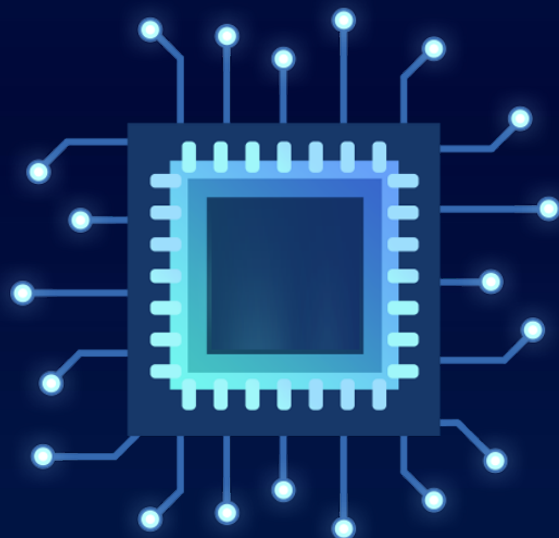




基于静态随机存储器的存算一体化进展与挑战

The Progress and Challenge of CIM based on Static Random Access Memory

汇报人：蔺智挺

时间：2022年6月22日



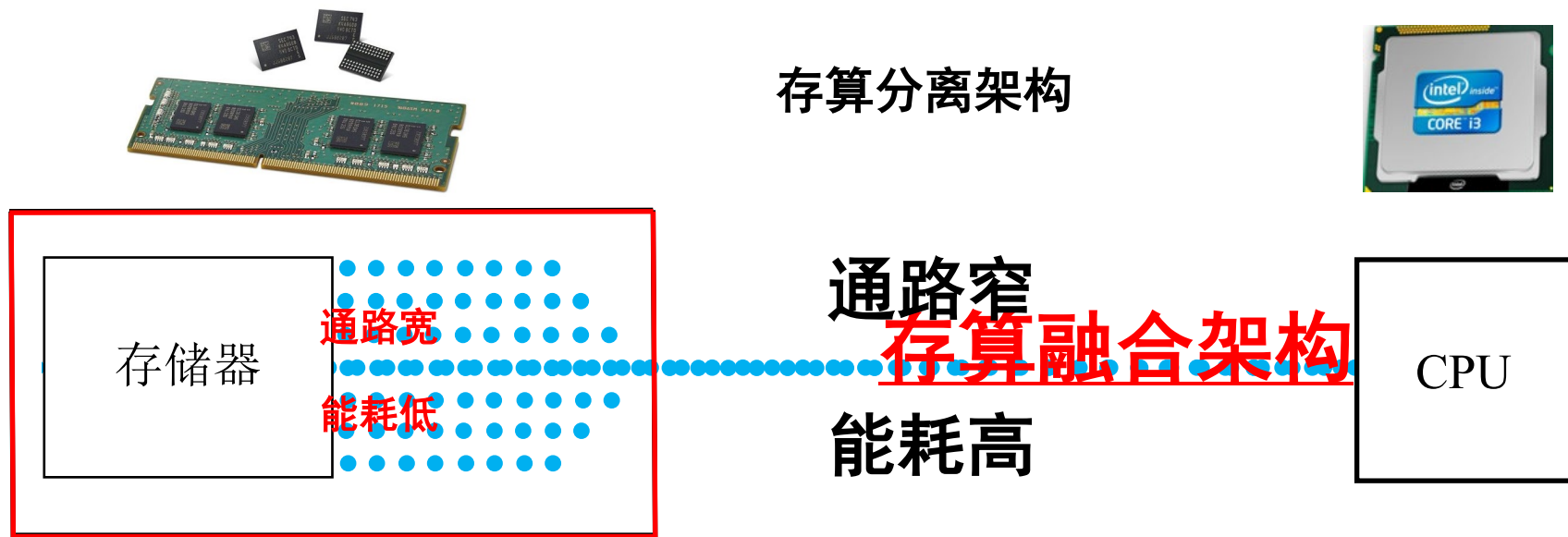
PART·ONE

基于SRAM存内计算研究进展

RESEARCH STATUS OF IN-MEMORY COMPUTING BASED ON SRAM

1 基于SRAM存算技术背景

随着“**算力时代**”到来，大规模数据需要在存储器和处理器之间往返，然而传统冯诺依曼架构中计算与存储分离，无法满足频繁访问的需求。存内计算技术的诞生打破了传统计算架构中的“**存储墙**”，对于“**算力时代**”具有**革命性意义**。



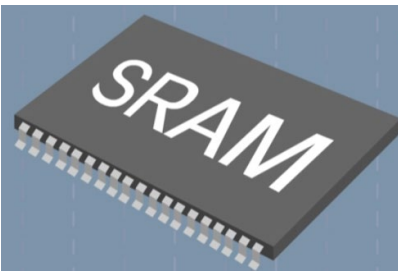
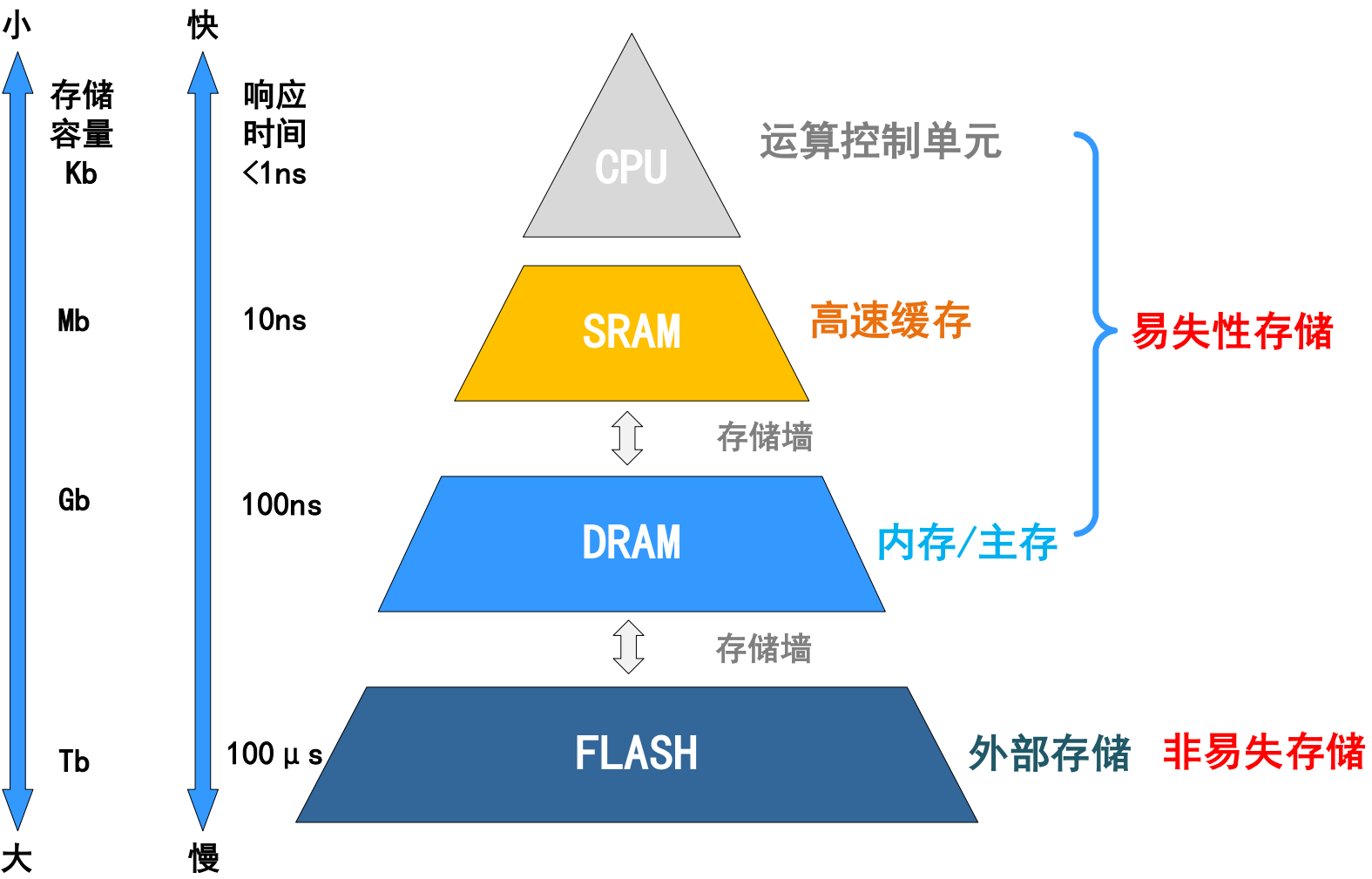
- ◆ 大数据和AI时代，数据量和算力需求增长迅猛。
- ◆ 传统存储器和处理器分离的**Von Neumann架构**存在“存储墙”问题。
- ◆ 围绕存储器开展**战略性技术攻关**是国家亟需、产业亟需！

➤ 存储金字塔

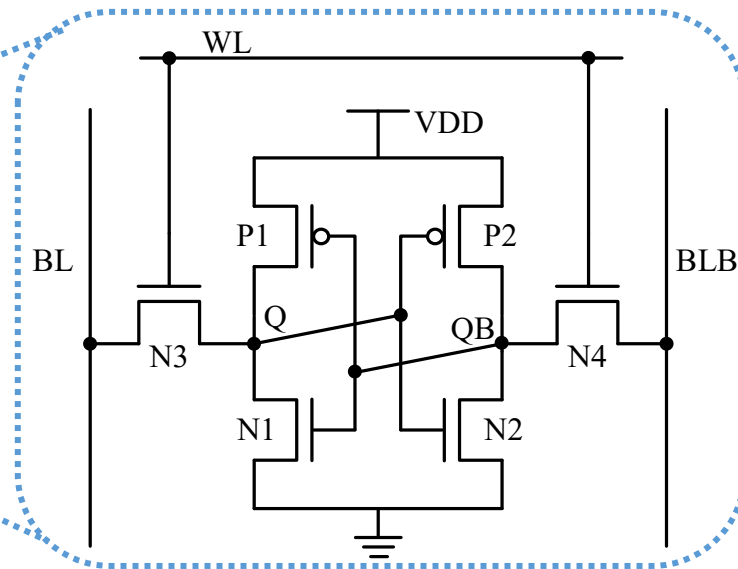
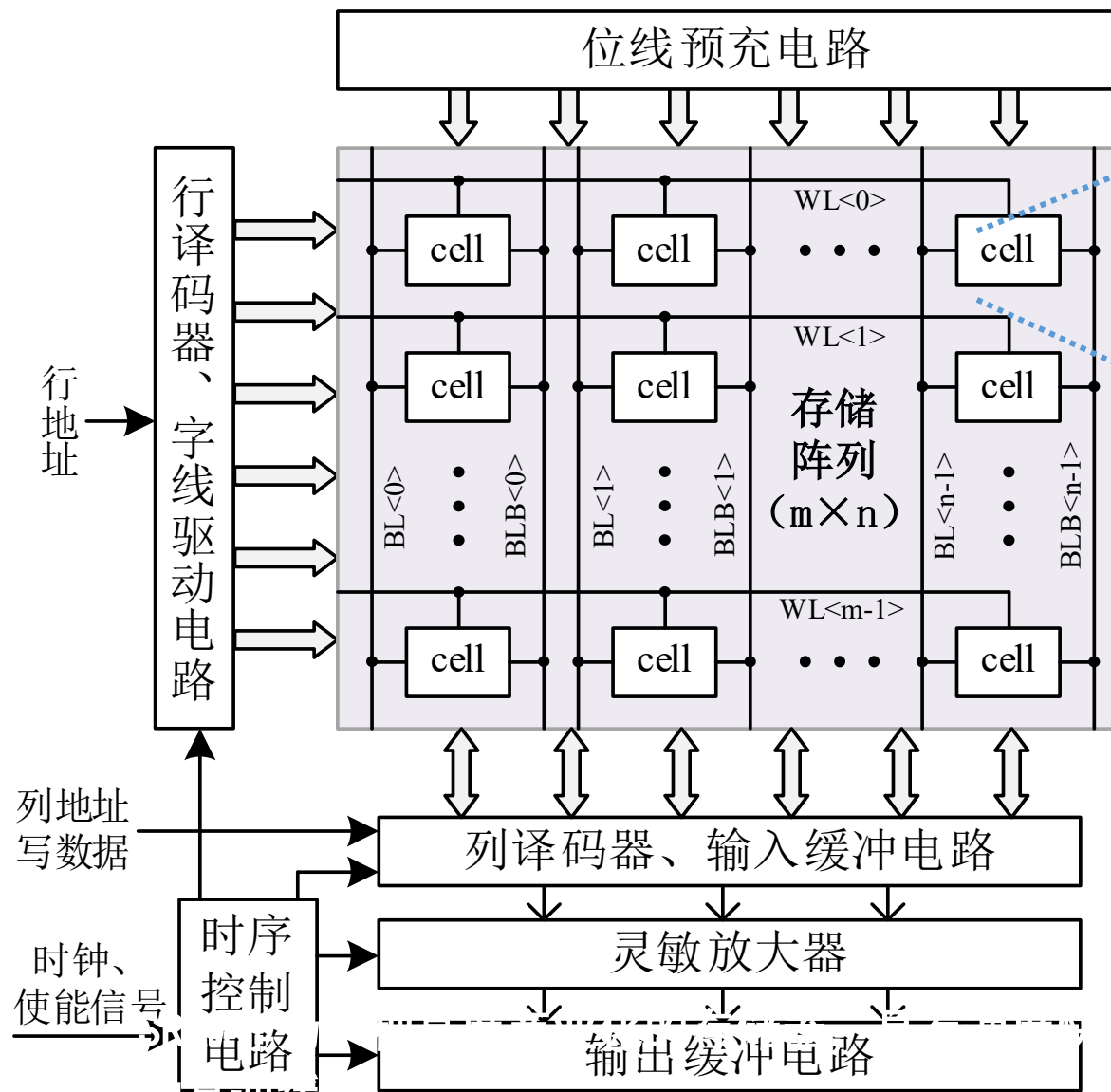


