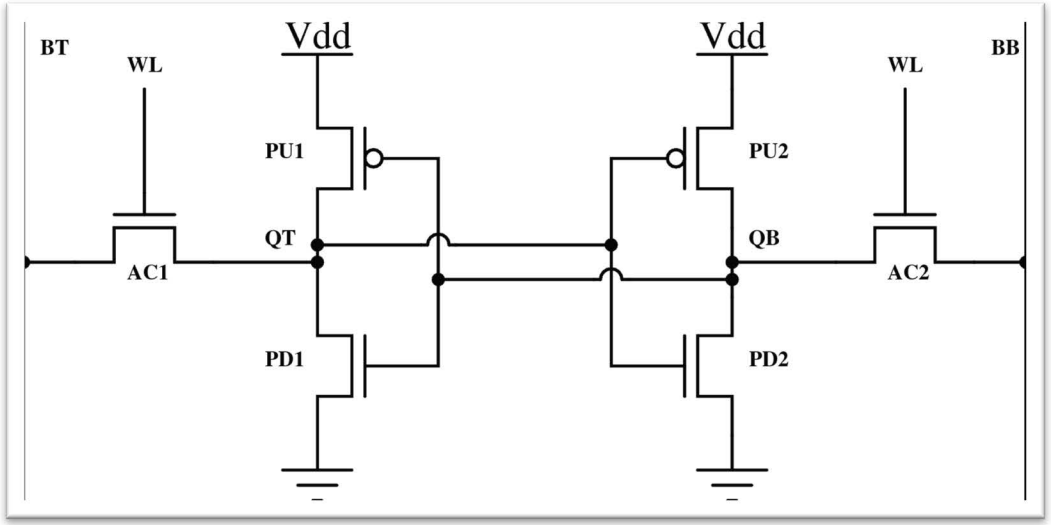


LATCH

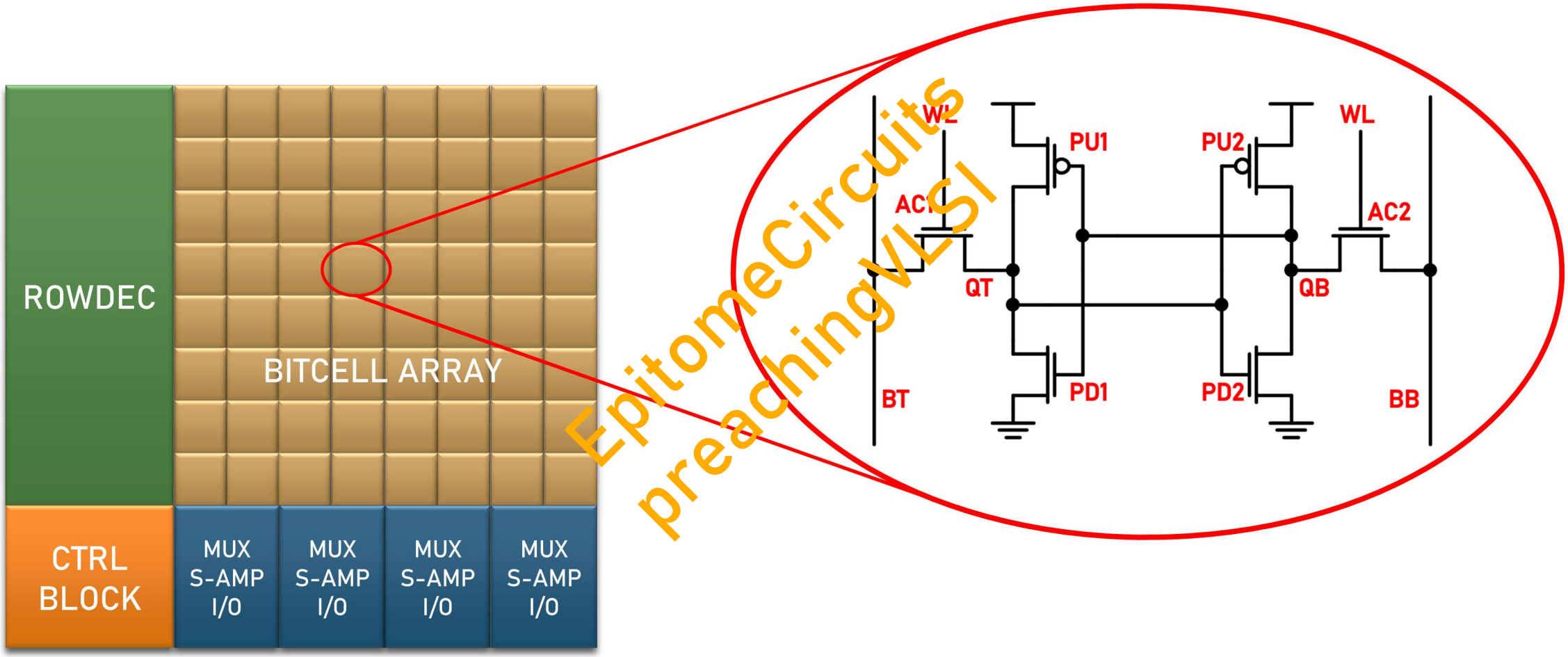
6T-SRAM

- Consists of only 6 transistors → Less Area
- Data is stored in the Latch which can be accessed using access transistors AC1 & AC2
- Highly symmetrical
- Volatile
- Assignment: Read on Latch working
- What should be the size of these 6 transistors?
  - WL : Word Line
  - BT/BB : Bit Line pair (True & Complement)
  - PU : Pull Up transistor (pMOS)
  - PD : Pull Down transistor (nMOS)
  - AC : Access transistor (nMOS)
  - QT/QB : Latch outputs

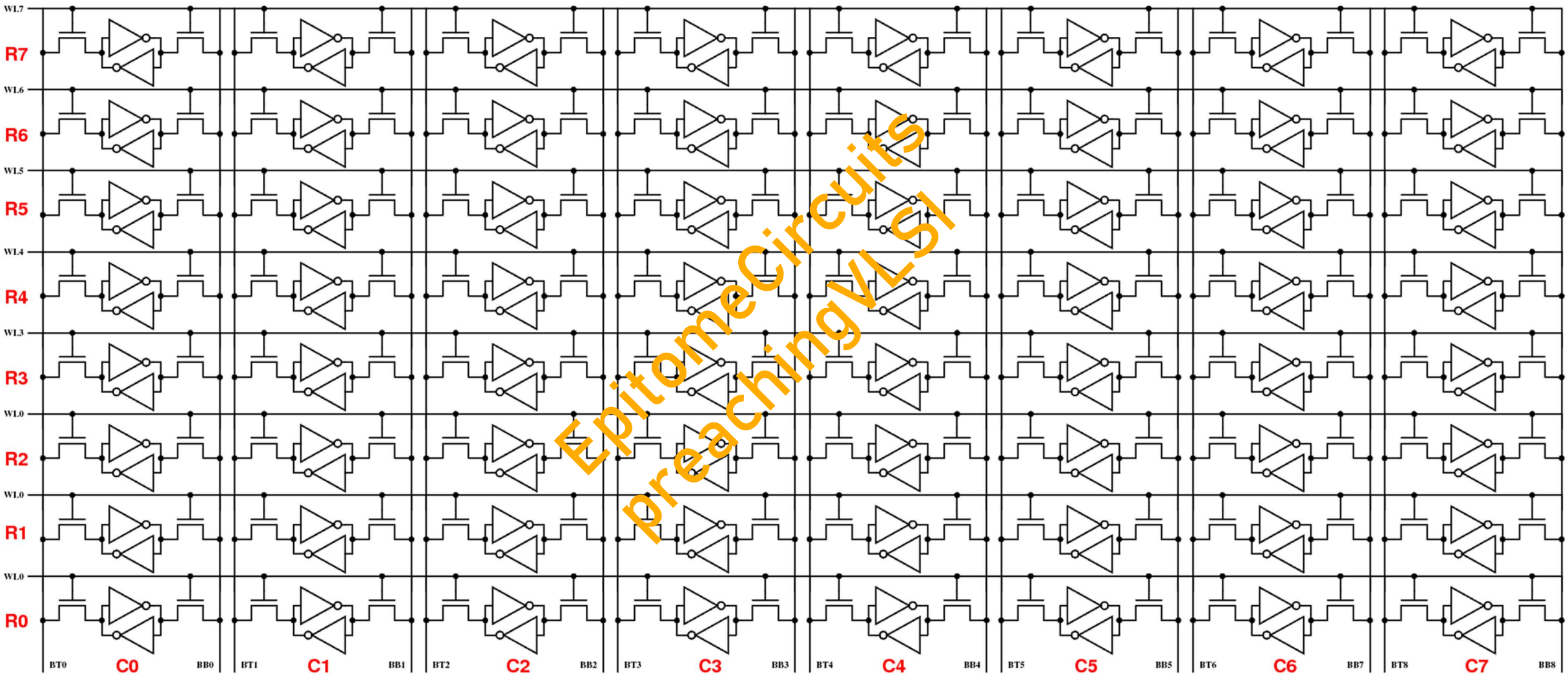




BITCELL

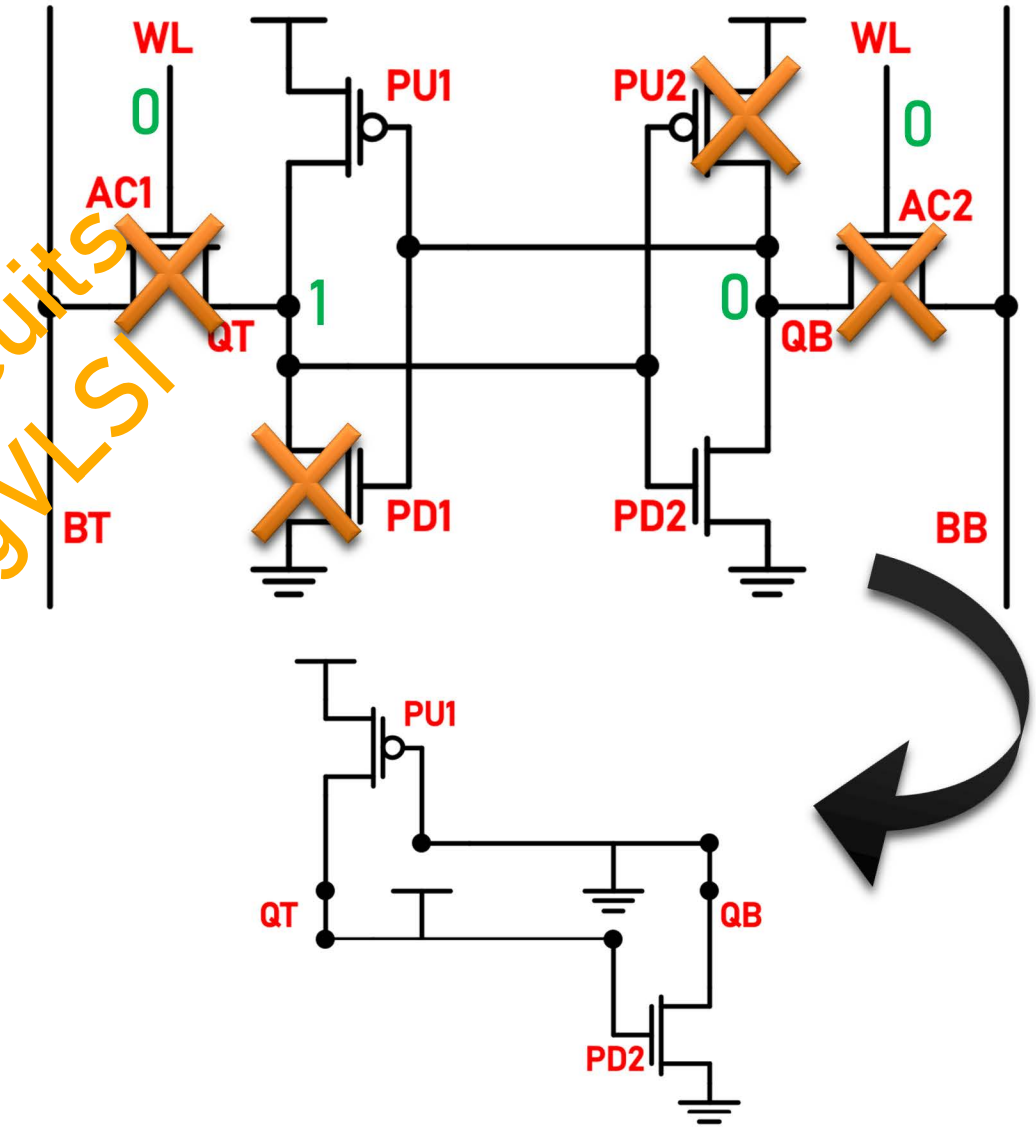


BITCELL ARRAY



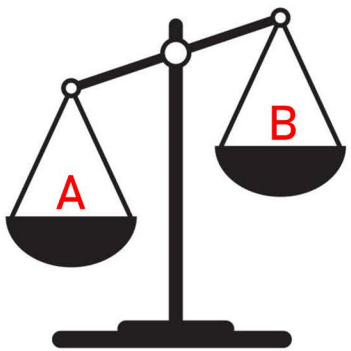
STANDBY

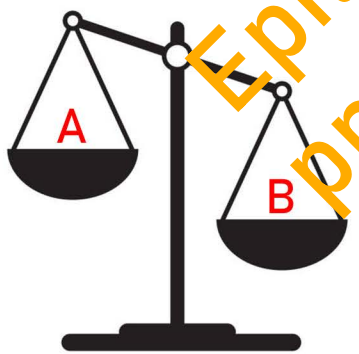
- Retention/Hold mode → Data is just retained/held
- Access transistors are cut-off → WL = 0 (GND)
- AC1 & AC2 are cut-off
- QT/QB nodes retain the voltage levels they posses
- Assume QT = 1 (VDD), QB = 0 (GND)
- PU2 & PD1 are cut-off
- PU1 & PD2 are on

