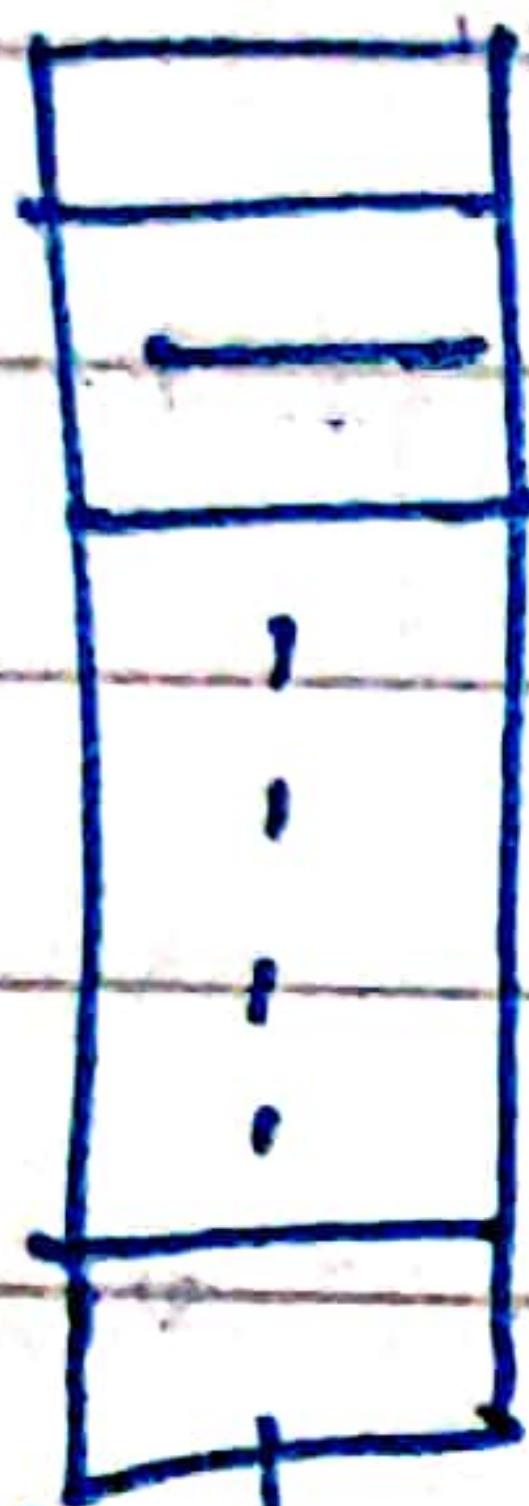


Basic Structure of a vector-register architecture

Vector-Vector Instruction

v_j register



v_k Register .

