



# 南湖架构后端流水线的设计与改进

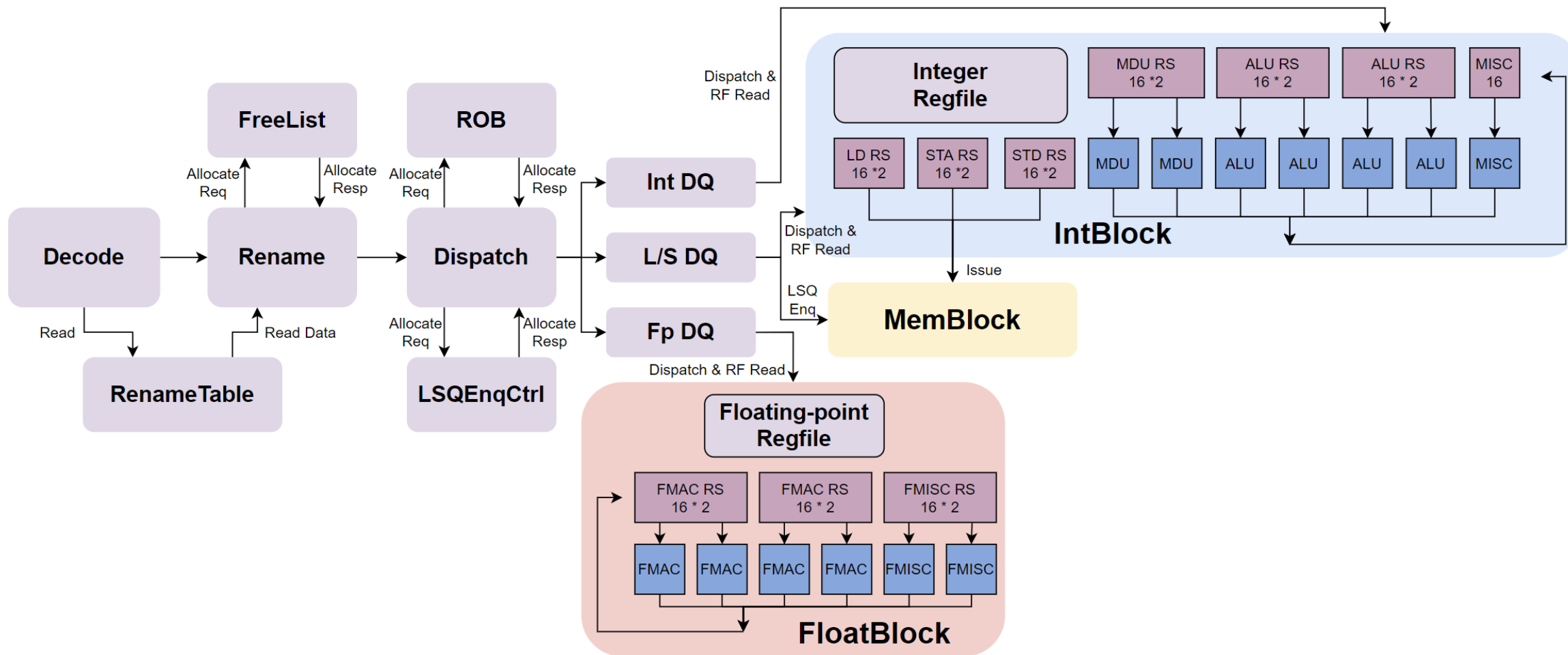
---

张紫飞、徐易难、蔺嘉炜、王华强

中科院计算所

2022年8月25日

# 后端流水线





# 后端流水线改进

CtrlBlock	雁栖湖架构	南湖架构	变化
Decode/Rename/ Dispatch/Commit	6	6	-
Instruction Fusion	NO	YES	新增
Move Elimination	NO	YES	新增
ROB Size	192	256	+ 64
ExuBlock	雁栖湖架构	南湖架构	变化
Integer RegFile Size	160	192	+ 32
Float RegFile Size	160	192	+ 32
Reservation Station Size	16 ; 1i1o	32 ; 2i2o	半分布式
ALU + MDU	4 + 2	4 + 2	-
FMAC + FMISC	4 + 2	4 + 2	-
B-extension	NO	YES	新增
K-extension	NO	YES	新增
Debug-extension	NO	YES	新增