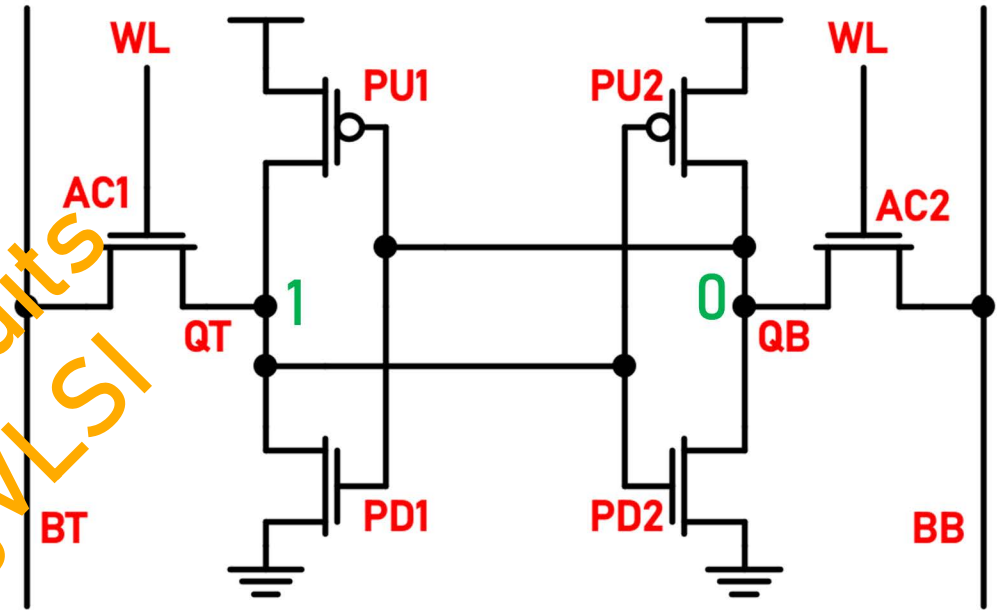


- Assignment: Draw the Circuit when QT/QB = 0/1

READ





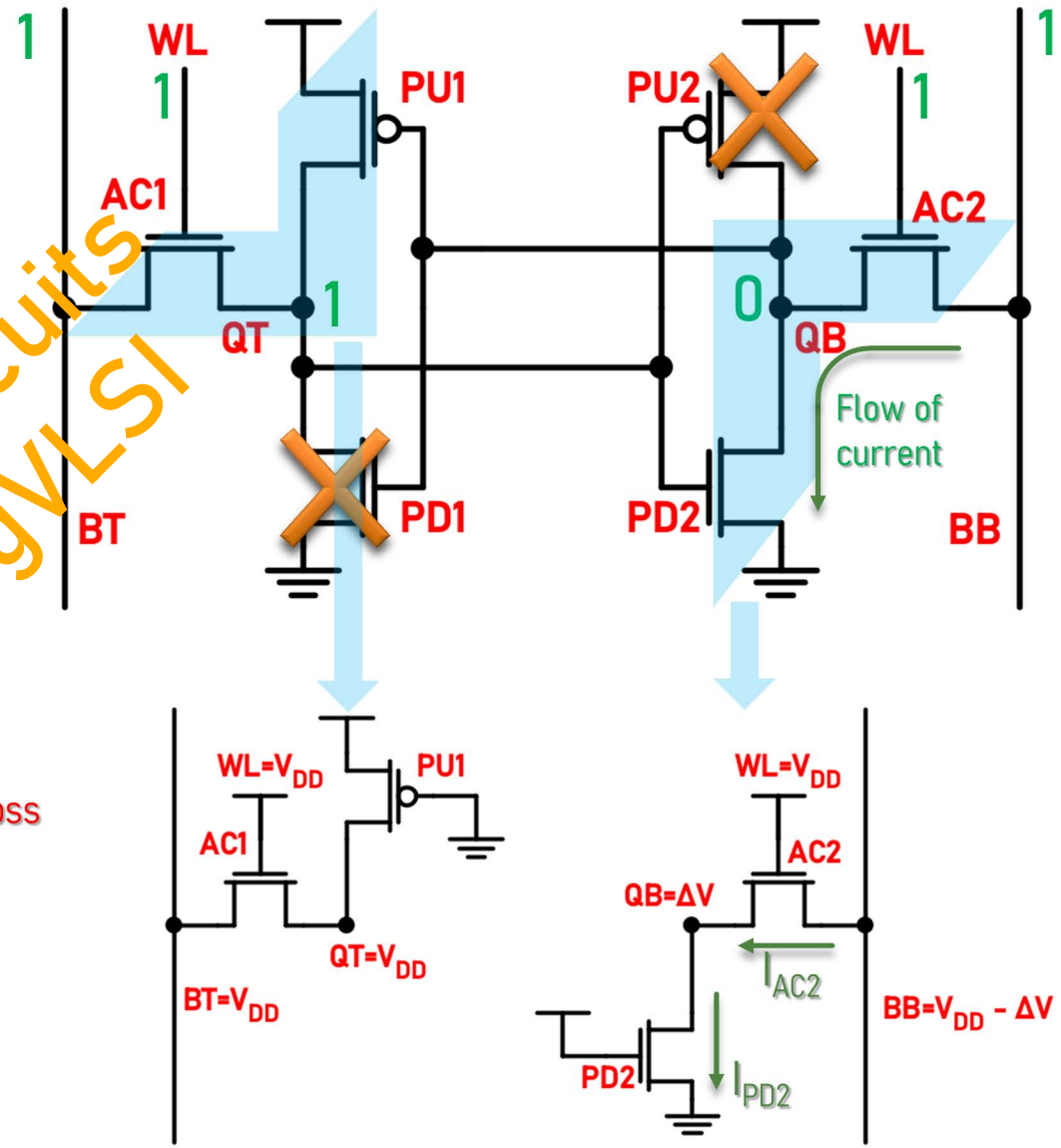


READ

- Assume QT/QB = 1/0
- Preconditioning of Bit-line pairs: BT/BB both charged to V_{DD}
- Access transistors are on → WL = 1 (V_{DD})
- PU2 & PD1 cut-off; PU1 & PD2 on
- Effectively the circuits look like
- On left side: since QT & BT are at same potential nothing happens
- On right side:
  - Pseudo-nMOS circuit → "nMOS inverter"
  - PD2 tries to pull down QB node to GND while AC2 tries to pull up QB node to V_{DD}.
  - AC2 & PD2 fight each other
  - Voltage at node QB rises slightly ≈ ΔV
- As QB node rises, the voltage on BB decreases slightly ≈ ΔV)
  →Voltage on BB ≈ V_{DD}-ΔV
- Read operation is successful if we are able to develop a differential voltage across bit-line pairs i.e., BB needs to be pull down slightly
- Therefore PD2 need to win the fight. Hence PD2 needs to be stronger than AC2
- w.k.t I_{AC2}=I_{PD2}

  For PD2; V_{GS}:V_{DD} V_{DS}:ΔV $V_{GS}-V_{T,n} > V_{DS}$ → Linear region

  For AC2; Drain & Gate are shorted to V_{DD} → Saturation region

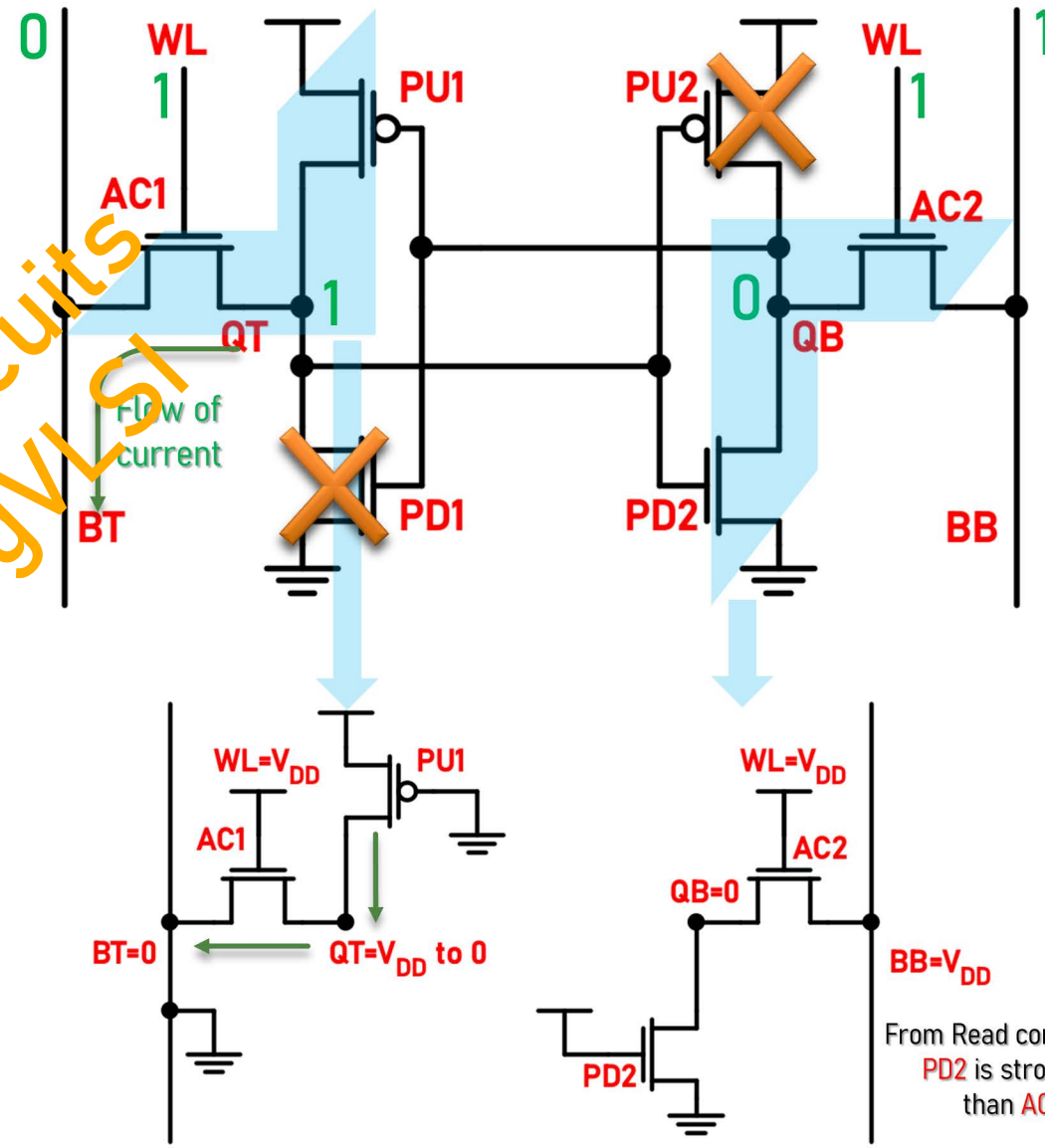


WRITE

- Assume QT/QB = 1/0 & 0/1 has to be written
- Preconditioning of Bit-line pairs by Write drivers: BT is discharged to GND & BB is charged to V_{DD}
- Access transistors are on → WL = 1 (V_{DD})
- PU2 & PD1 cut-off; PU1 & PD2 on
- Effectively the circuits look like
- On left side: AC1 tries to discharge QT while PU1 tries to charge QT. So there is a fight between AC1 & PU1
- On right side:
  - AC2 tries to charge QB while PD2 tries to discharge. From Read constraint, PD2 is stronger than AC2. So to hold 1 on QB, PD2 must be cut-off. For this QT must be pulled down to a point such that the RHS inverter trips.
  - To achieve this AC1 should discharge QT faster than PU1 charges QT i.e., AC1 should win the fight
  - Hence Write always begins by writing 0 at the node where 1 is stored
- w.k.t $I_{AC1}=I_{PU1}$