静态随机存取存储器(SRAM), 利用双稳态结构来存储数据。SRAM的**速度快**, 但在相同面积中SRAM的容量要比其他类型的**存储器小**。

SRAM的应用:
主要用于**高速缓存**。
作为微控制器的**RAM**或者**cache**
用于**SOC, FPGA与CPLD**



- ✓ 较高的性能
- ✓ 功耗小
- ✗ 集成度低



SRAM五大组成模块：

存储阵列 译码电路

灵敏放大器 时序控制电路

输入输出电路

1 基于SRAM存算技术背景

算力是人工智能的第一生产力！

Overall framework of SRAM-based

CIM

Circuit level:

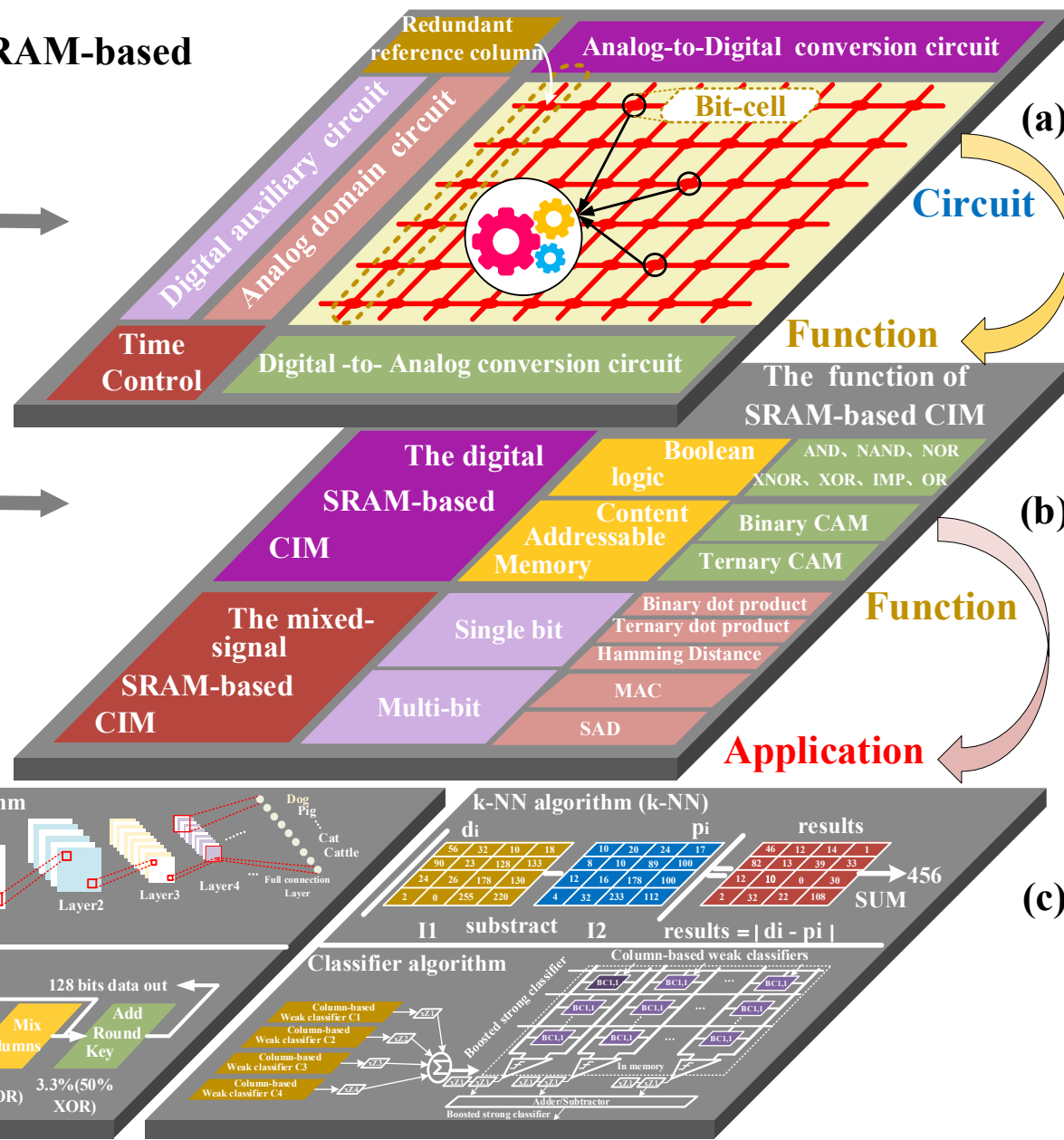
- ① Bitcell structures
- ② Peripheral circuits

Function level:

- ① Digital CIM
- ② Mixed-signal CIM

Application level:

- ① CNN
- ② AES
- ③ k-NN
- ④ Classifier

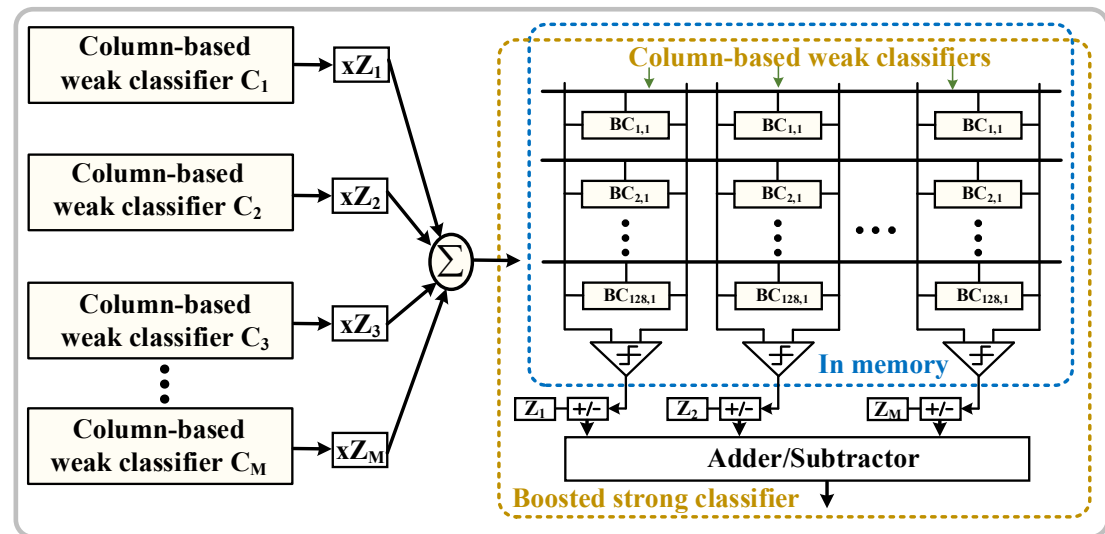
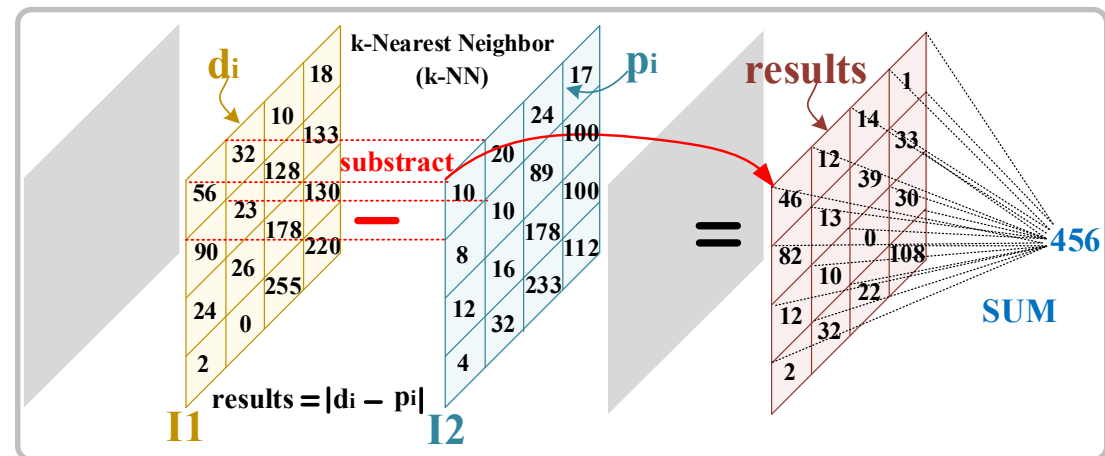
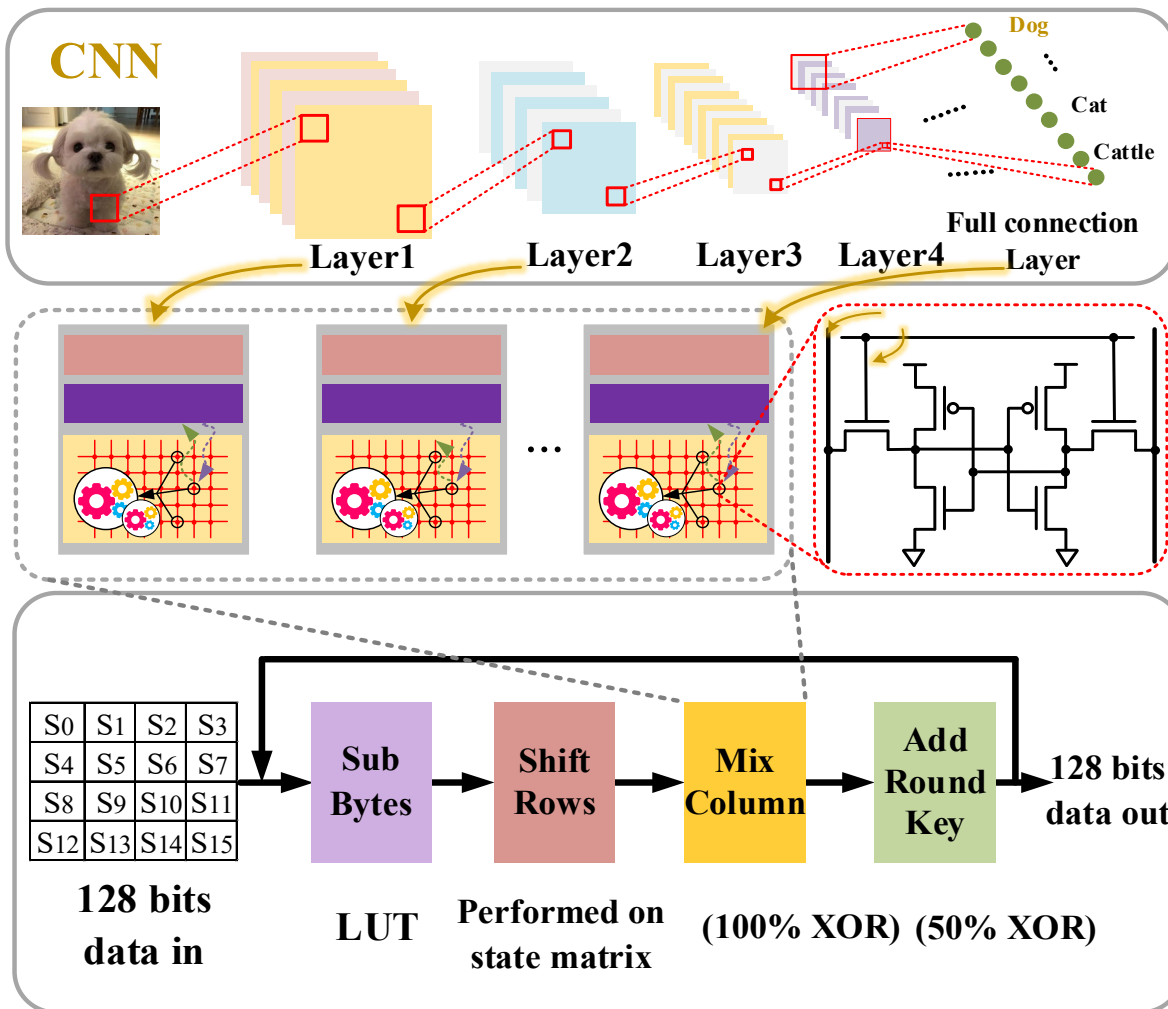


电路层：通过增加外围电路或者重构基本单元，一些复杂的运算可以在存内被实现。

功能层：目前基于SRAM的存内计算所能实现的功能主要分为纯数字运算和数模混合运算。

应用层：一系列高开销的算法可以基于SRAM实现加速。大大提高了数据的吞吐量与运算速度！

1 SRAM存算主要应用



分析

目前基于SRAM所能实现的存内计算操作主要应用于各种AI或加密算法，这些算法包括卷积神经网络算法、AES加密算法、模式匹配算法以及分类器算法。**大大提高了能量效率与计算速度！**