# Instruction Fusion

## • 执行

- 两个指令融合为一个 Micro Op
- 复用原有计算类型
- 新计算类型
- 立即数快速传递

## • 目的

- 减轻后端流水线执行压力
- 消除数据前递延迟

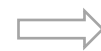
```
slli r1, r0, 32  
srli r1, r1, 32
```



```
add.uw r1, r0, zero
```

复用：得到64位数据的低32位

```
srli r1, r0, 8  
andi r1, r1, 255
```



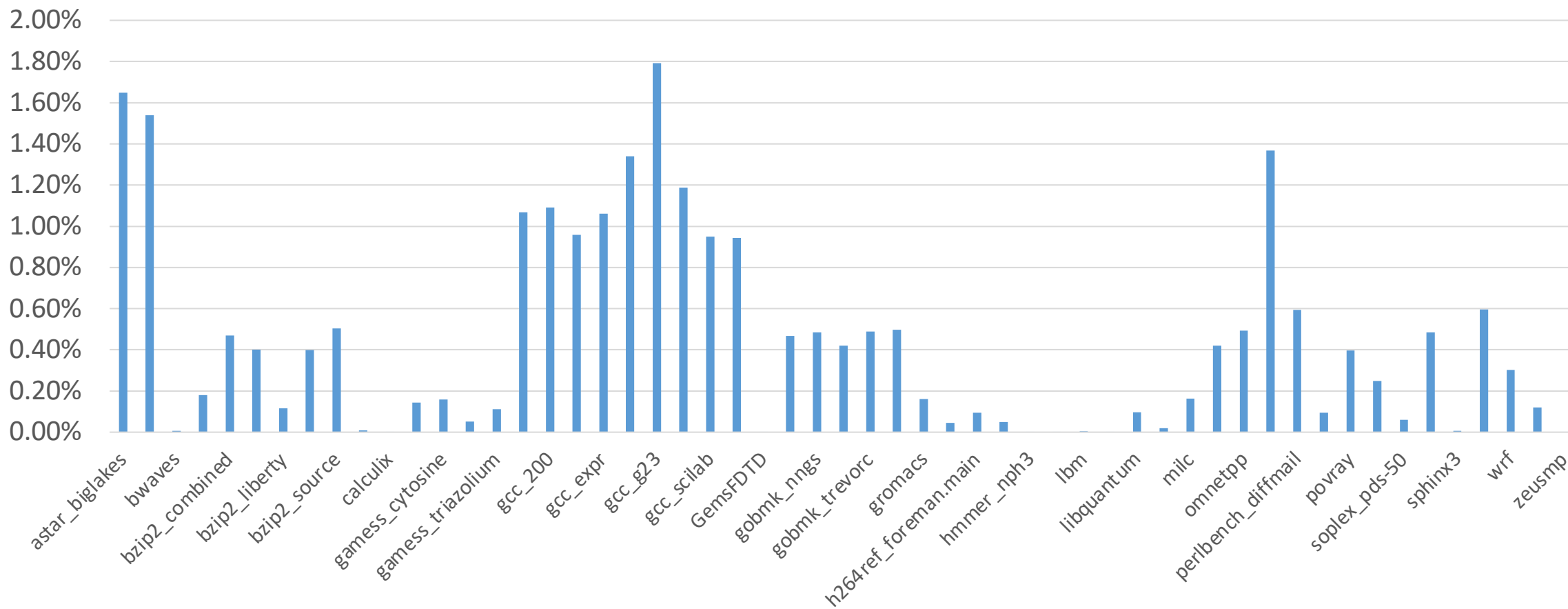
```
byte2 r1, r0
```

新计算类型：得到 64 位数据的第二个 8 比特



# Instruction Fusion

## SPEC2006 指令融合 Uops 减少百分比



# Rename & Move Elimination

- 维护逻辑寄存器和物理寄存器的映射