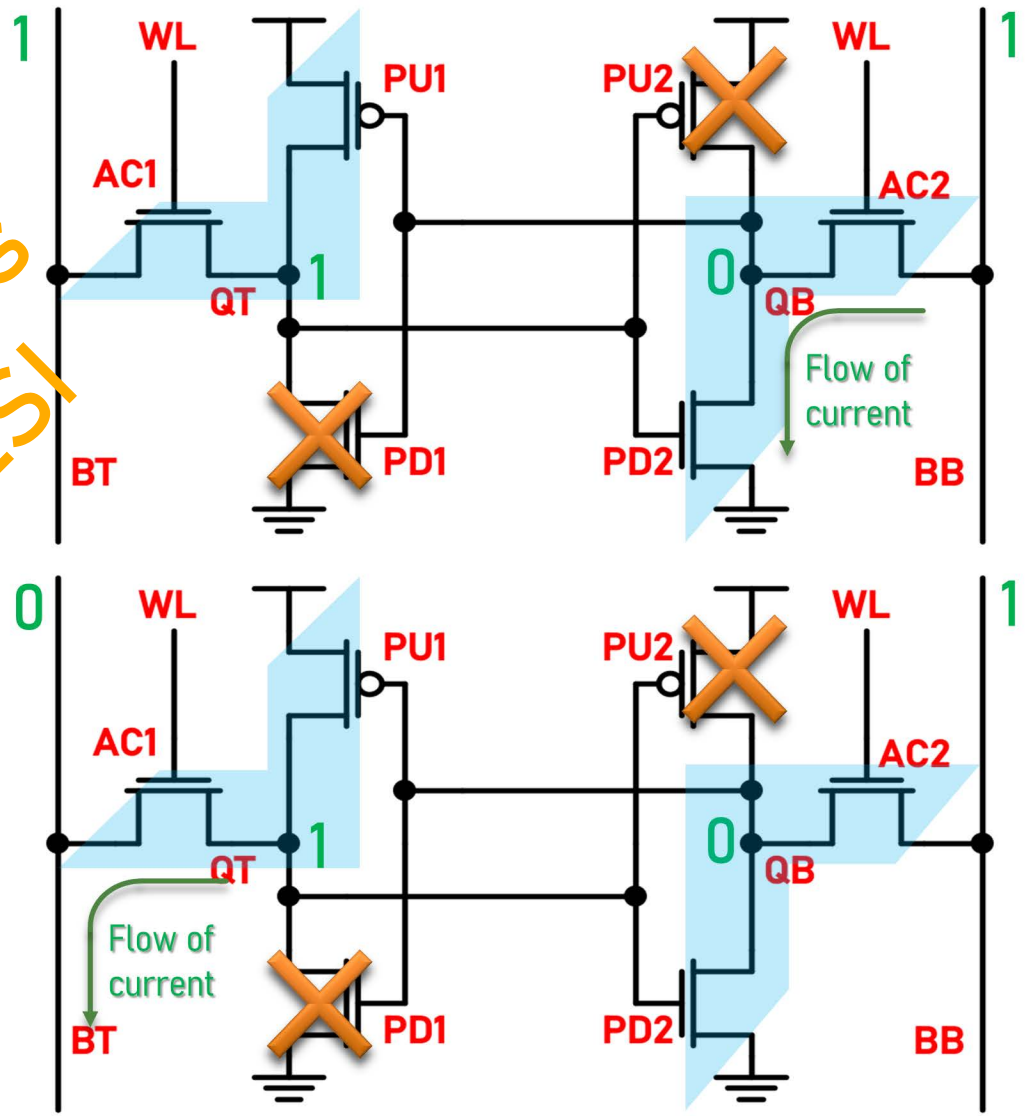
  For PU1; V_{GS}:-V_{DD} V_{DS}:-V_{DD} $V_{GS}-V_{T,p} > V_{DS}$ → Saturation region

  For AC1; V_{GS}:V_{DD} $V_{GS}-V_{T,n} > V_{DS}$ when $V_{QT} < -V_{T,n \text{ of } PD2}$ → Linear region



Summary

- There are two basic requirements which dictate the Cell ratios of the bitcell
  - The data read operation should not destroy the stored information in the SRAM Cell→ Flipping of cell is an issue
  - The cell should allow modification of the stored information during the data write phase→ Not flipping of cell is an issue
- Read Operation
  - Before every read/write operation, BL & BT are precharged to V_{DD}
  - When the WL is activated the current flows through either AC1 or AC2
  - If the node QB stores 0 BB discharges while BT remains at the same potential (V_{DD}) & vice versa
  - Voltage difference between nodes BT and BB is sensed by the sense amplifier (present in I/O) to detect it as 1 or 0
  - WL is deactivated after successful read
- Write Operation
  - To write 0 (1) BT is pulled to 0 V (V_{DD}), BB is pulled to V_{DD} (0 V) through a write driver circuit in I/O
  - When the WL is activated the current flows through access transistors and writes the data into the bitcell
  - Write always begins with write 0 at the node where 1 is stored
  - WL is deactivated after successful write



Bitcell sizing influence on Parameters

- Access Transistors (AC1 & AC2) : If made bigger
  - High I_{READ} → Improves Read access time
  - Better Write margin
  - High I_{LEAK}
  - Load on Bit-line pairs increases → So Slow discharge rate → may takes time to develop differential voltage hence may affect read operation
  - Degraded SNM → It has to be kept weaker than PD
- Pull Down Transistors (PD1 & PD2) : If made bigger
  - Improved SNM
  - Degraded Write margin
  - Area Increases
    - Keeping the Area concern in mind size of PD devices are optimally sized such that they are stronger than AC devices (or equally strong some times)
- Pull Up Transistors (PU1 & PU2) : If made bigger
  - Good SNM
  - Degraded Write margin
  - Area Increases
  - For successful write operation need to be kept weaker than

Cell Beta Ratio & Cell Pull-up Ratio

Beta Ratio

Cell Beta Ratio = $\frac{\text{Width of PD}}{\text{Width of AC}}$

Higher the Beta value better is SNM. So higher Beta favours cell stability but has a negative impact on I_{READ}

Similarly lower Beta value increases I_{READ} however it increases the risk of data flips → Less stable

Pull-up Ratio

Cell Pull-up Ratio = $\frac{\text{Width of AC}}{\text{Width of PU}}$

A failure to write occurs when the access transistor is not strong enough to overpower pull-up pMOS & pull the internal node to ground (Writing '0')

Different Types of Bitcell

HD → High Density

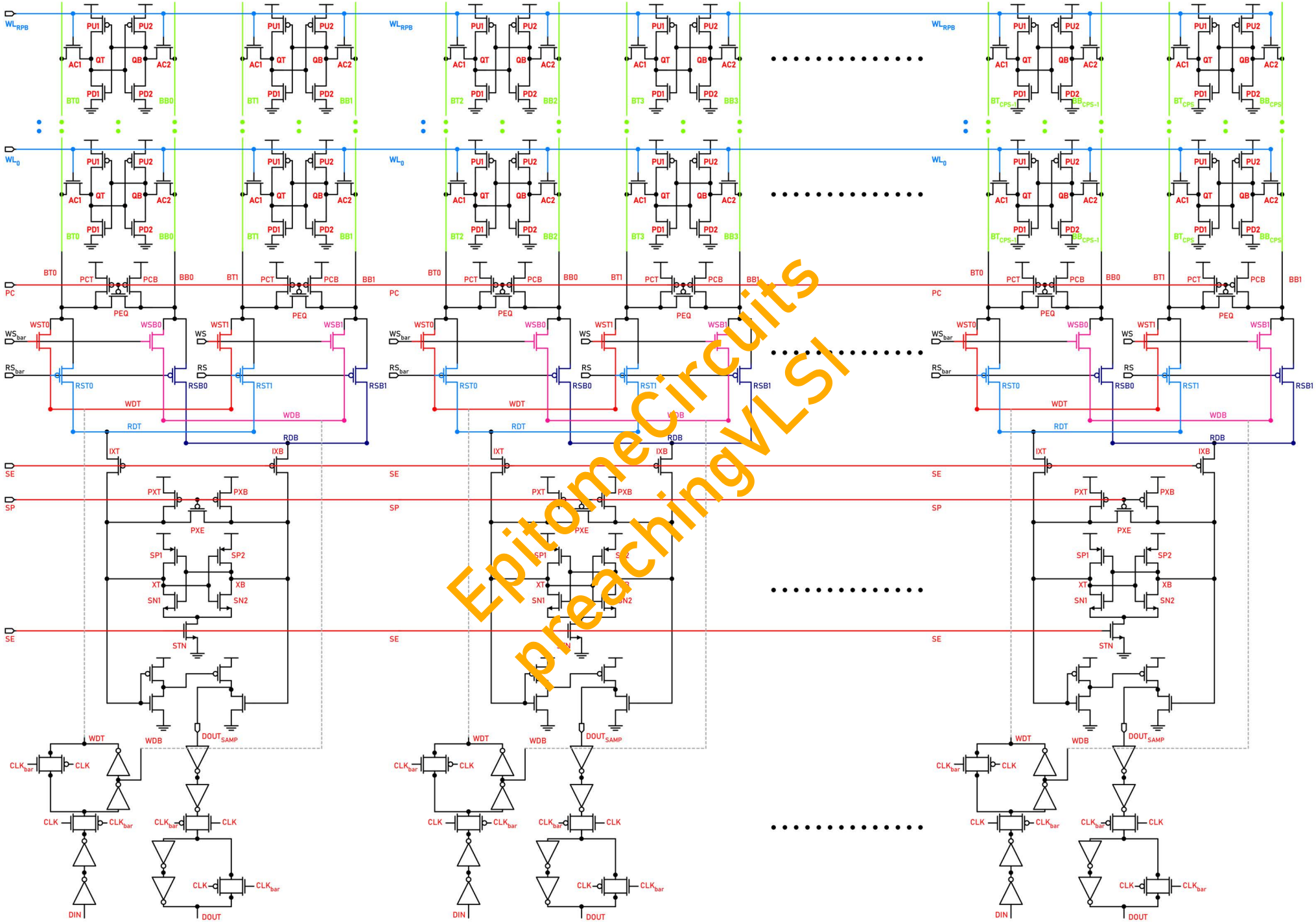
- PU:PG:PD = 1:1:1
- Lowest Single Bitcell footprint
- Better in leakage
- I_{READ} is comparably less & Slower
- Compromised performance

HC → High Current

- PU:PG:PD = 1:2:2 or 2:3:3
- Occupies More Area than HD
- Higher leakage
- Improved Read-write operation compared to HD
- Requires Bigger SA & drivers

HP → High Performance

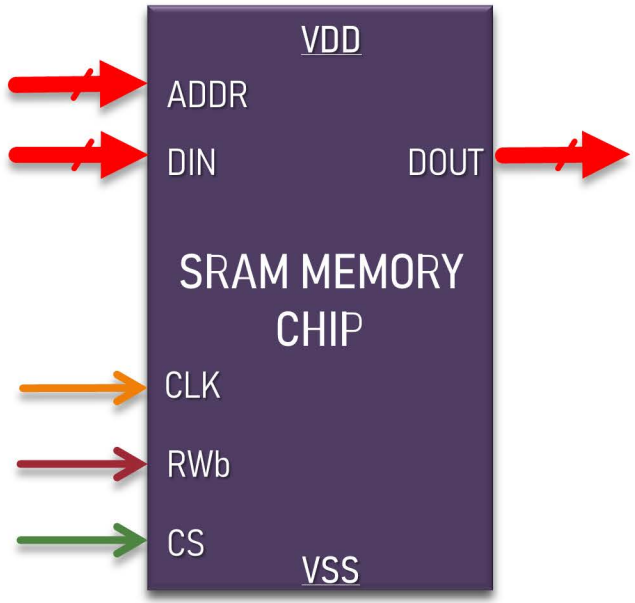
- PU:PG:PD = 1:2:3 or 2:3:4
- Occupies More Area than HD/HC
- Improved Read-Write operation & reduced leakage & hence lower power consumption
- Requires Bigger SA & drivers