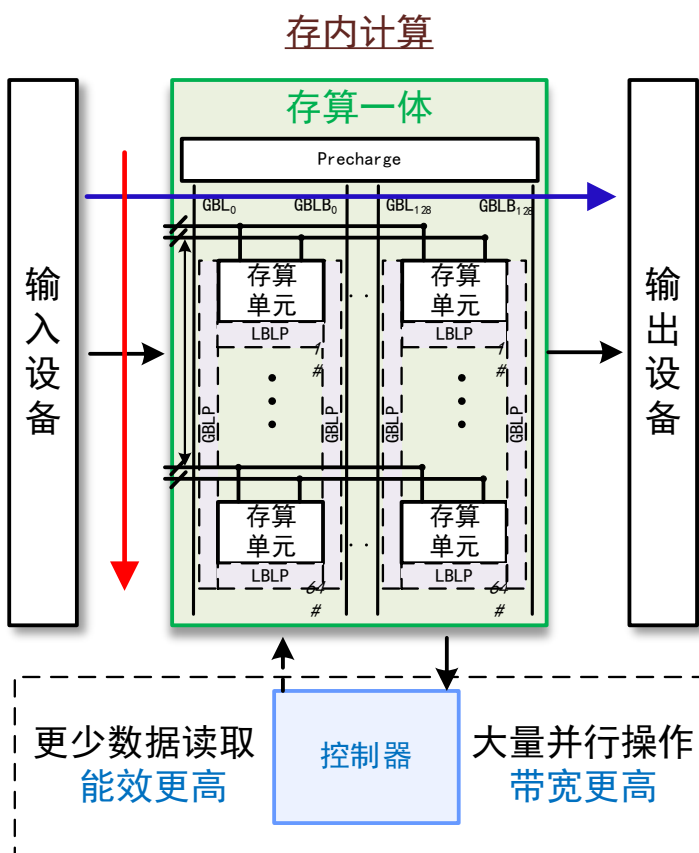


PART·TWO

面 临 的 挑 战

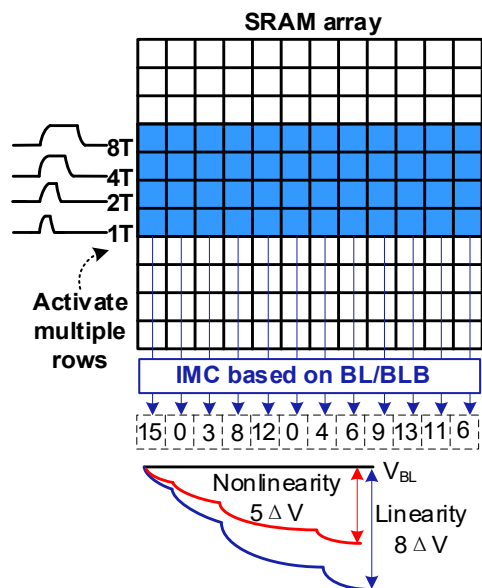
RESEARCH CHALLENGES OF IN-MEMORY COMPUTING BASED ON SRAM

➤ 可靠性问题：线性度、一致性、读破坏



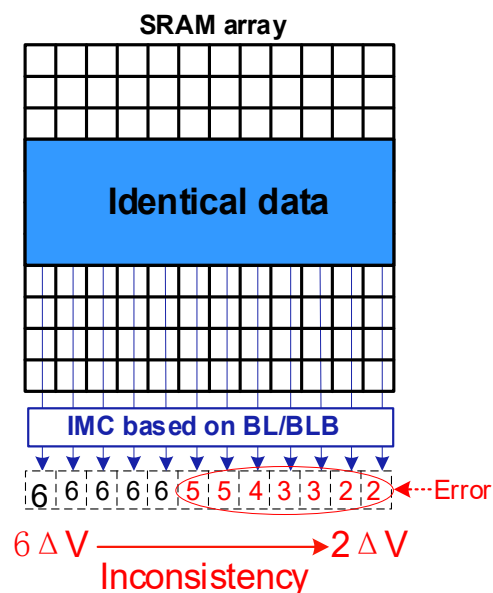
①：线性度

现有的研究成果较少，总体来说理论分析不够全面，没有从根源上解决问题。



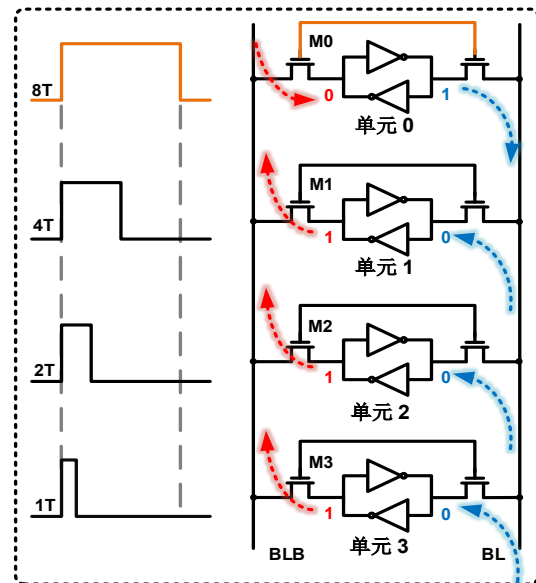
②：一致性

存内计算领域还没有研究人员提出相应的解决方案。



③：读破坏

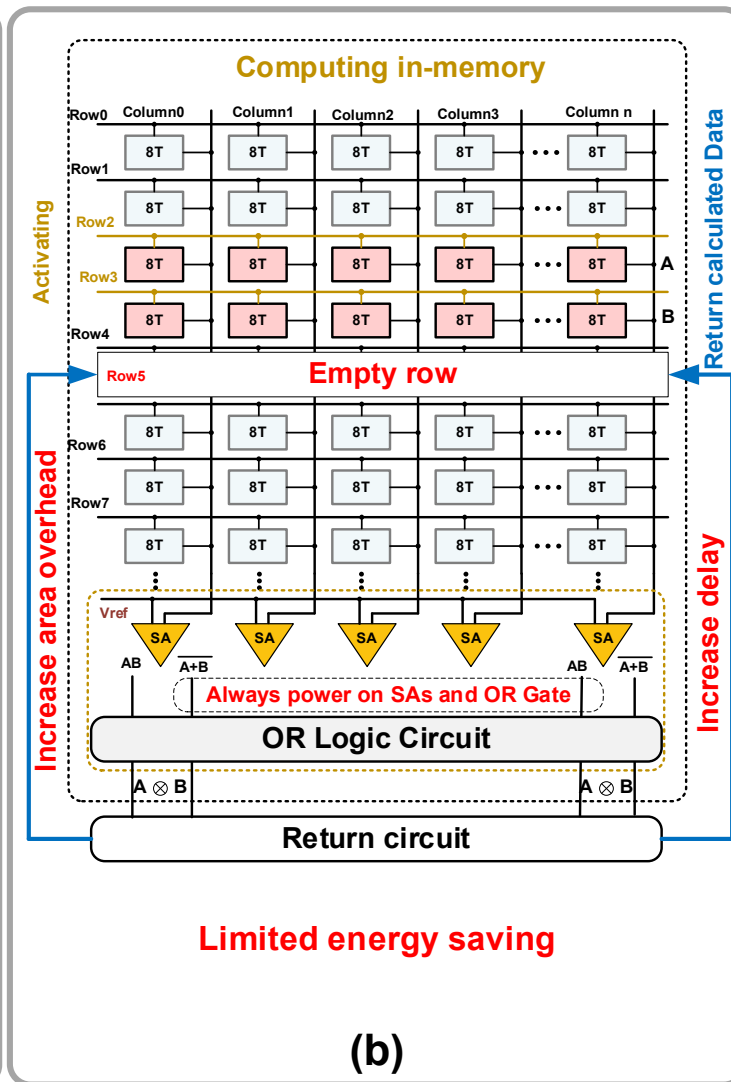
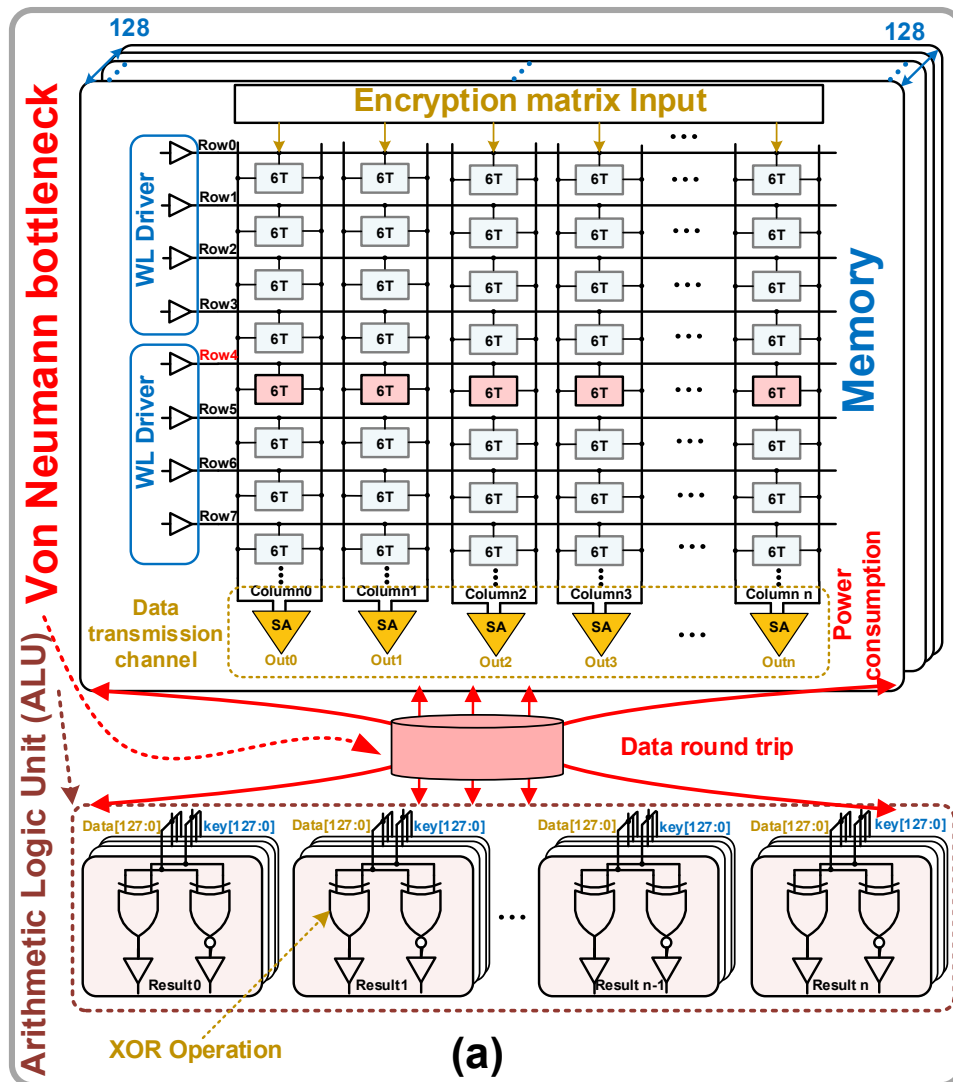
主要集中在使用读写分离的单元结构。