- Move Elimination
  - Move 指令进入 ROB 时标记为已完成，无需发射执行
  - 通过修改重命名表的映射关系实现
  - RefCounter 记录物理寄存器被映射的次数
  - RefCounter 为0时，释放物理寄存器

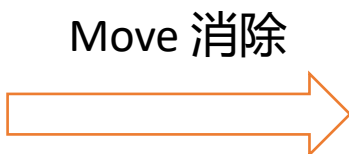


# Move Elimination

- 例子

程序流：

```
0: add x2, x1, x1
1: mv x3, x2
2: add x5, x3, x4
```



```
0: add x2, x1, x1
1: mv x3, x2
2: add x5, x3, x2
```

重命名表：

x0	0
x1	1
x2	2
x3	3
...	...

0: 初始值



x0	0
x1	1
x2	32
x3	3
...	...

1: add



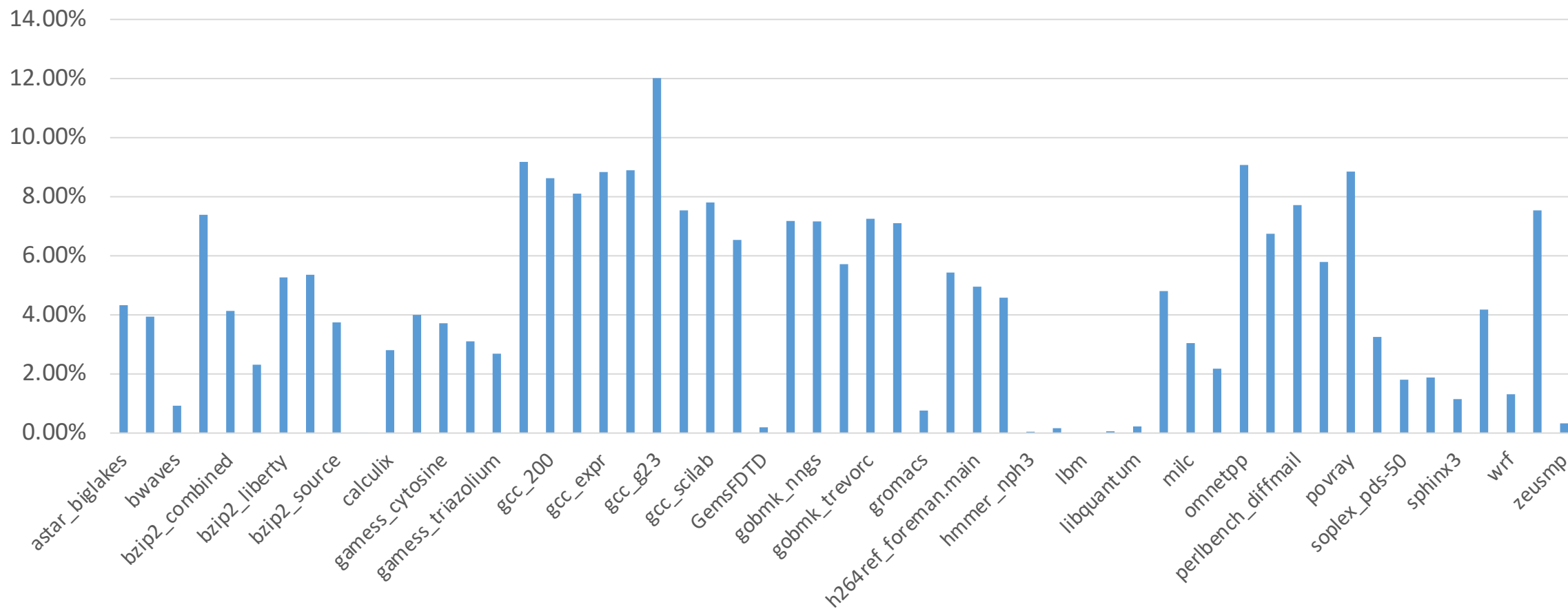
x0	0
x1	1
x2	32
x3	32
...	...