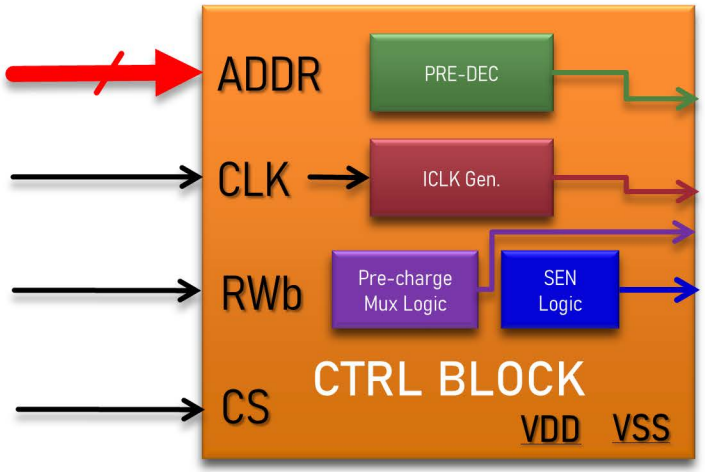


Control Block

Control block Mainly comprises of following sub blocks & pins

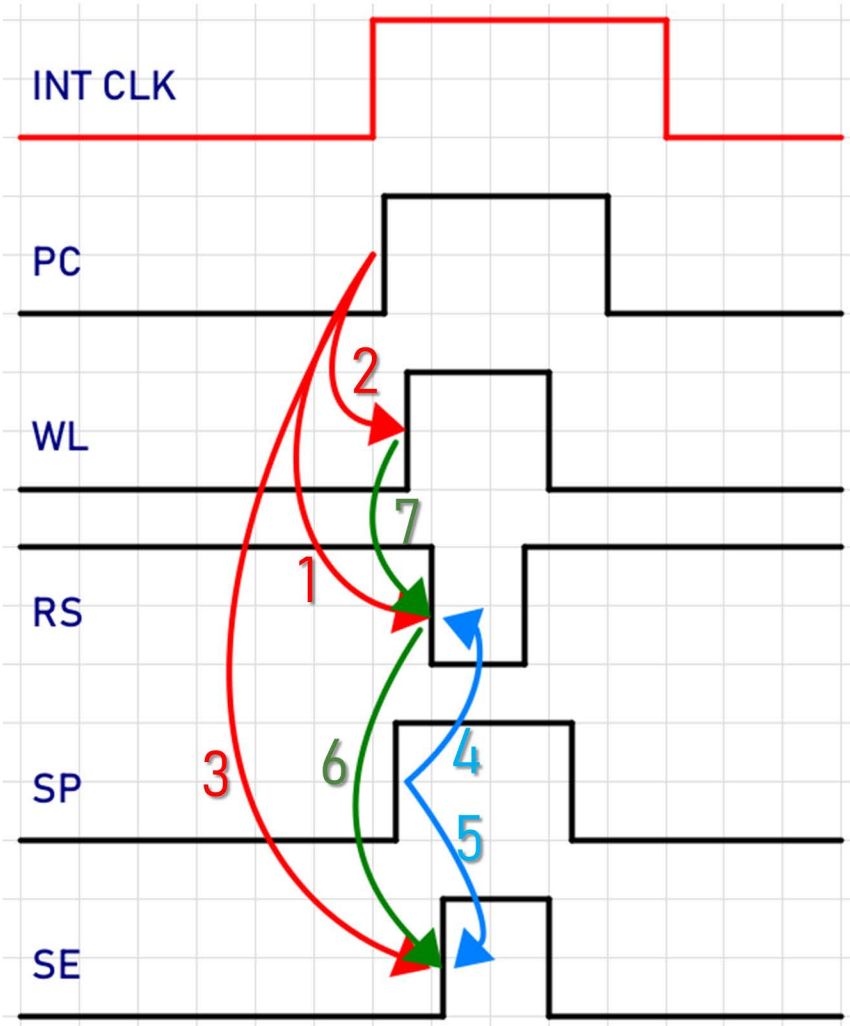
1. Pre-decoder logic
2. Address latches
3. Internal clock generator logic
4. Sense enable generator logic
5. Column mux control signal generator logic
6. Precharge control logic
7. I/P pins
    1. CS → Chip Select: Used to enable Memory chip
    2. CLK → Global Clock: Used to generate internal clock & synchronize the memory operations
    3. ADDR[8:0] → Address Bits
    4. RWb → Read Enable (or Write Enable Bar)
        - high → read
        - low → write
    5. Supply → VDD,VSS