存内计算领域面临的共性问题！

基于SRAM存内计算的挑战

➤ 一体化优化问题：写入、读取、计算、量化以及回写



传统冯诺依曼架构中
计算与存储分离，无法
满足频繁访问的需求。

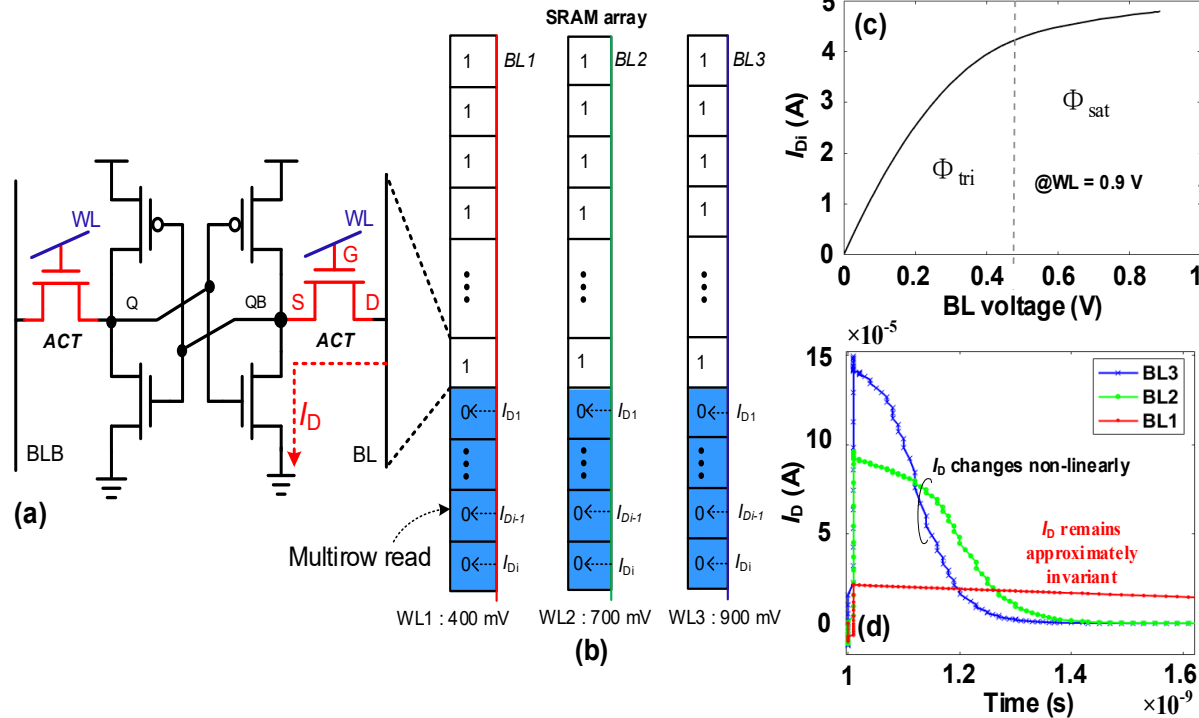
✓ 存内计算更高能效

- ✗ 存储方式的冲突
- ✗ 量化代价
- ✗ 写入时延

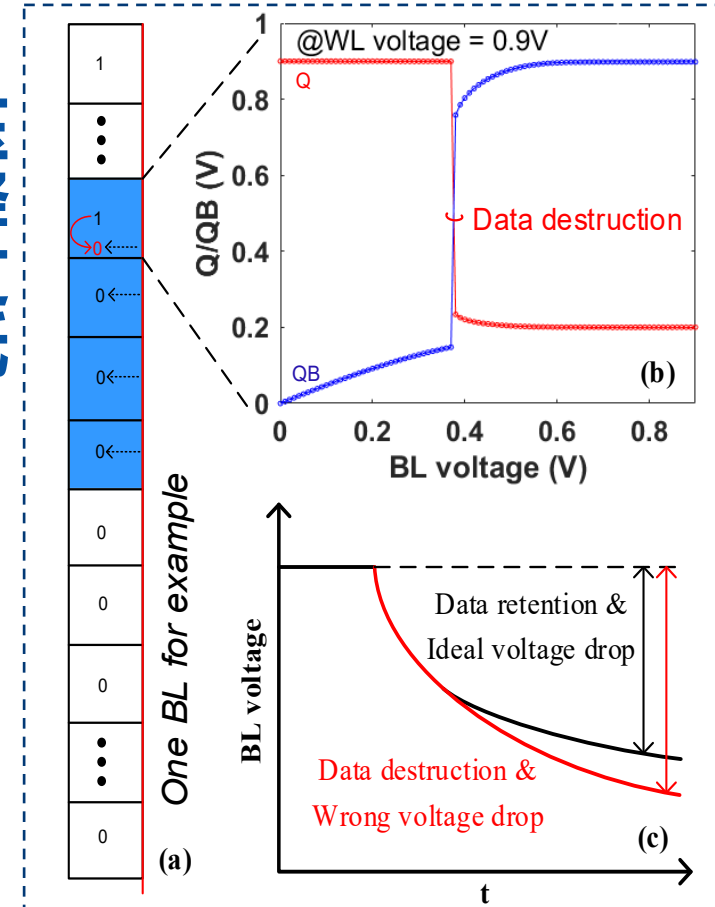


线性度分析

工作区间



压降过低



分析

饱和区:

$$I_{Di} \approx \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})^2 (1 + \lambda V_{DS})$$

三极管区:

$$I_{Di} \approx \mu_n C_{ox} \frac{W}{L} [(V_{GS} - V_{TH}) V_{DS} - \frac{1}{2} V_{DS}^2] (1 + \lambda V_{DS})$$

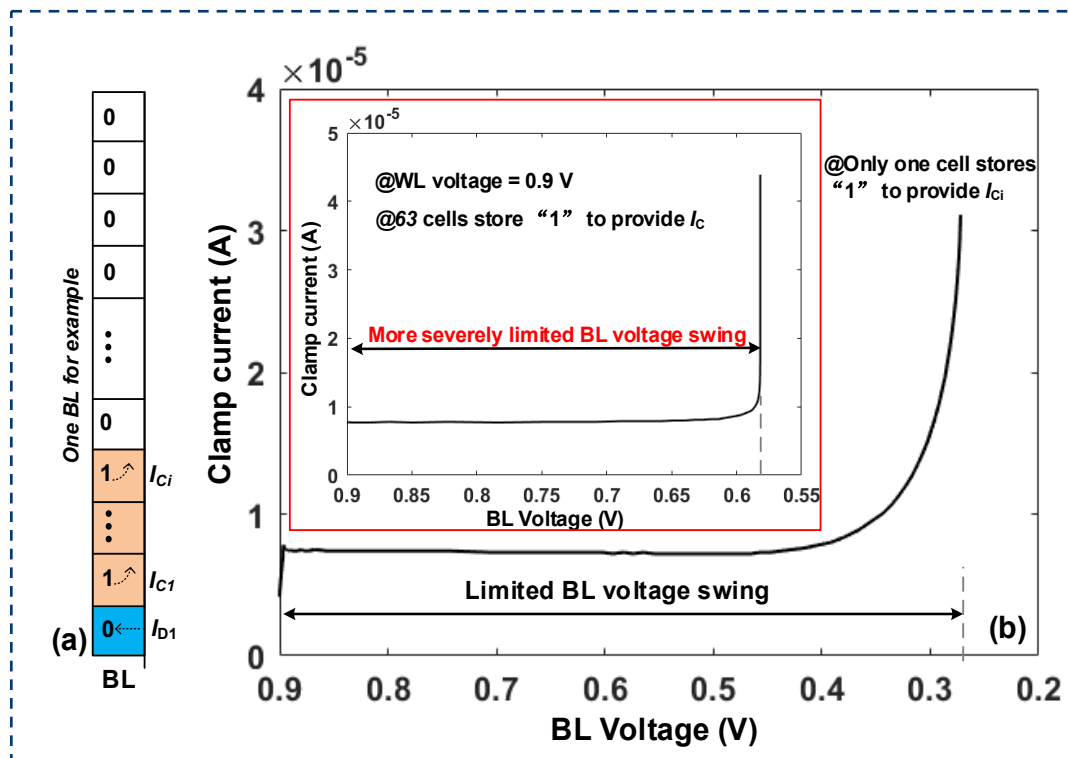
读取破坏:

$$I_{Rd} = n \beta$$

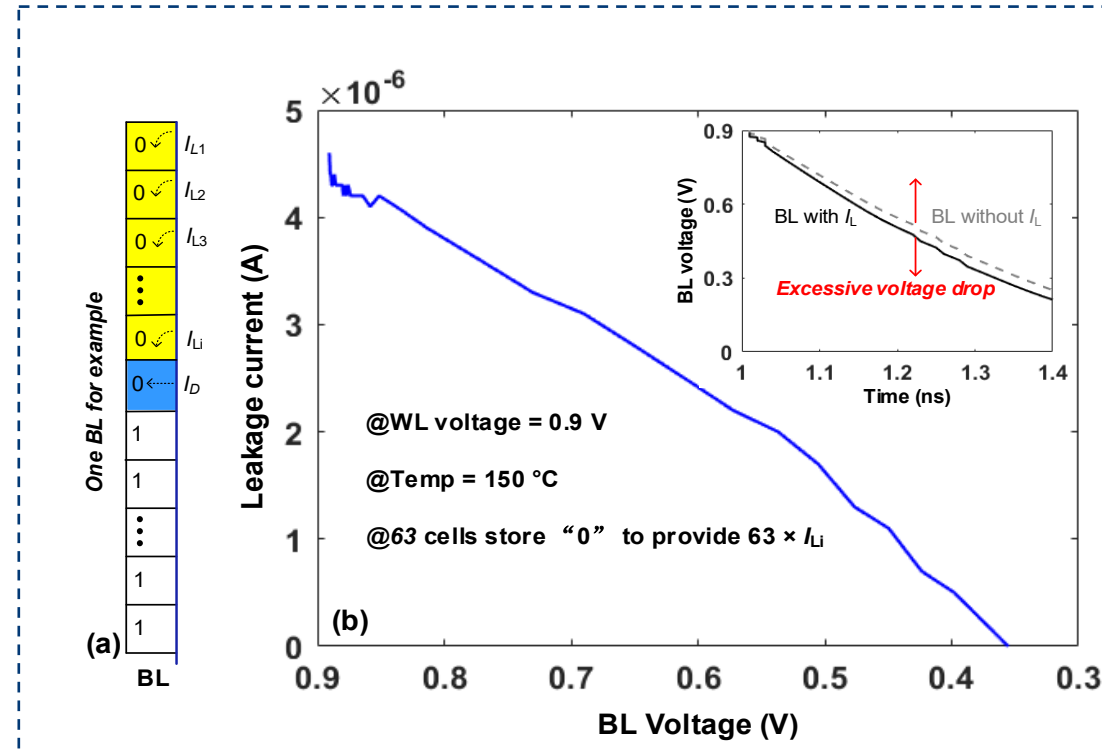


线性度分析

钳位电流



泄漏电流



分析

钳位电流:

$$I_C = f(m, V_{BL})$$

泄漏电流:

$$I_L = \sum_{i=1}^m I_{Li}$$

➤ 线性度综合分析

➤ 工作区间

饱和区:

$$I_{Di} \approx \frac{1}{2} \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})^2 (1 + \lambda V_{DS})$$

线性区:

$$I_{Di} \approx \mu_n C_{ox} \frac{W}{L} [(V_{GS} - V_{TH})V_{DS} - \frac{1}{2}V_{DS}^2] (1 + \lambda V_{DS})$$