2: mv



# Move Elimination

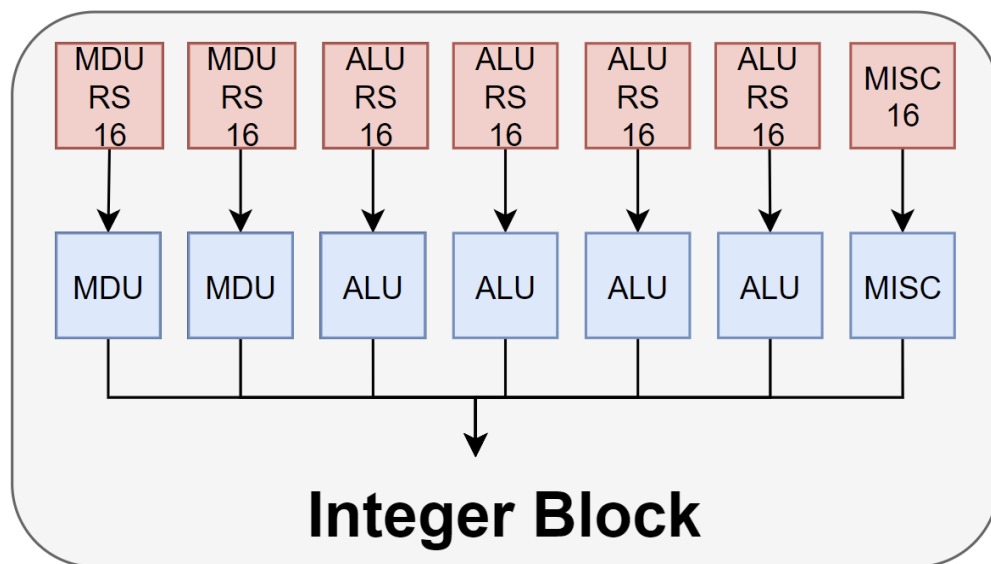
## SPEC206 中 Move Elimination的指令百分比



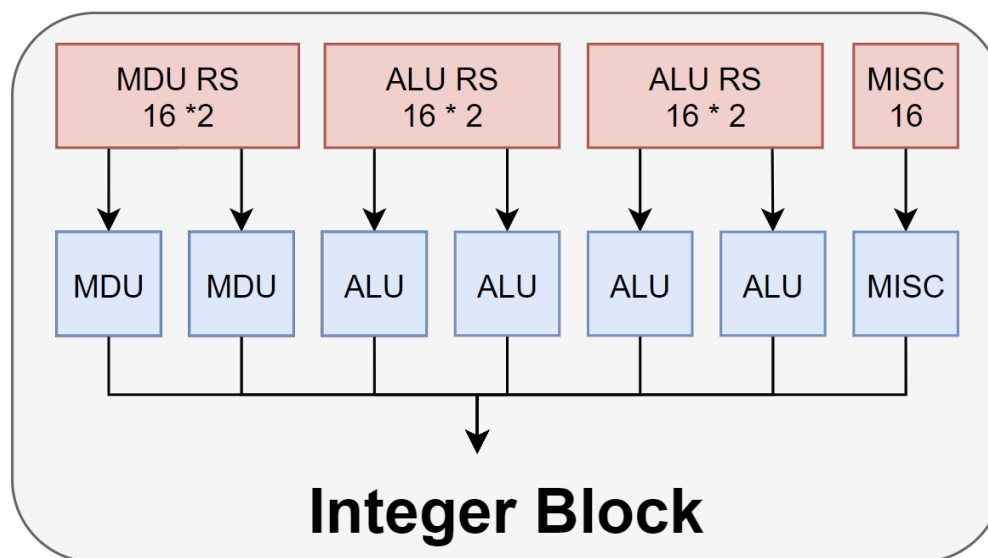


# Reservation Station

- 组织关系
  - 2进2出
  - 一个保留站均衡两个执行单元



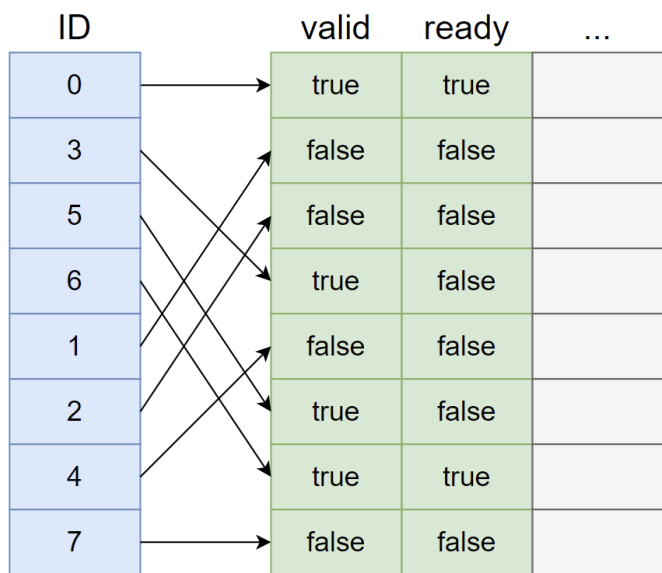