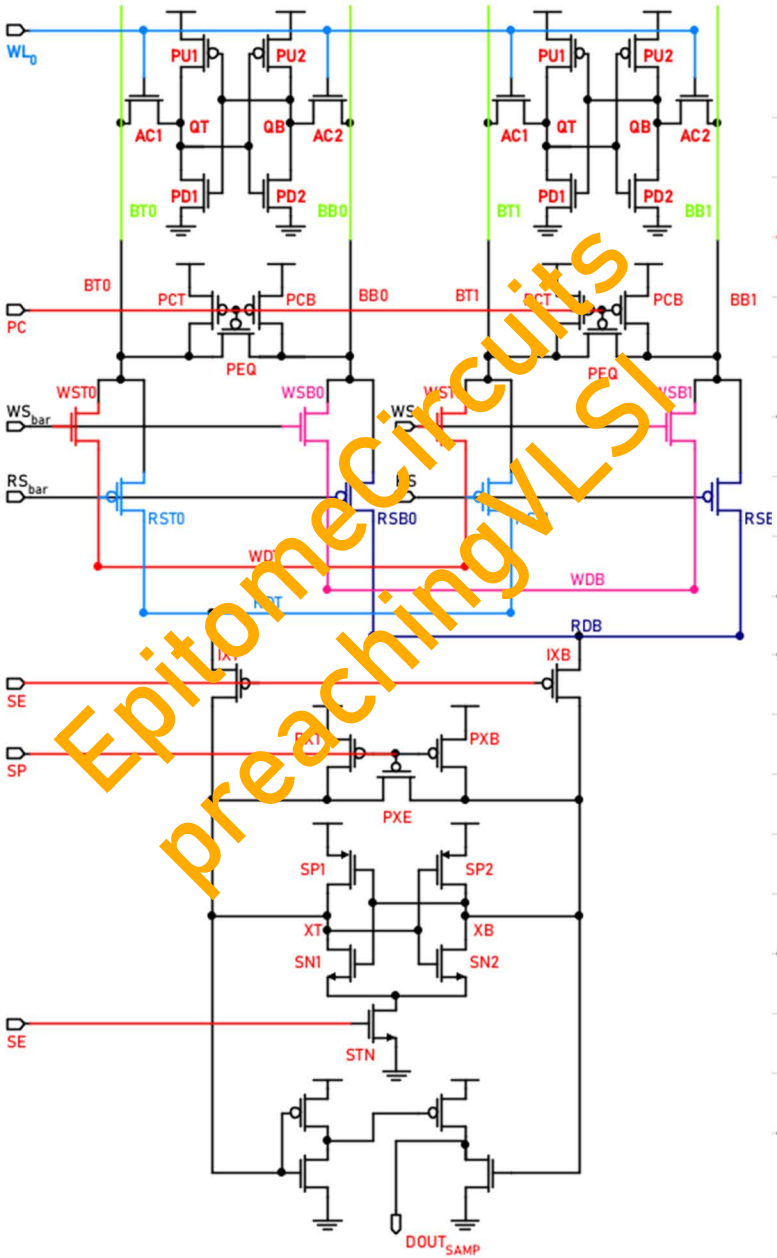




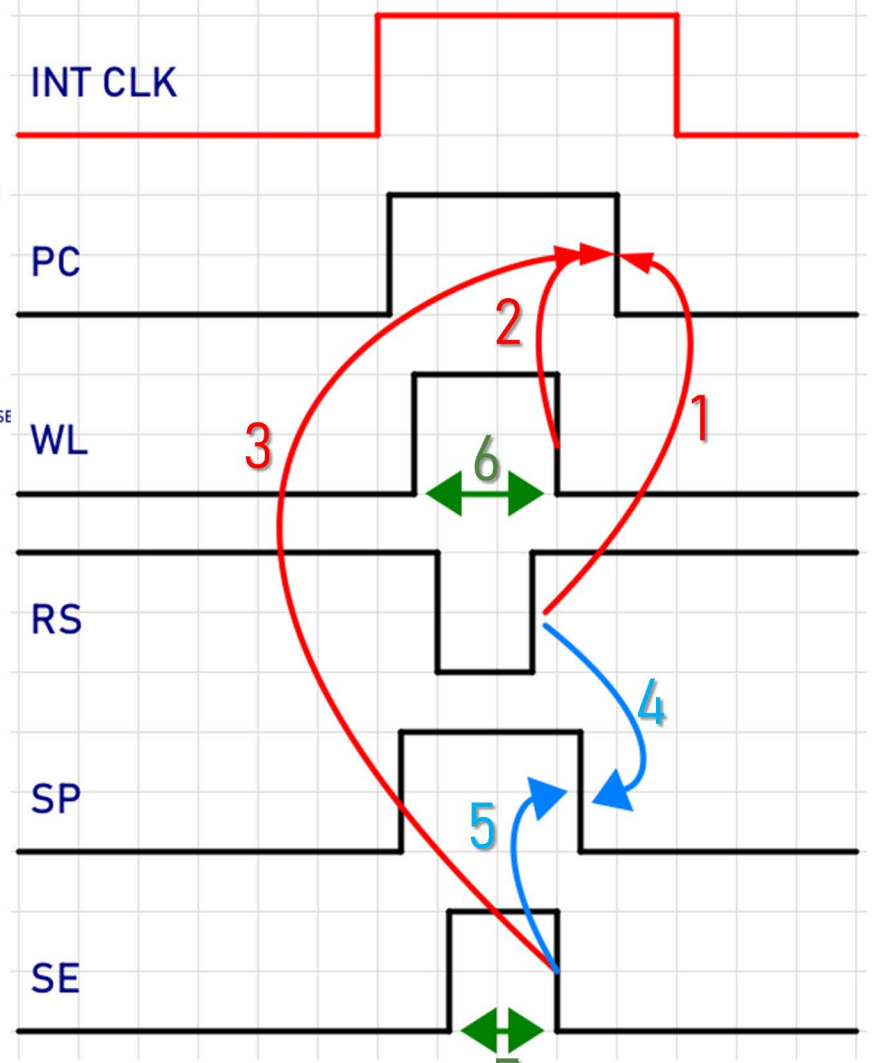
Read Operation Sequence

Read Setup Timing Sequence



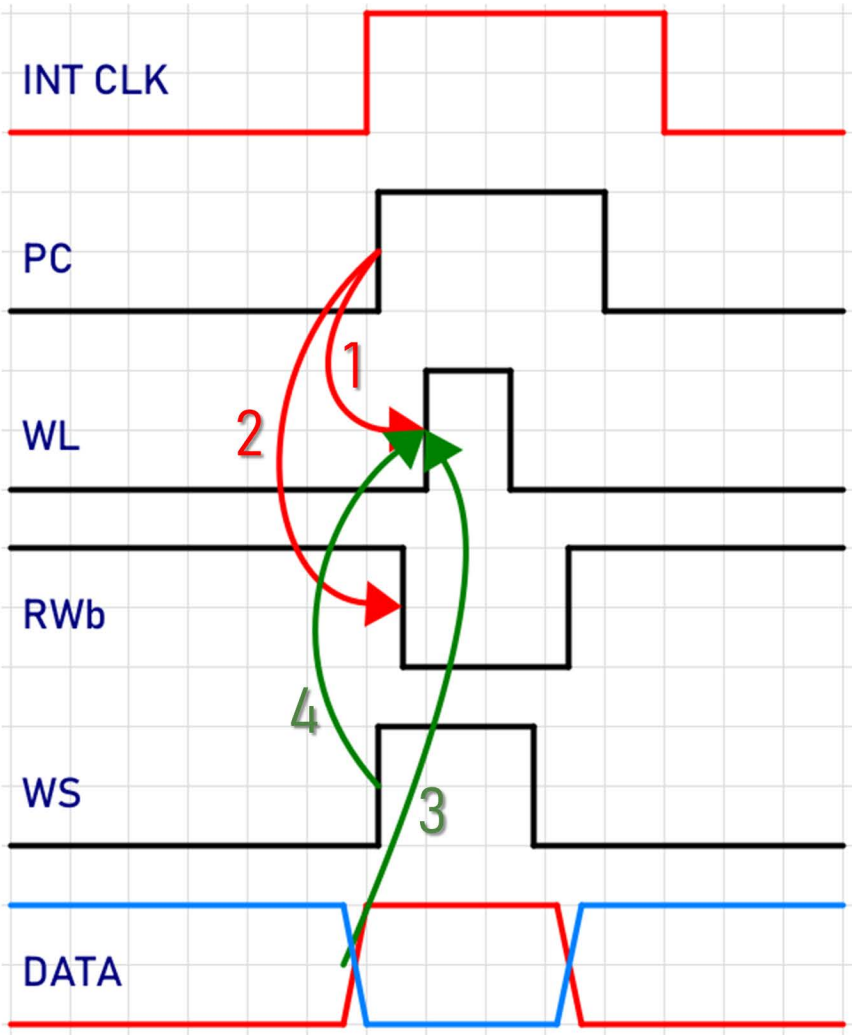


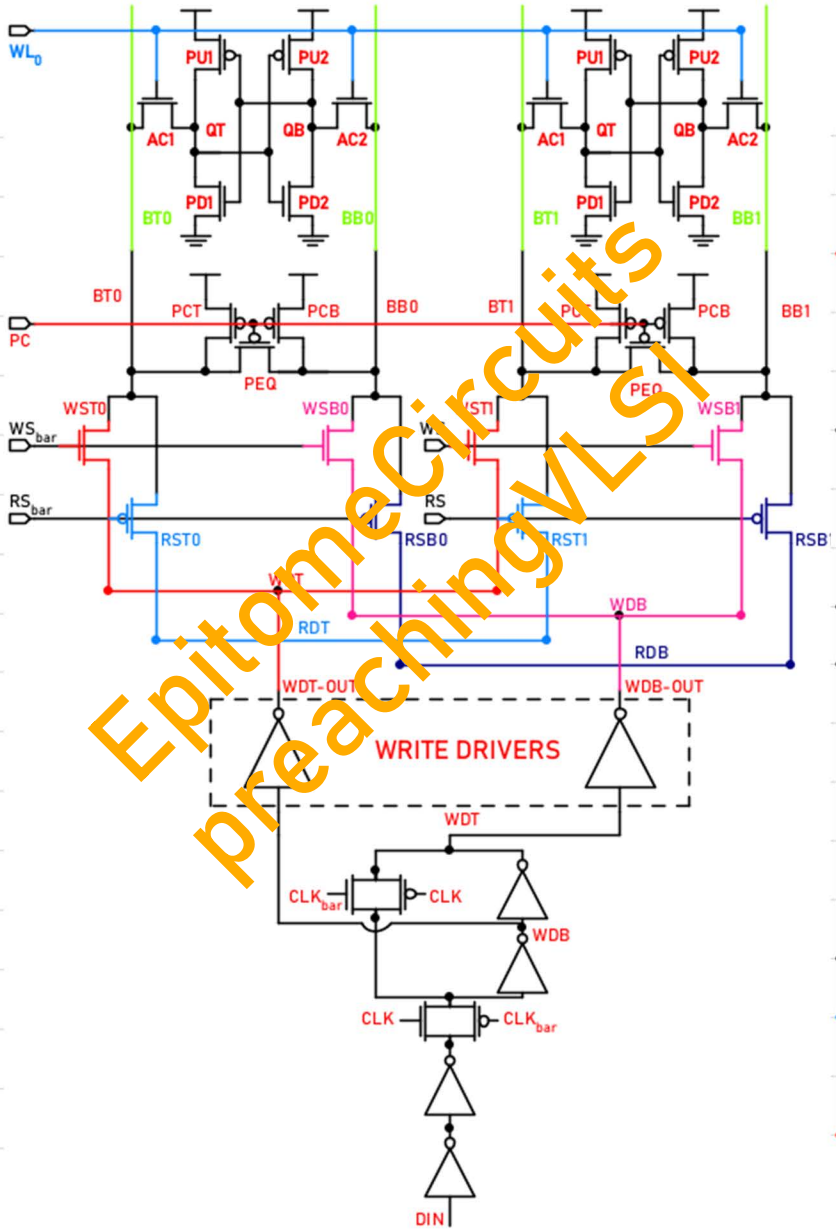
Read Hold Timing Sequence



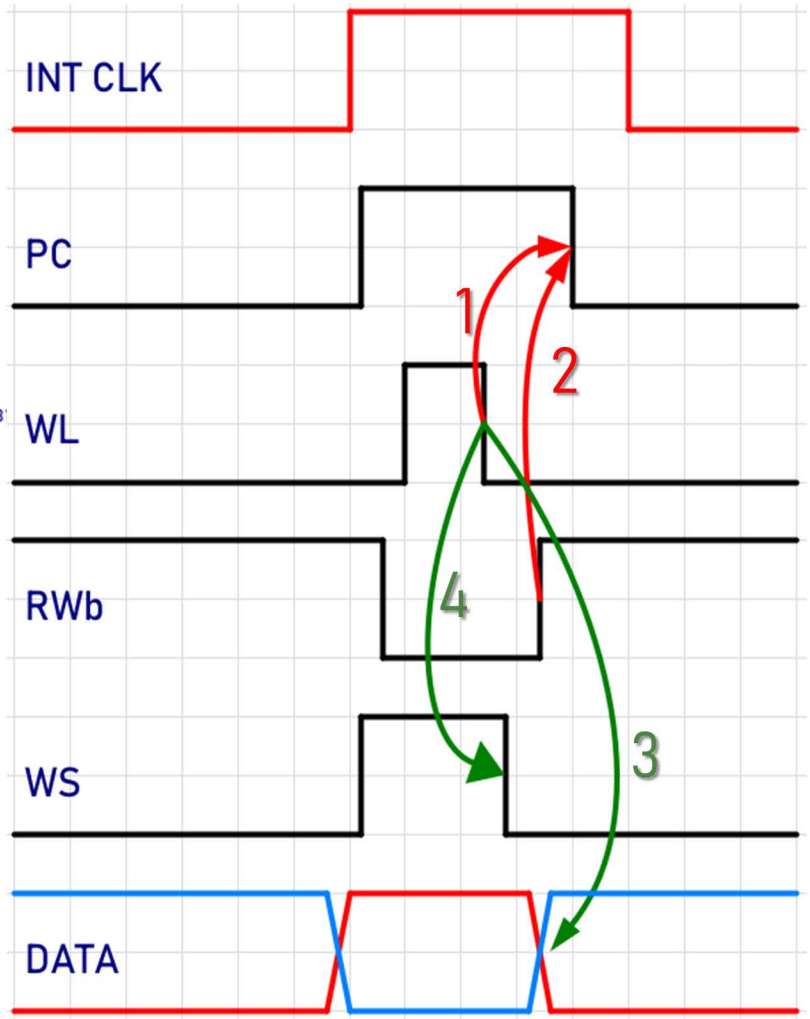
Write Operation Sequence

Write Setup Timing Sequence

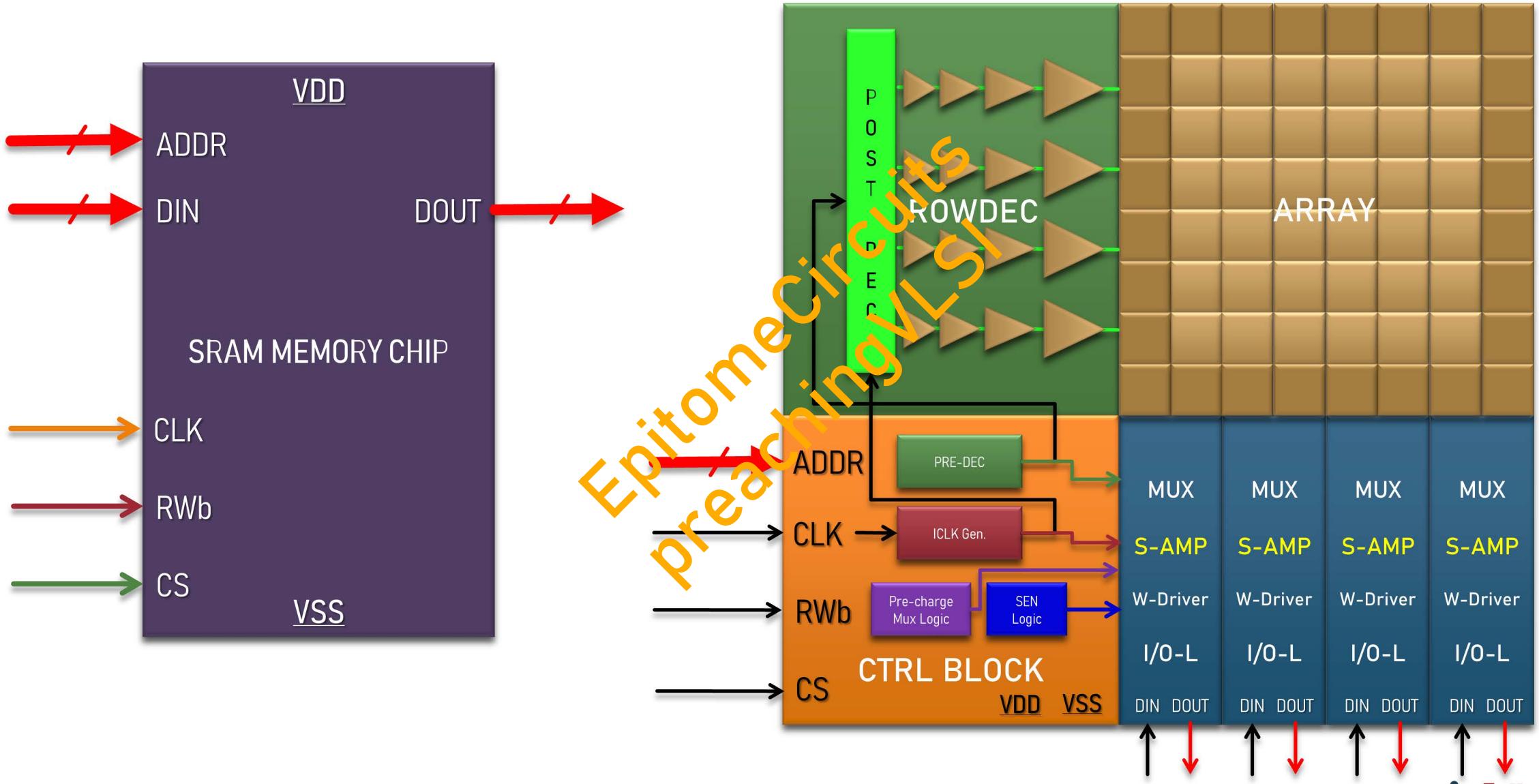




Write Hold Timing Sequence

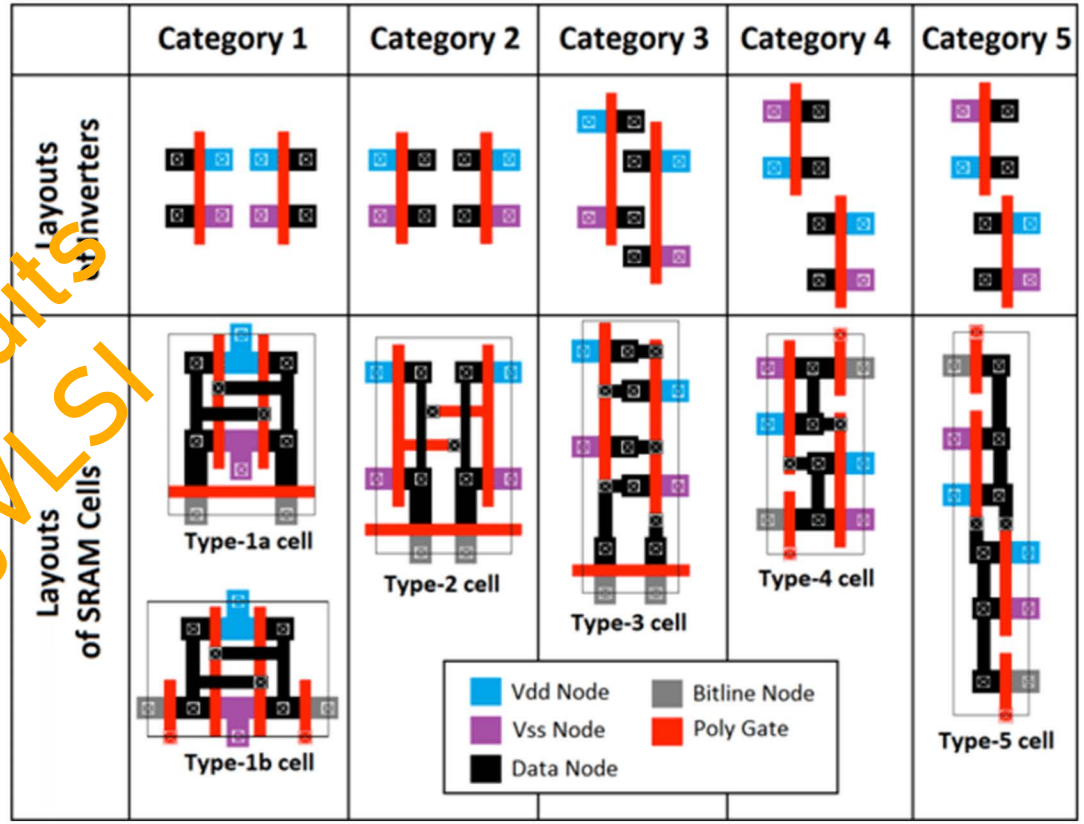


Memory Chip



Different Layouts of SRAM

- Various types of 6T SRAM cell layout architectures and corresponding 16-bit arrays have been implemented and compared at the 32 nm, in terms of
  - area,
  - power dissipation
  - read/write delay.
- Some cells are properly flipped horizontally or vertically in order to partially merge and overlap with adjacent cells.
- This results in different cells sharing the same polysilicon, diffusion or n-well areas, as well as metal wires and contacts.
- Furthermore, n-well taps and substrate contacts may be shared among multiple cells for additional area efficiency.
- The thin cell topology(Type-4 cell) has proved to be the best design on all aspects.



Write delay in ps					
Cells	Supply voltage in V				