➤ 钳位电流

$$I_C = f(m, V_{BL})$$

➤ 泄漏电流

$$I_L = \sum_{i=1}^m I_{Li}$$

➤ 读取破坏

$$I_{Rd} = n\beta$$

➤ 总公式 $\Delta I_{BL} = \Delta I_D + \Delta I_L + \Delta I_C + \Delta I_{Rd}$

$$\Delta V = T \frac{\Delta I_{BL}}{C_{BL}}$$

较低的位线电压

较高的位线电压

ΔI_D 和 ΔI_L 不恒定

ΔI_D 和 ΔI_L 恒定

I_C 不可忽略

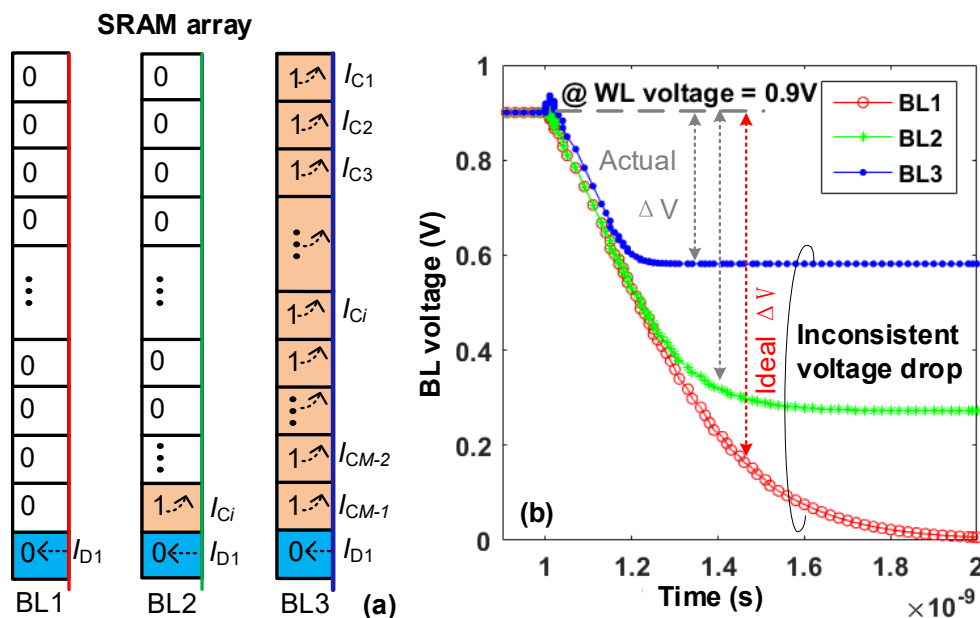
I_C 可忽略

易发生读破坏

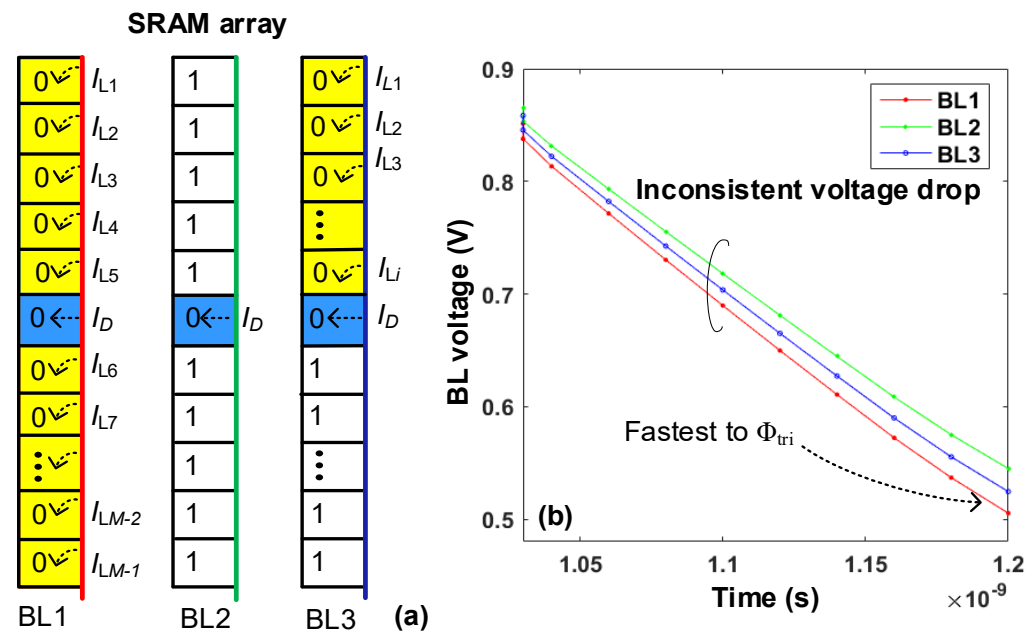
无读破坏

一致性分析

钳位电流



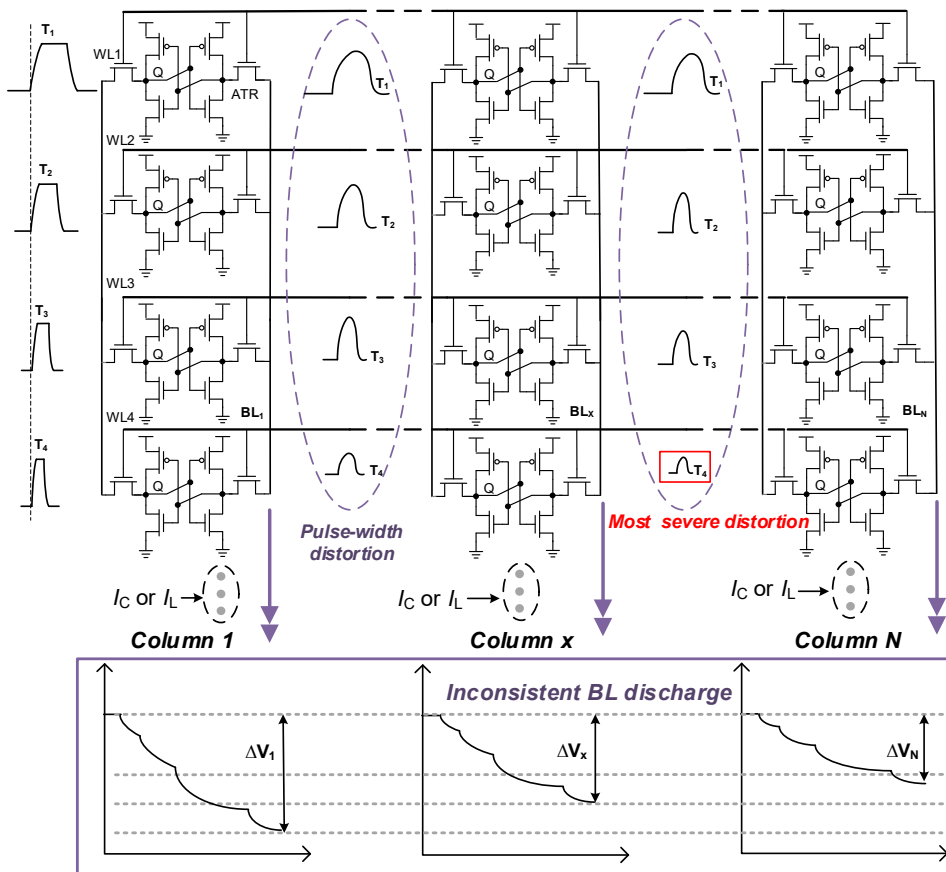
泄漏电流



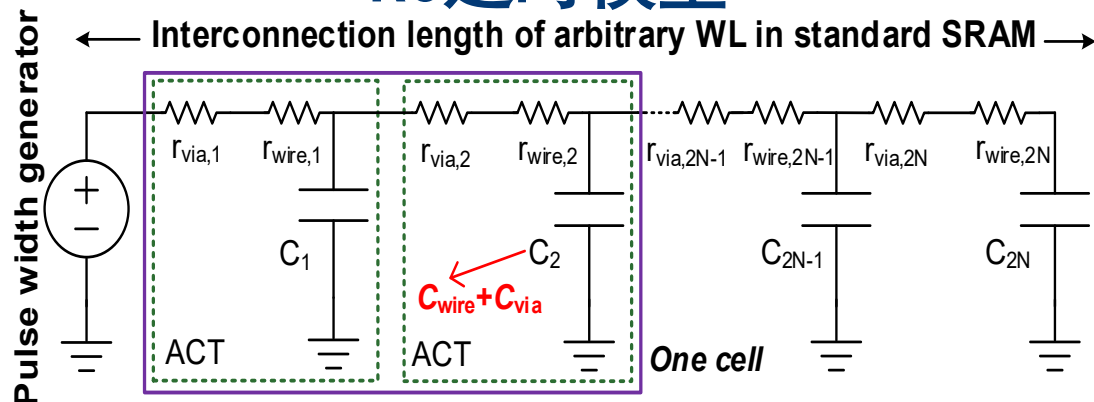
分析

在**钳位电流**或者**泄漏电流**的影响下，SRAM中任意两列都可能具有**不同的位线电压变化，具有不一致性。**

一致性分析



RC延时模型



$$\tau_{2N} = \sum_{i=1}^{2N} C_i \sum_{j=1}^i (r_{\text{wire},j} + r_{\text{via},j})$$

$$\tau_2 = (r_{\text{wire},1} + r_{\text{via},1})C_1 + [r_{\text{wire},1} + r_{\text{via},1} + r_{\text{wire},2} + r_{\text{via},2}]C_2$$

$$\tau_{128} = \sum_{i=1}^{128} C_i \sum_{j=1}^i (r_{\text{wire},j} + r_{\text{via},j})$$

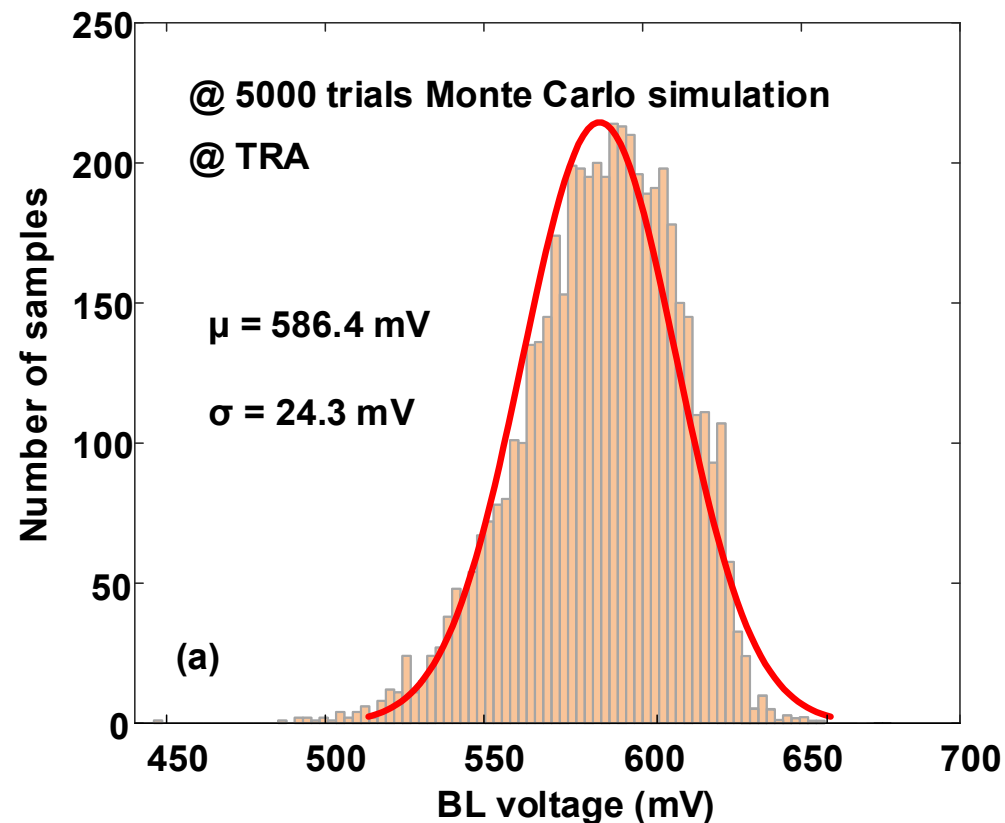
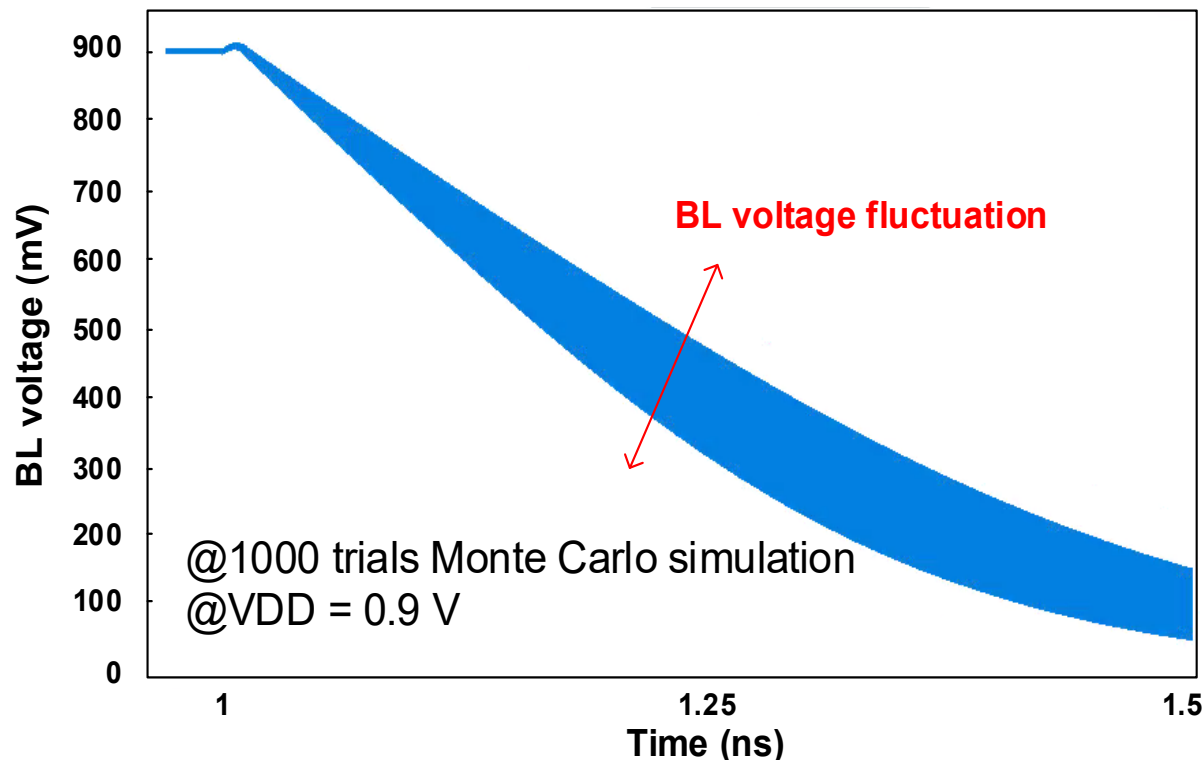
以上是四行读取示例及RC模型。以64*64的阵列为例，第一列与第64列的延时分别等于 2τ