雁栖湖架构保留站组织形式



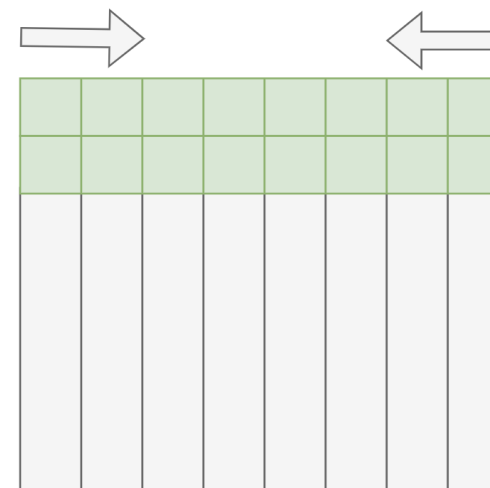
南湖架构保留站组织形式

# Reservation Station: Select Policy

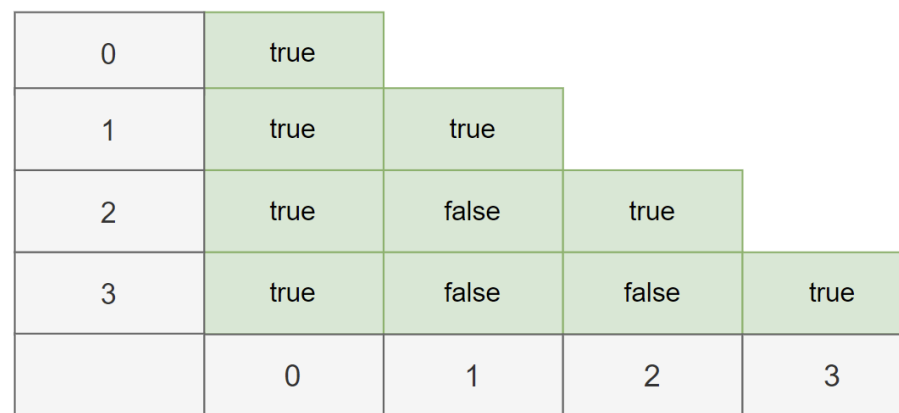
- Normal Select
  - 根据位置信息选择多个
- Age Select
  - 利用 Age Matrix 选择 1 个