	0.4	0.5	0.6	0.7	0.8
Type 1b	21	12	8.5	7.5	6.5
Type 2	20	11.5	8.5	7.5	6
Type 4	**17**	**10**	**7.5**	**6**	**5.5**
Type 5	22	13	10	8	7

Read delay in ps					
Cells	Supply voltage in V				
	0.4	0.5	0.6	0.7	0.8
Type 1b	21	11	8	6	6
Type 2	20	11	8	6	5
Type 4	**20**	**11**	**8**	**6**	**5**
Type 5	20	11	8	6	5

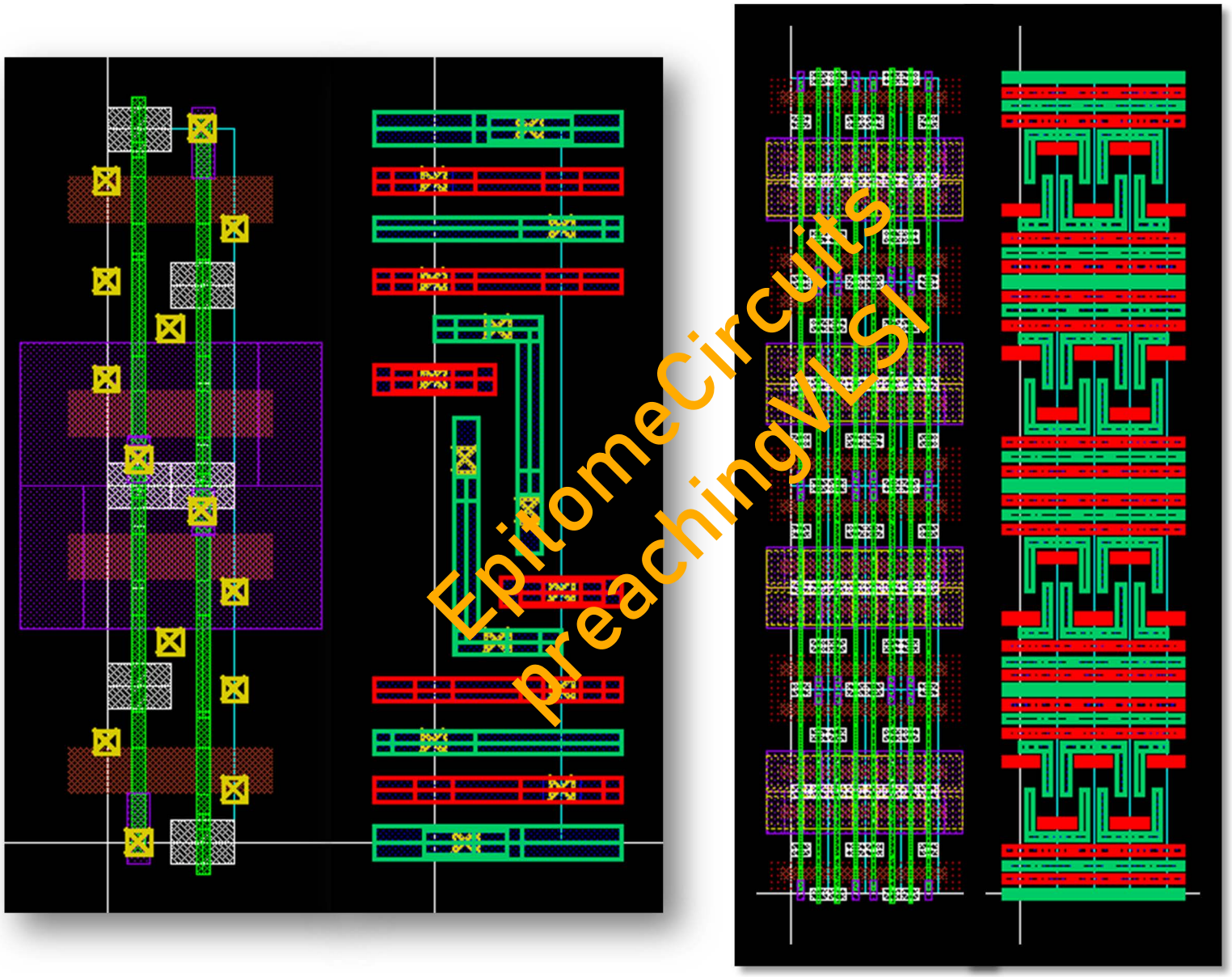
Power dissipation of SRAM Bitcells in nW					
Cells	Supply voltage in V				
	0.4	0.5	0.6	0.7	0.8
Type 1b	14	25	37	52	71
Type 2	13	24	36	51	69
Type 4	**12**	**22**	**32**	**45**	**62**
Type 5	15	27	40	56	77

Power dissipation of SRAM Arrays in nW					
Arrays	Supply voltage in V				
(16-bit)	0.4	0.5	0.6	0.7	0.8
Type 1b	112	180	270	392	560
Type 2	109	176	260	369	522
Type 4	**99**	**158**	**236**	**343**	**492**
Type 5	109	174	262	438	599

Area and bit density of 16-bit SRAM arrays				
Arrays (16-bit)	Width in µm	Height in µm	Area in μm^2	Bit-Density in μm^2/bits
Type 1b	2.34	2.13	4.984	0.312
Type 2	1.59	2.355	3.744	0.234
Type 4	**2.655**	**1.2**	**3.186**	**0.199**
Type 5	4.38	1.08	4.73	0.296

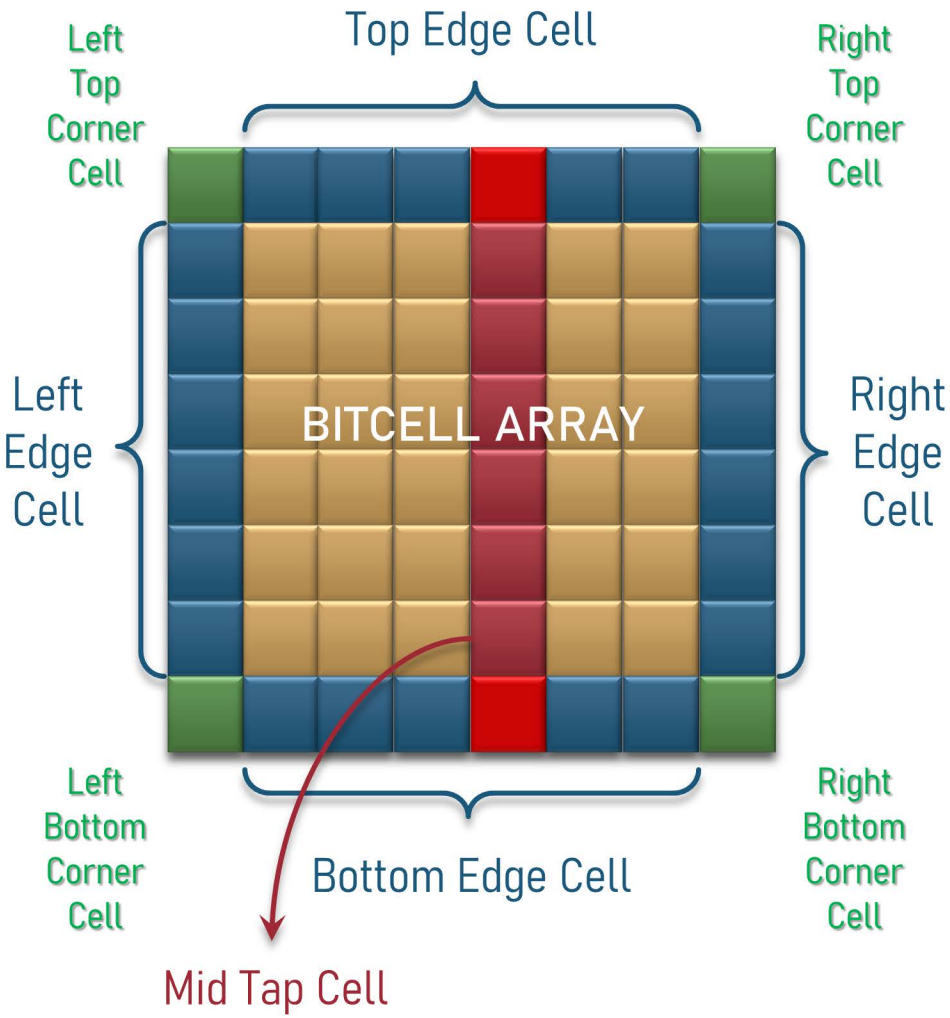
Source: https://www.researchgate.net/publication/312094888_Design_and_Simulation_of_6T_SRAM_Cell_Architectures_in_32nm_Technology