和 128τ ，从这两个公式可以看到，两者之间有很大的差异。



一致性分析

阈值电压失配



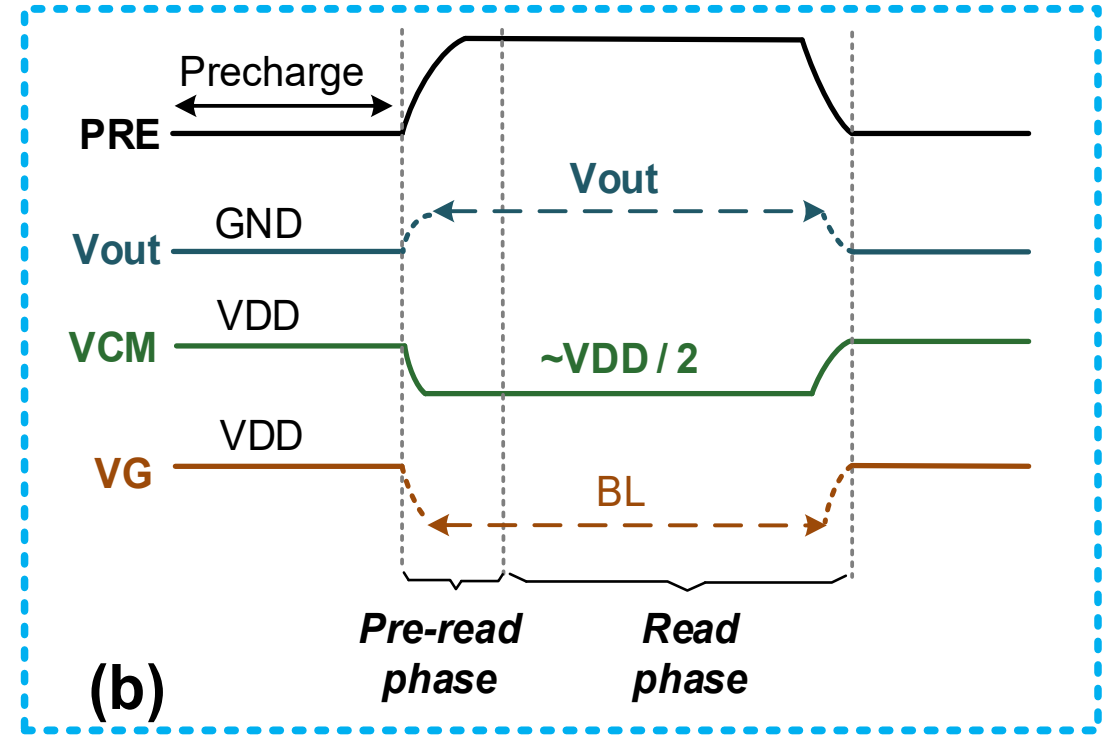
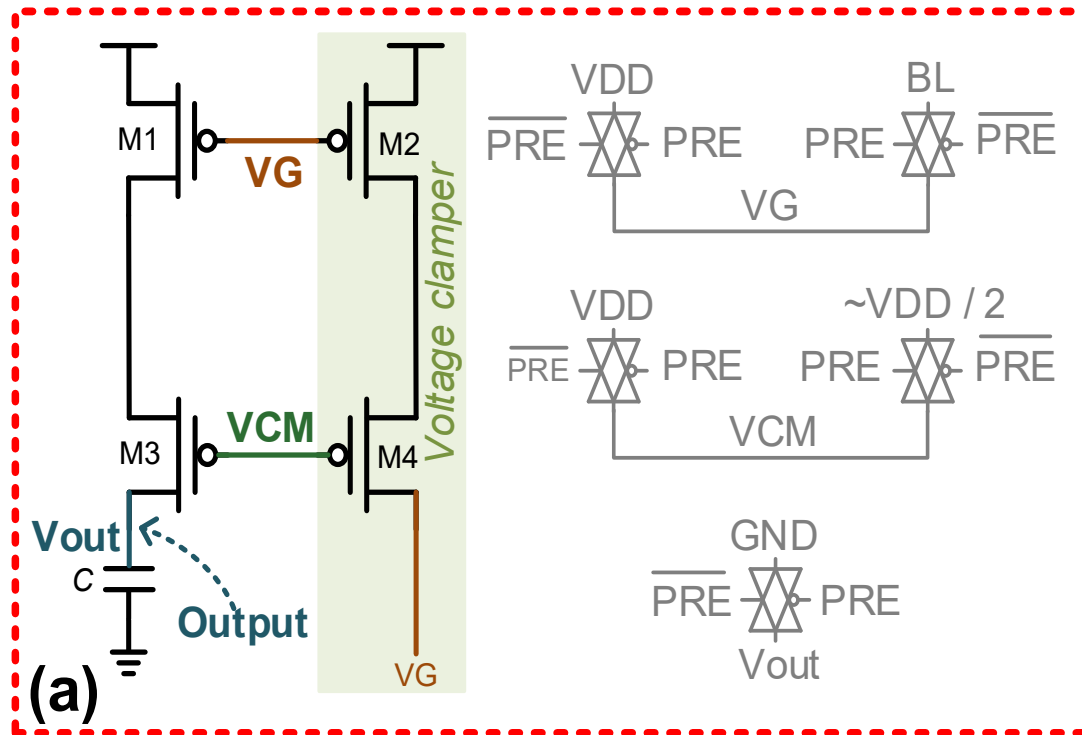
分析

蒙特卡洛仿真表明在失配的情况下，放电过程中位线电压产生**随机波动**，蒙特卡洛仿真表明，传统存内计算电路中位线放电的标准差可达几十mv。

➤ 电路设计

共源共栅电流镜电路

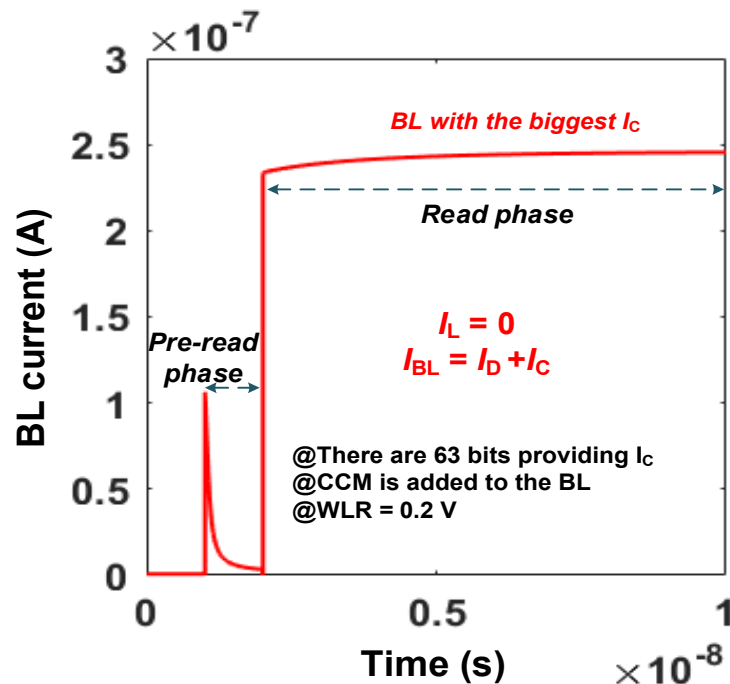
同时解决存算线性度和一致性的方案!



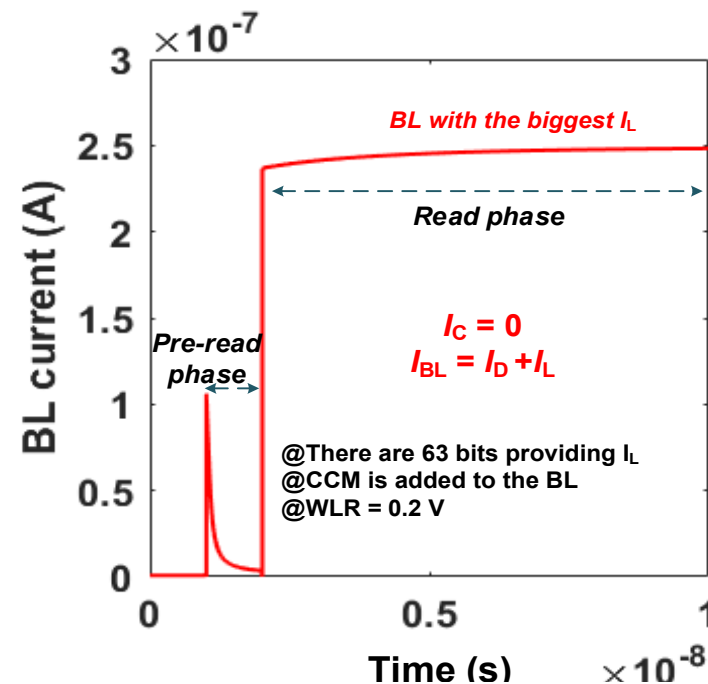


电路设计

➤ 具有稳定的钳位电流



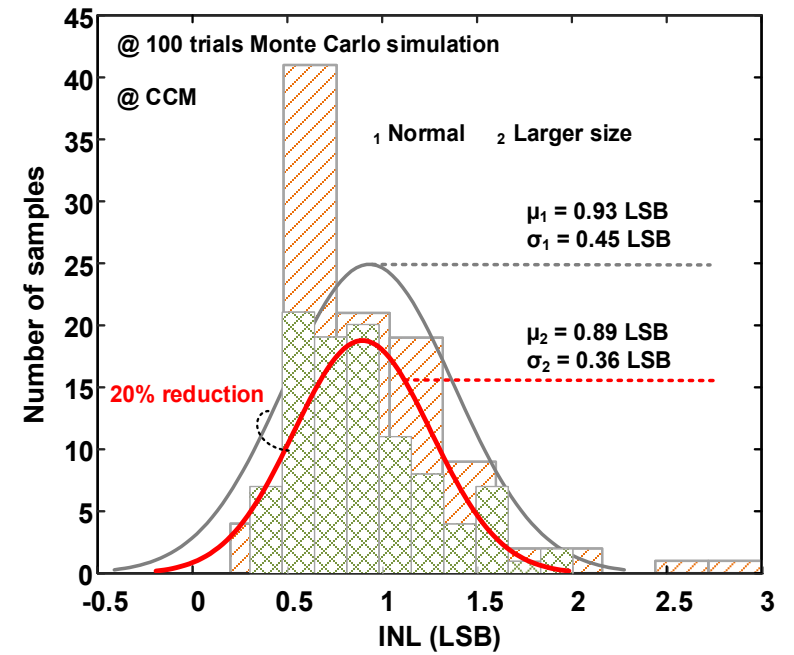
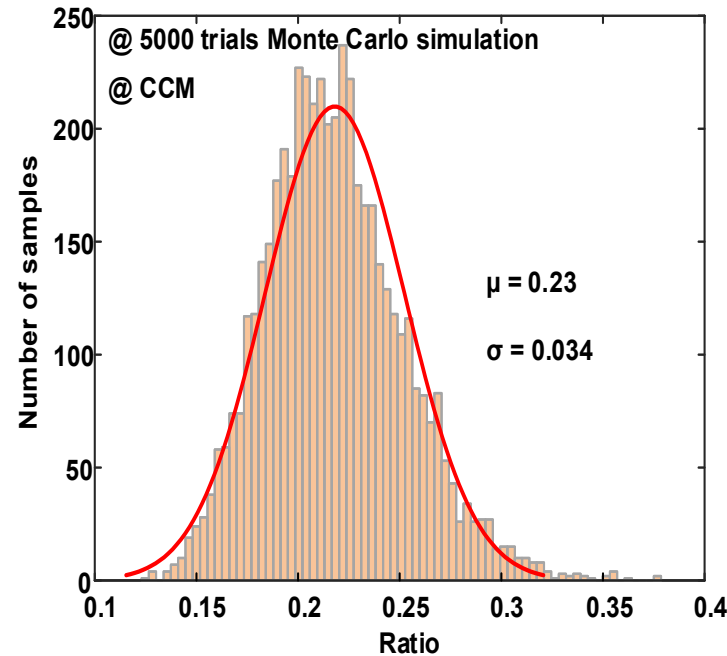
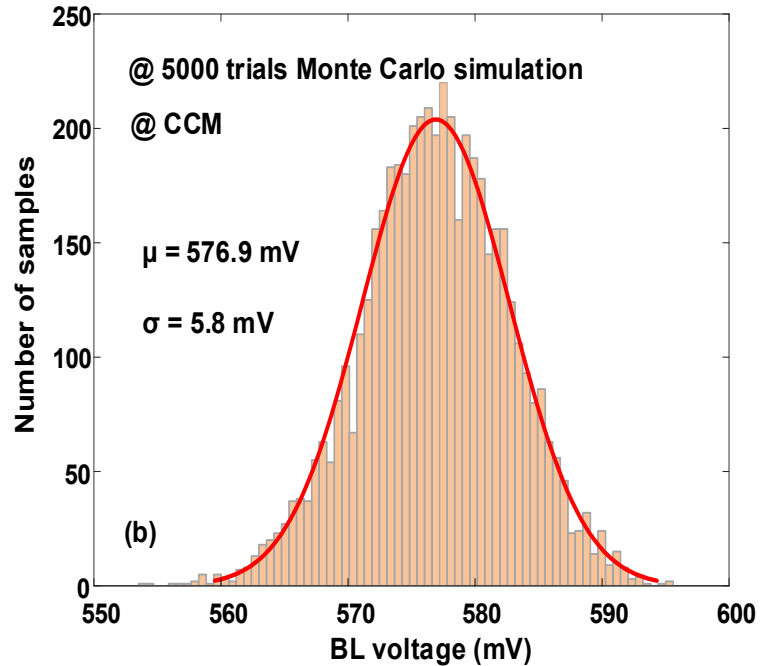
➤ 具有稳定的泄露电流