雁栖湖架构 IndexQueue Age 算法



南湖架构 Normal Select

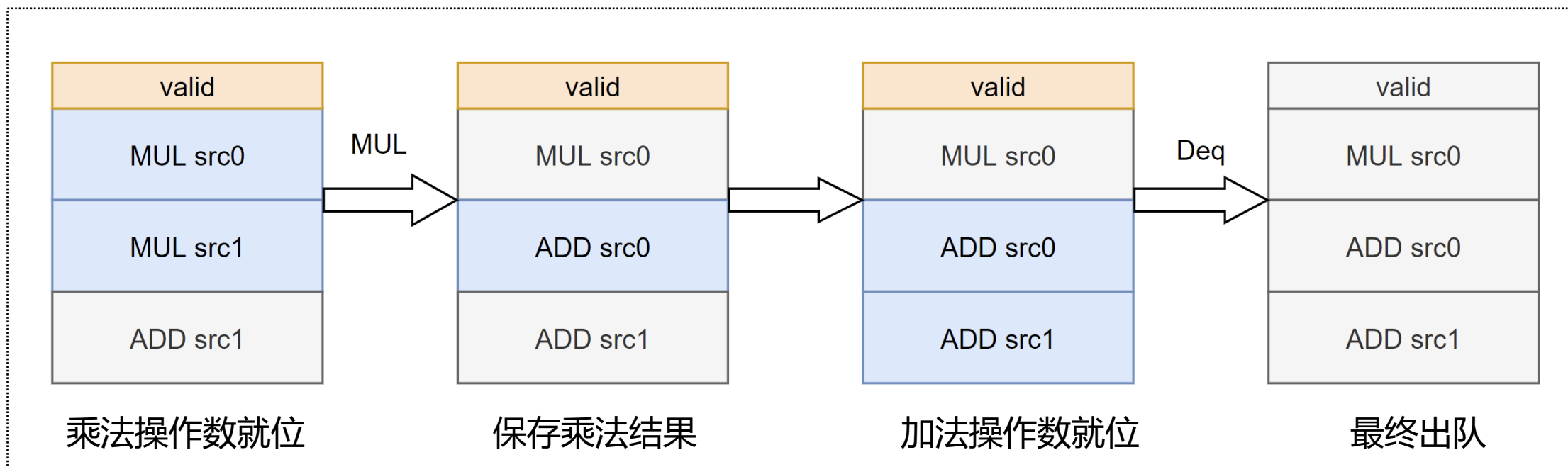


南湖架构 Age Matrix 算法



# Reservation Station: FMA 2-Step Execution

- 乘法操作数如果就位，可以提前出队计算
  - 乘法计算结果存到保留站，等待加法操作数
  - 保留站前递/时序定点优化



FMA 保留站 一项的变化过程



# B & K Extension

- B: Bit-Manipulation Extension , 位操作扩展
- K: Cryptographic Extension , 标量密码学扩展

1. Extensions .....	
1.1. Zba extension .....	
1.2. Zbb: Basic bit-manipulation .....	
1.2.1. Logical with negate .....	
1.2.2. Count leading/trailing zero bits .....	
1.2.3. Count population .....	
1.2.4. Integer minimum/maximum .....	
1.2.5. Sign- and zero-extension .....	
1.2.6. Bitwise rotation .....	
1.2.7. OR Combine .....	
1.2.8. Byte-reverse .....	
1.3. Zbc: Carry-less multiplication .....	
1.4. Zbs: Single-bit instructions .....	