6T-SRAM HD Bitcell Layout

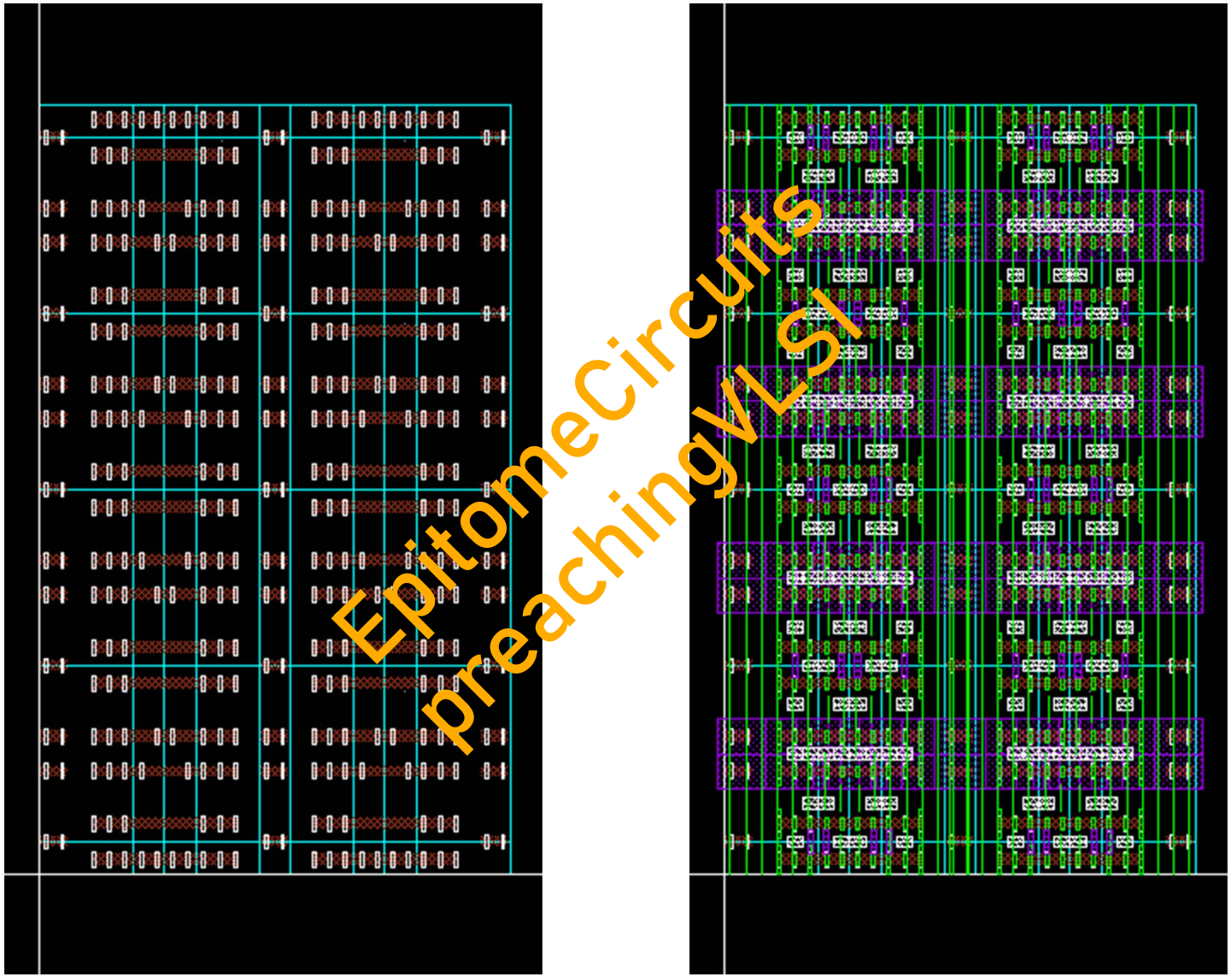


Edge cells

- These cells are used to complete the end connections of a block and to fulfill DRC requirement.
- They also provide tap connections & increase the reliability/yield by providing identical RC environment for the Bitcells at the perimeter
- They generally comprise of tap cell and/or dummy bitcell
- Dummy bitcells are created by shorting all 3 terminals of the transistors to supply rails. They are not used to store data & hence no read/write operations are performed on them. The only purpose of Dummy bitcell is to mimic the RC environment for the bitcells at the boundary
- In order to clean LVS for each edge cell, separate schematic and symbols are created.
- In lower technology nodes device formation can be inhibited by using CUT layers. In such cases, there is no need of creation of schematic for edge cells to clean LVS
- Different types of edge cells are:
  - Left & Right edge cells
  - Bottom & Top edge cells
  - Corner cell
  - Mid-tap cells
    - To provide tap connections in the middle of a bigger array & avoid latch-up



4x4 Bitcell Array



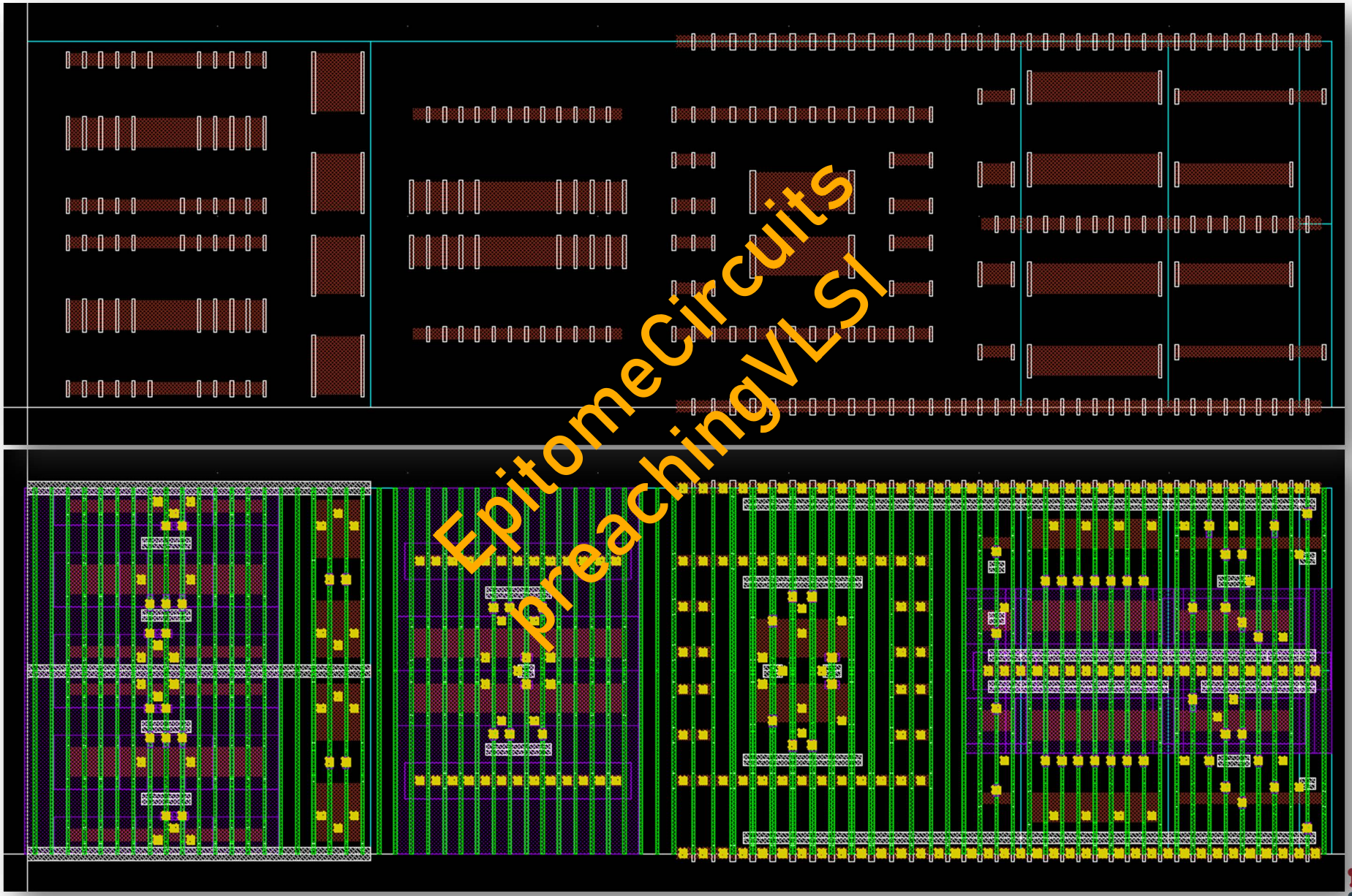
Column-I/O Layout Guidelines

- Column I/O has following sub-blocks

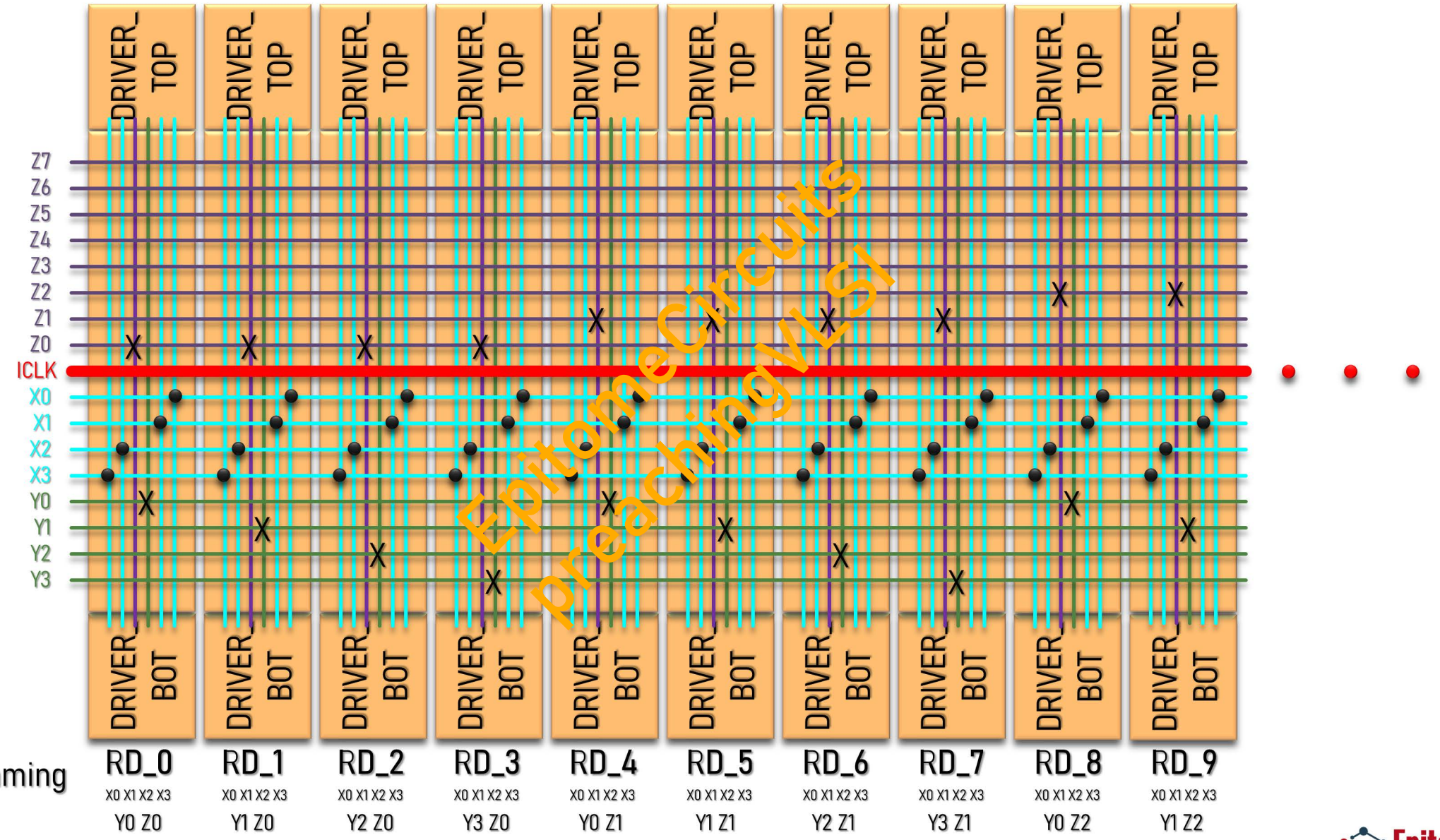
  Precharge Mux Sense Amplifier I/O Latch Write Driver Antenna diode

- Height of the column-I/O depends on the column-mux size
  - For Mux-2; the column I/O height is kept 2 bitcell height
  - For Mux-4; the column I/O height is kept 4 bitcell height
  - Hence these leaf cells are called "Pitched-cells"
- Guidelines for drawing Pins
  - Keep the pins in the topmost layer. For ex:
    - VDD/VSS pins in M4
    - Column-I/O global signals(PC,SP,SE,SEB,CLK etc..) in M2
    - BT/BB in M1
  - Keep the pins square shaped
    - If the pins are kept on a Metal whose width is 64nm then keep the height & width of the pin equal to 64nm.
    - If the pins are kept on a Metal whose width is 32nm then the height & width of the pin layer will be 32nm
- Follow the half-cell approach for all the sub-blocks