分析

两种情况下的位线总电流在计算时几乎保持不变，因此根据之前推导的总公式，可以得到线性的 ΔV ，有效地提高计算的线性度。

蒙特卡洛仿真



分析

电流镜将位线电压波动
降低了76.1%

电流镜复制电流的标
准差为0.034

线性度的波动降低
了20%

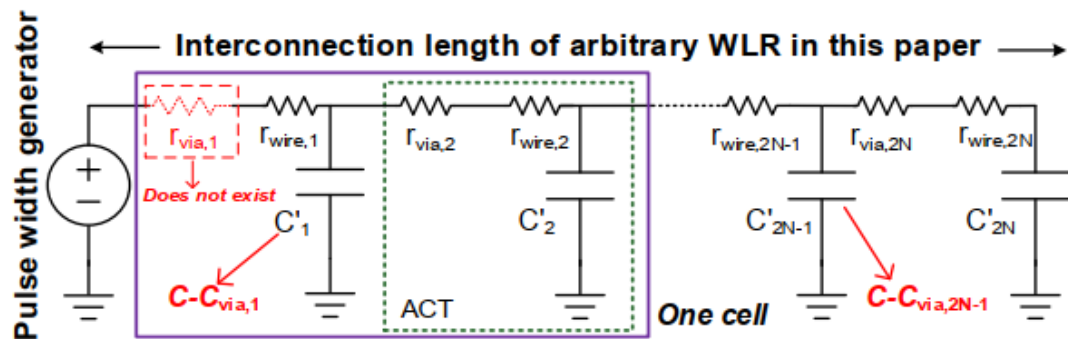
三种技术提高一致性

P1: 增强字线有效信号

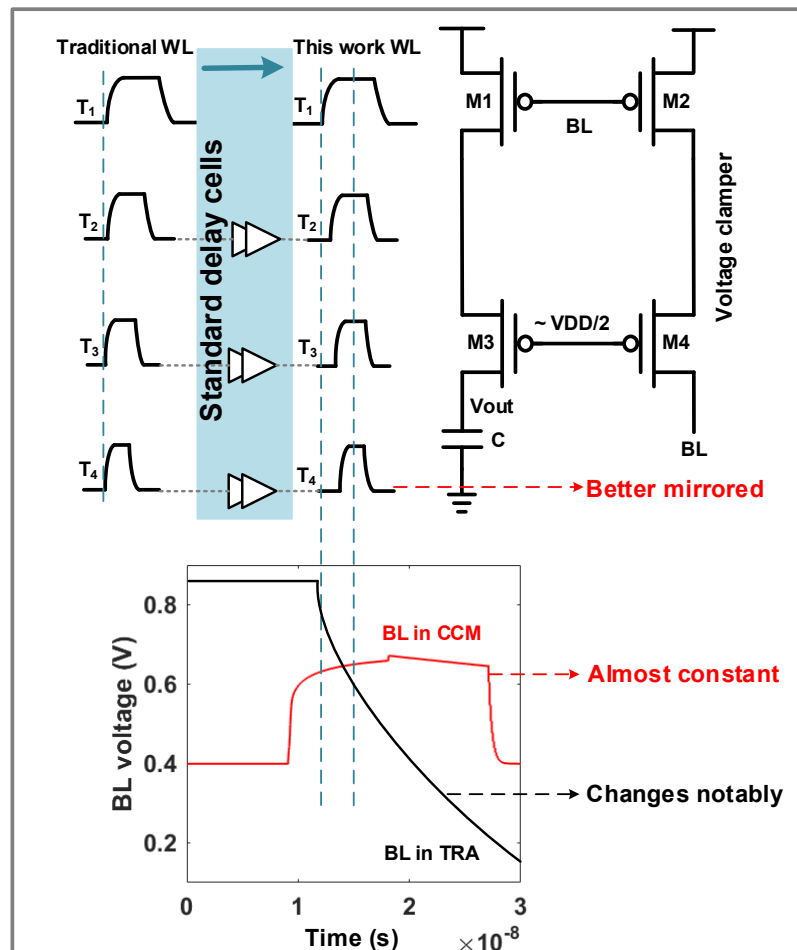
得益于电流镜的钳位作用，扩大字线脉宽，降低上升沿下降沿在整体信号中的占比。

P2: 双字线结构降低延迟

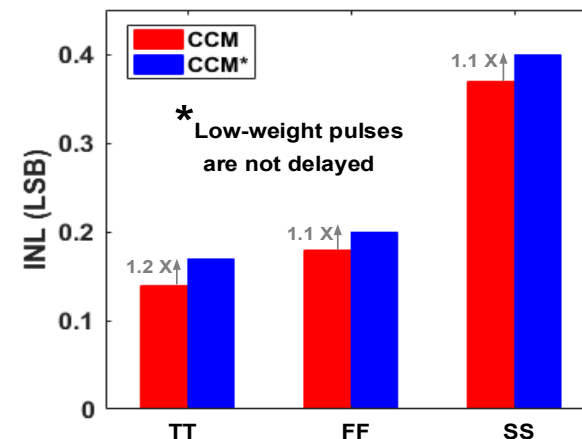
字线上的过孔电容和过孔电阻减半。



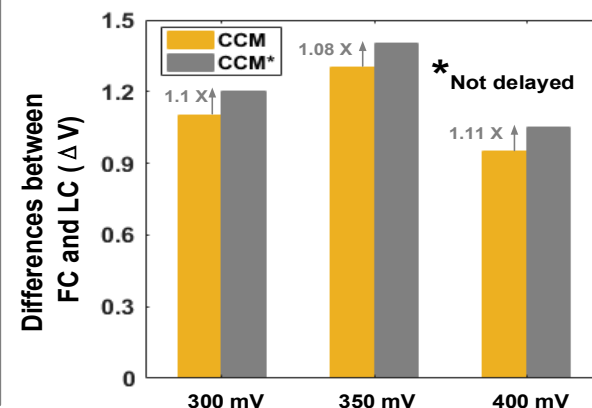
P3: 低权重脉冲延时开启



不同工艺角的后仿真

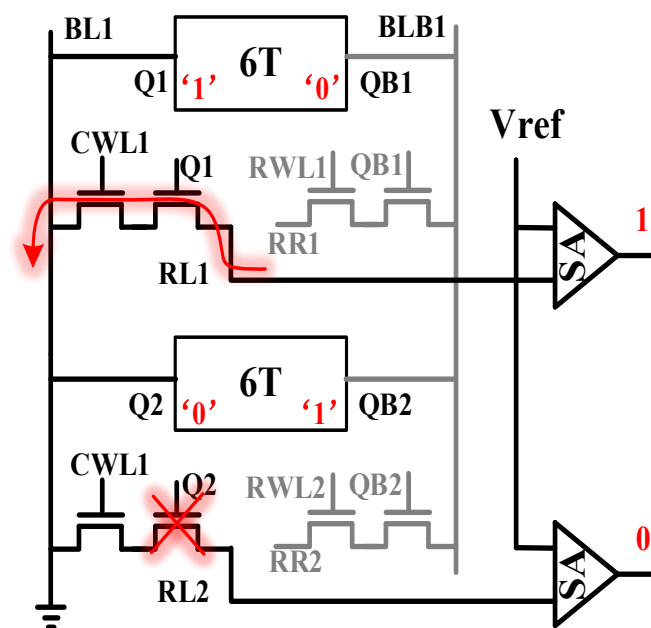
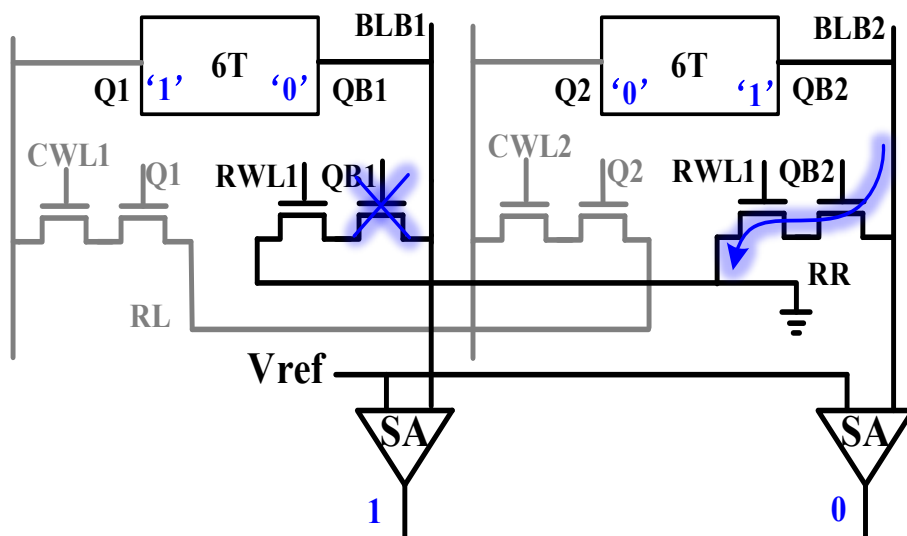


不同字线电压的后仿真



➤ 存储方式冲突

💡 常规SRAM模式下，除了传统的数据按行读取（行向读操作）以外，还兼容数据按列读取，即列向读操作



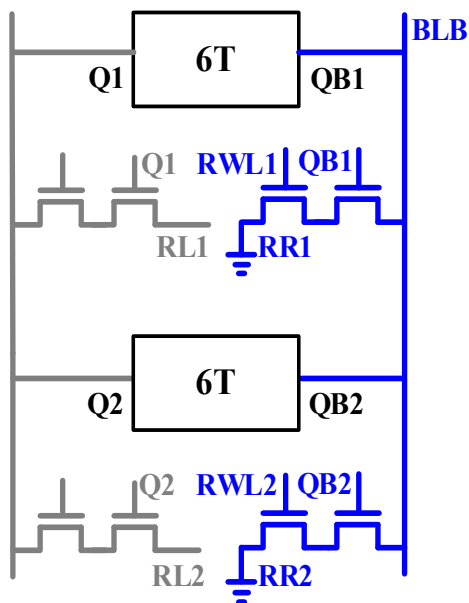
行向读 “1”、读 “0”

均为单端

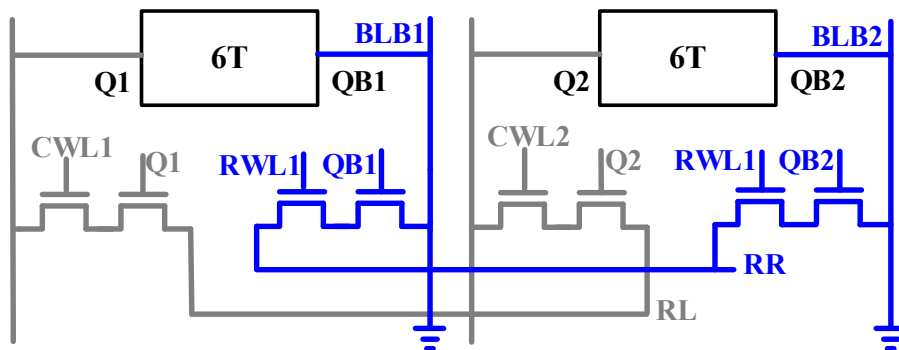
列向读 “1”、读 “0”

➤ 存储方式冲突

行向与运算



列向与运算



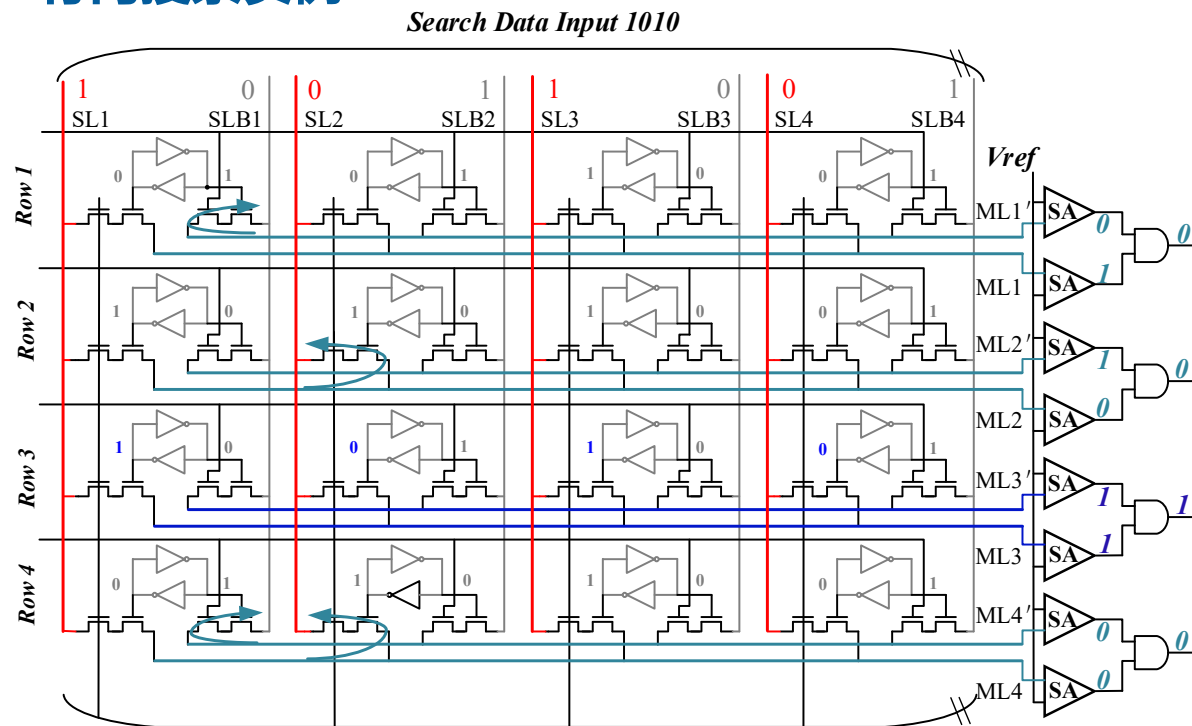
逻辑真值表

Q1	Q2	BLB	RR	AND
0	0	↓	↓	0
0	1	↓	↓	0
1	0	↓	↓	0
1	1	—	—	1

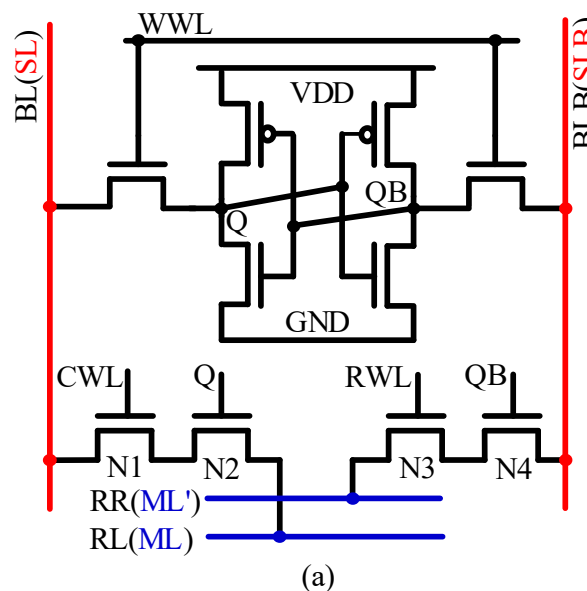
💡 右侧读端口 (BLB-RWL-RR) 配合水平字线RWL实现双向逻辑与运算，字线RWL负责激活参与运算的操作数，最终逻辑运算结果直接通过位线末端SA放大为标准逻辑电平0/1