## B Extension

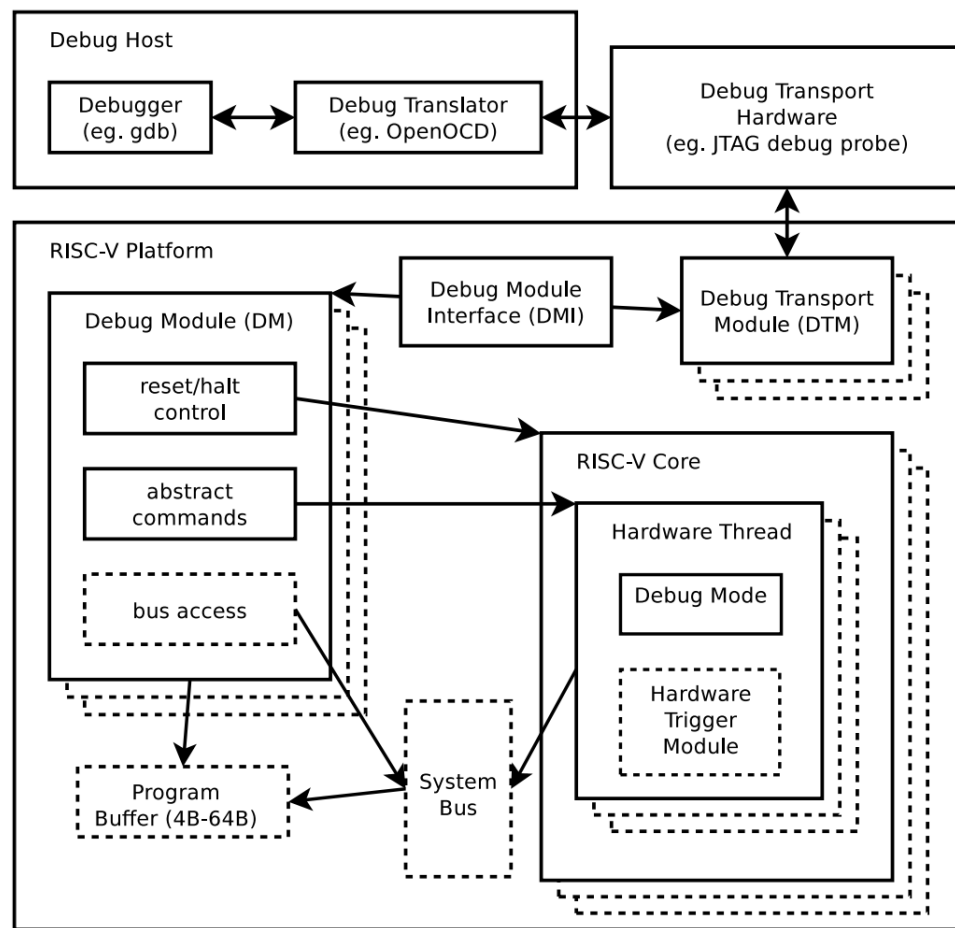
2. Extensions Overview .....	
2.1. Zbkb - Bitmanip instructions for Cryptography .....	
2.2. Zbkc - Carry-less multiply instructions .....	
2.3. Zbkx - Crossbar permutation instructions .....	
2.4. Zknd - NIST Suite: AES Decryption .....	
2.5. Zkne - NIST Suite: AES Encryption .....	
2.6. Zknh - NIST Suite: Hash Function Instructions .....	
2.7. Zksed - ShangMi Suite: SM4 Block Cipher Instructions .....	
2.8. Zksh - ShangMi Suite: SM3 Hash Function Instructions .....	
2.9. Zkr - Entropy Source Extension .....	
2.10. Zkn - NIST Algorithm Suite .....	
2.11. Zks - ShangMi Algorithm Suite .....	
2.12. Zk - Standard scalar cryptography extension .....	
2.13. Zkt - Data Independent Execution Latency .....	

## K Extension



# Debug Extension

- 让处理器上的程序通过gdb被远程调试
  - 暂停程序执行
  - 单步执行
  - 硬件断点 Trigger



**敬请批评指正！**