Column I/O



RowDecoder Central Part

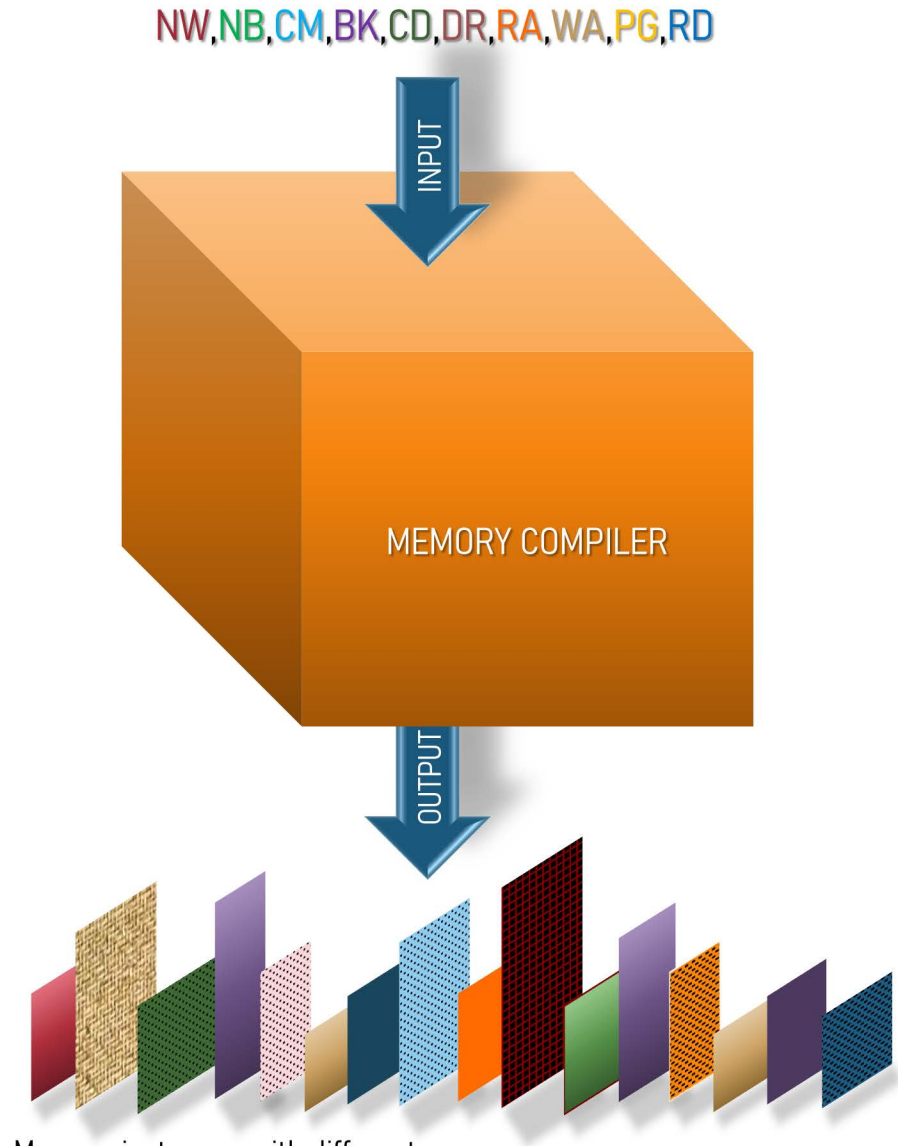


● → Hard connection

X → Programming connection

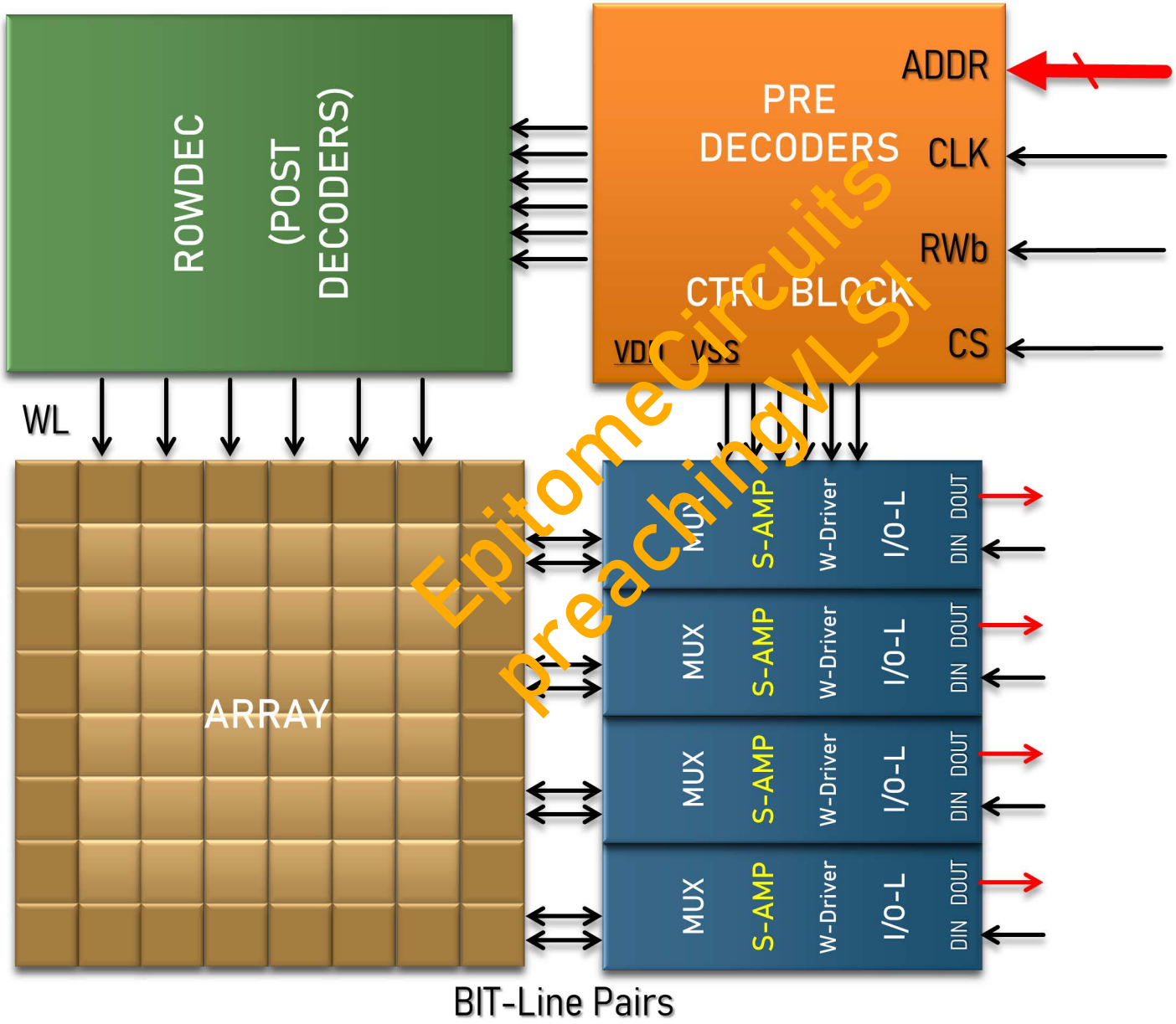
Memory Compilers

- Memory Compilers are software tools that can generate memory instances of different types and configurations based on user-defined parameters for specific technology node.
- The best analogy for a memory compiler is a Car Manufacturing company where it manufactures different types of Cars based on the market requirements. Some are fuel efficient, some can be driven very fast, some are stylish & sporty etc.,
- Manually creating memories can be time consuming and tedious and the designs are usually not so flexible
- Compilers can be parameterized by number of Words, number of Bits per word, desired Aspect ratio, number of Banks, degree of Column Muxing, Decoding style, addition of special features like Read Assist, Write Assist, Power Gating, Dual Rail, Redundancy etc., thus improving PPA (Power, Performance & Area)
- Different Types of compilers available in the market:
  - Based on Speed → High Speed, Ultra High Speed, High Density, Ultra High Density
  - Based on Number of Ports → Single Port, Dual Port
  - Memory Core → SRAM core, Register File, ROM core
  - Based on Read-Write Operation → 1 Read & 1 Write per Clock cycle, 1 Read or 1 Write per Clock cycle



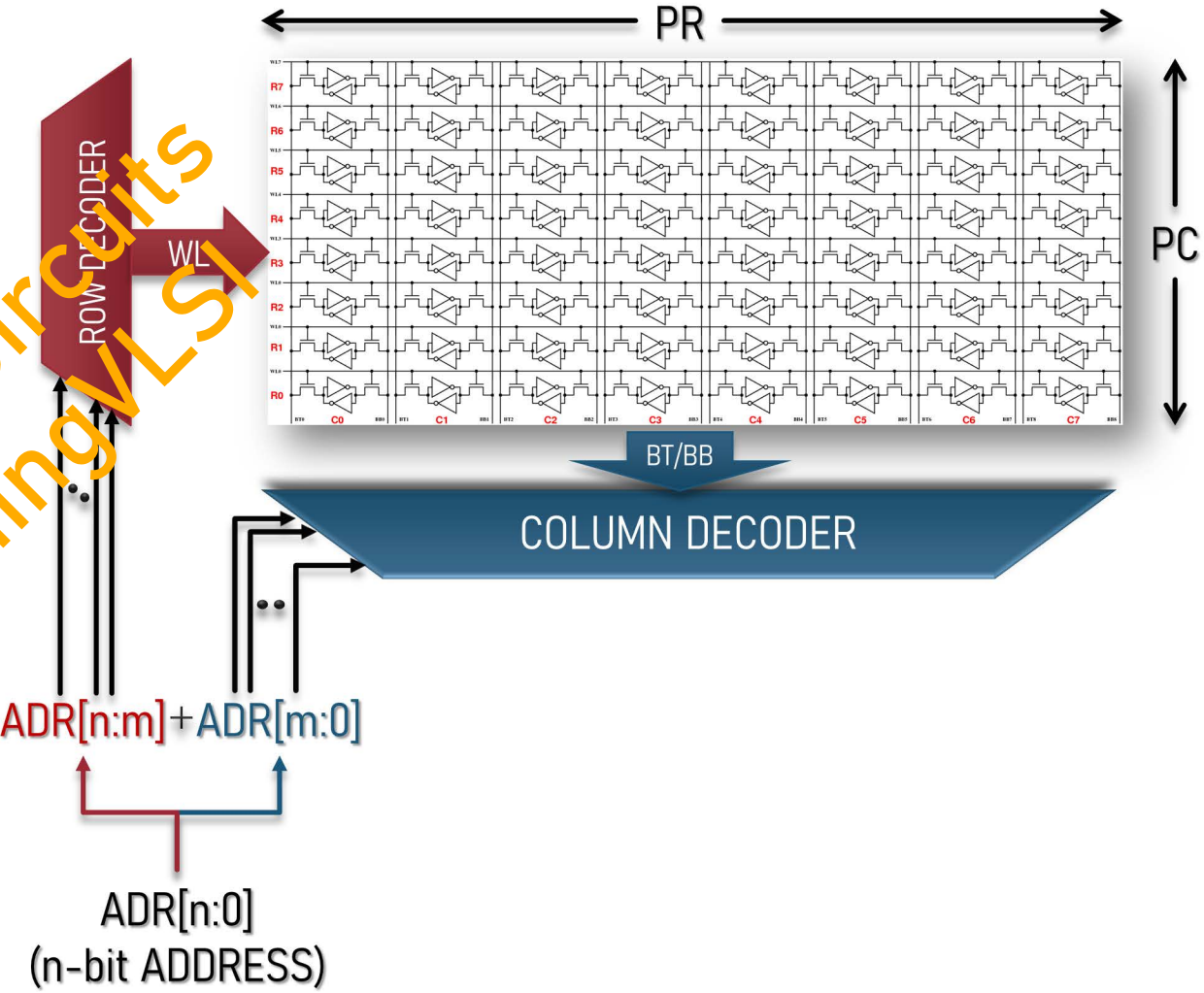
Memory instances with different sizes & features enabled

Memory Block Diagram (Floor Plan)



Memory Configuration

- Given the memory instance size i.e., NW x NB
  - Which CM-BK-CD configuration is best for AREA?
  - Which CM-BK-CD configuration is best for PERFORMANCE?
  - How does changing CM affects aspect ratio of Memory chip?
  - What is the most optimum decoding strategy?
  - Why 'Bank'ing is required?
  - How does 'Bank'ing improves performance?



Memory Instances