➤ 存储方式冲突

行向搜索实例



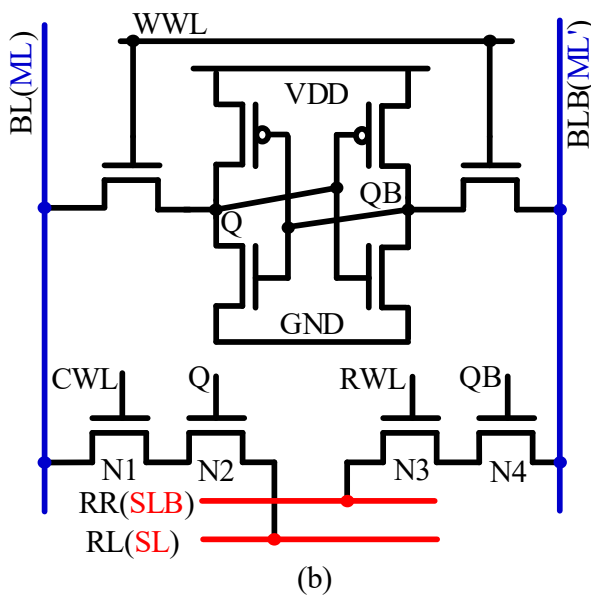
BCAM单元配置



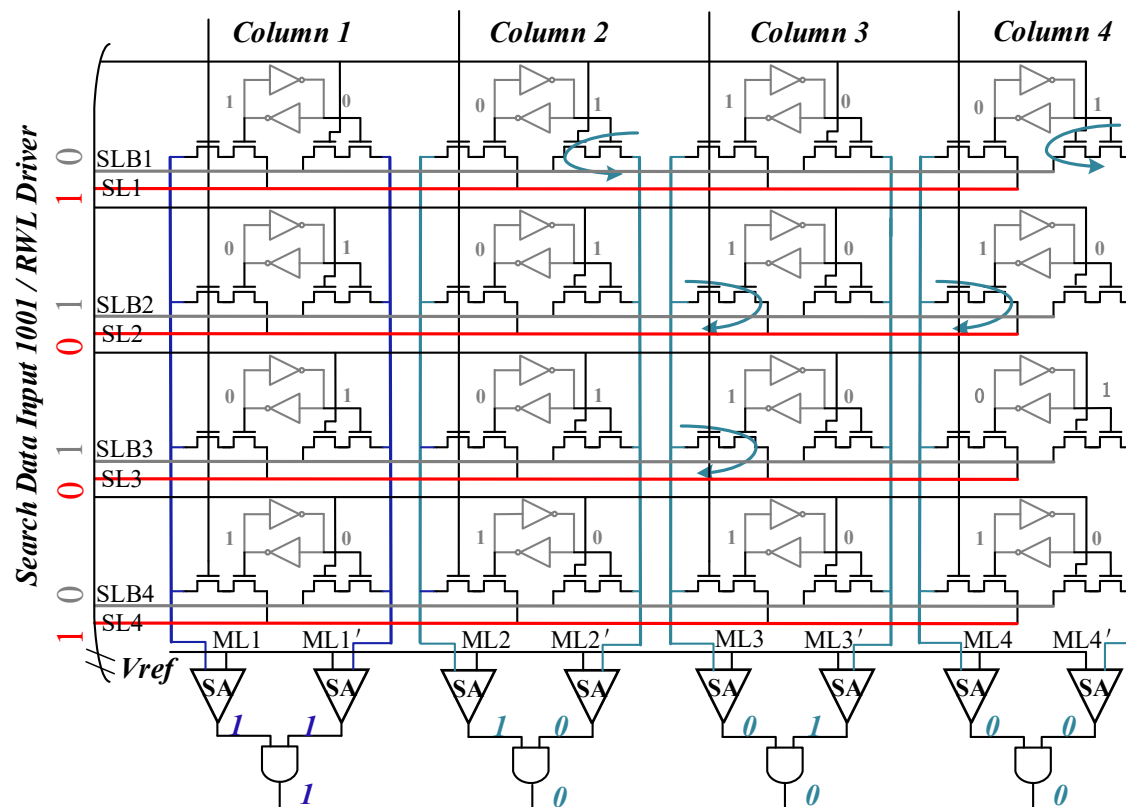
💡 内容可寻址存储器概念早已提出，除了数据存储外还具备数据并行搜索能力，是一类特殊的存内计算应用，根据搜索精度划分，可以分为BCAM和TCAM

➤ 存储方式冲突

BCAM单元配置



列向搜索实例



💡 区别于行向搜索，位线BL/BLB和RL/RR的配置互换，即BL/BLB作为匹配线，而RL/RR则作为搜索数据线



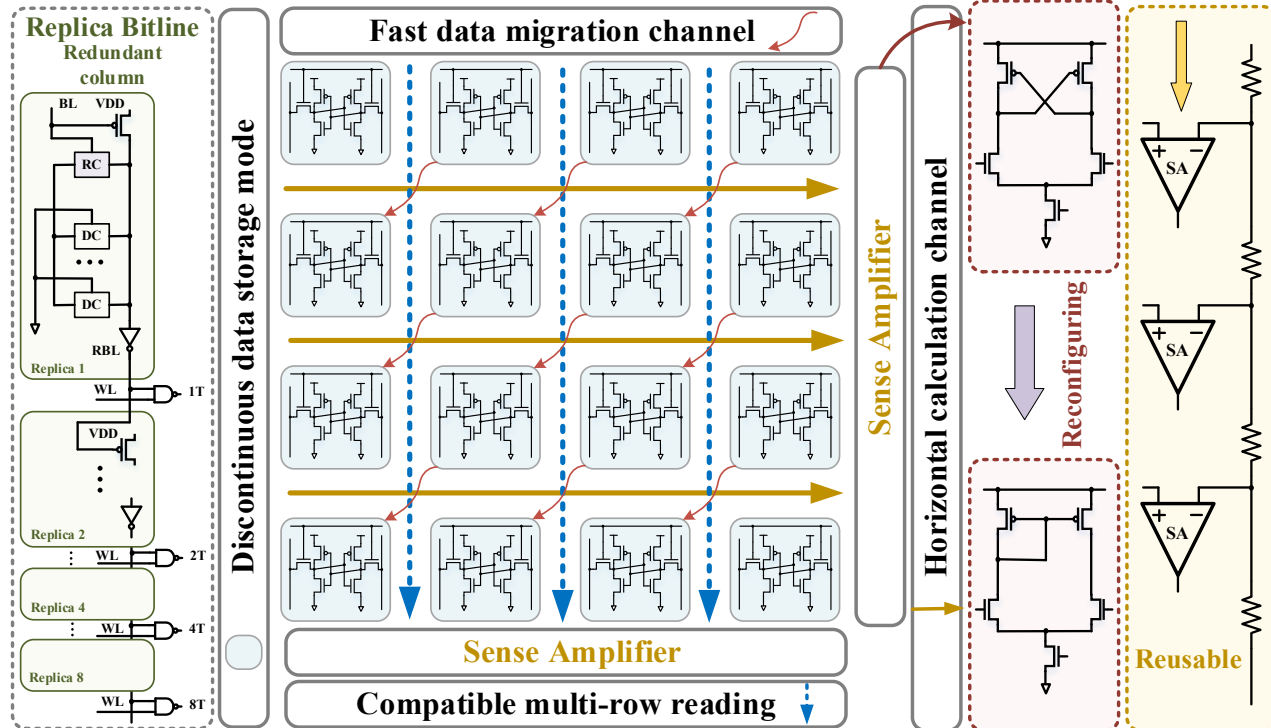
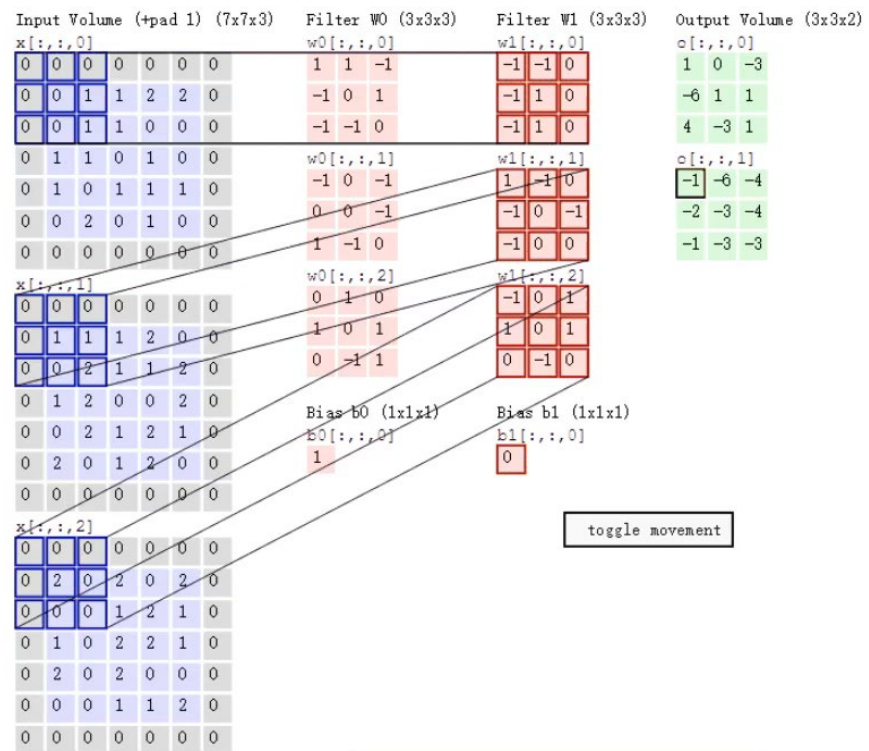
PART· THREE

基于SRAM的存内计算研究展望

RESEARCH PROSPECT OF IN-MEMORY COMPUTING BASED ON SRAM

3 SRAM存内计算研究展望

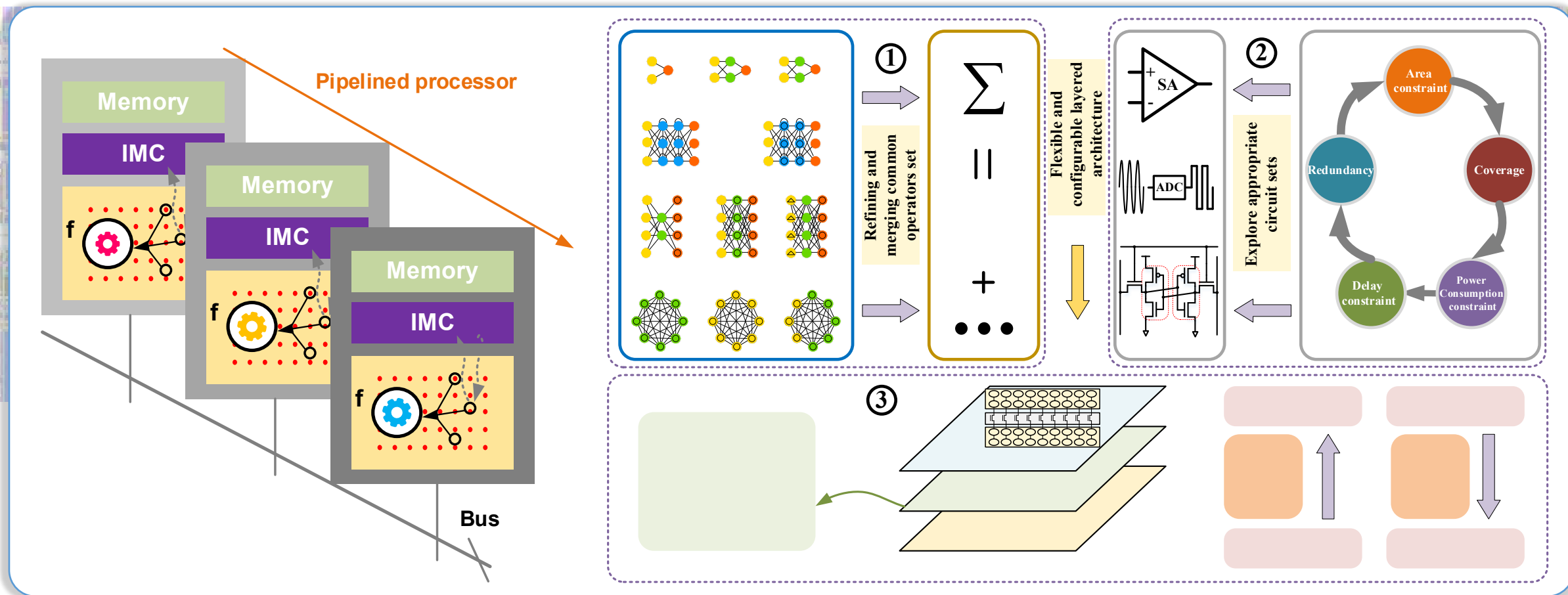
1. 研究数据存储模式与数据迁移电路



分析

存内计算非常适合处理神经网络，目前存内计算在神经网络中的研究非常广泛，但存在局限性：数据读写更新的能耗与所需时间远大于卷积的能耗与时间，增加功能，面积代价也增加

2. 新存算架构



分析定位不同Memory块的特点

多个Memory块流水线式运算

提炼归并共性算法集合

多层共享架构

谢谢各位聆听！