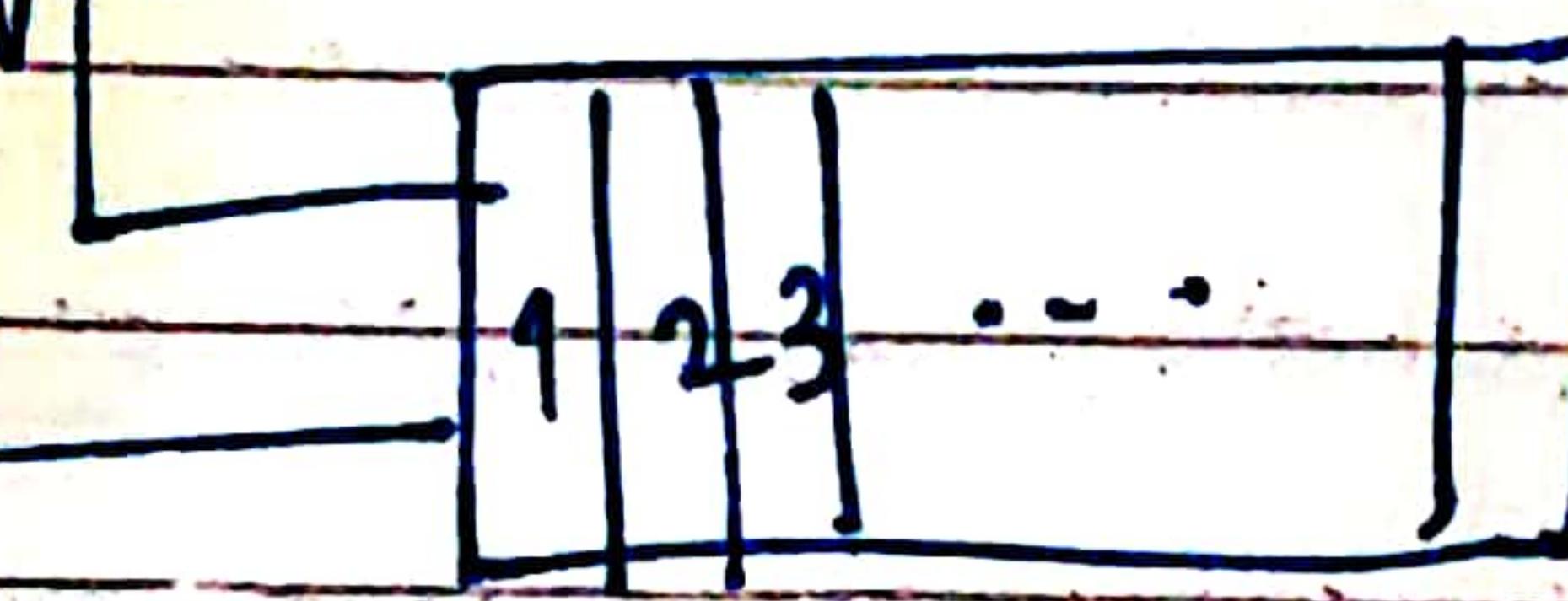


v_i Register



v_j Register.

v_k register

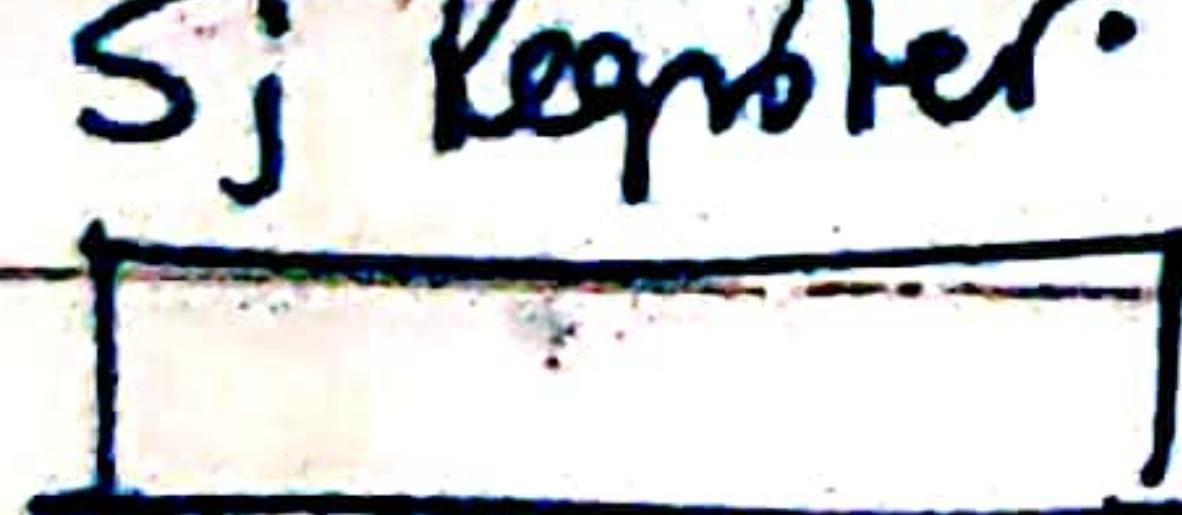


Functional Unit .

Vector-Scalar Instruction

v_k Register

s_j Register.

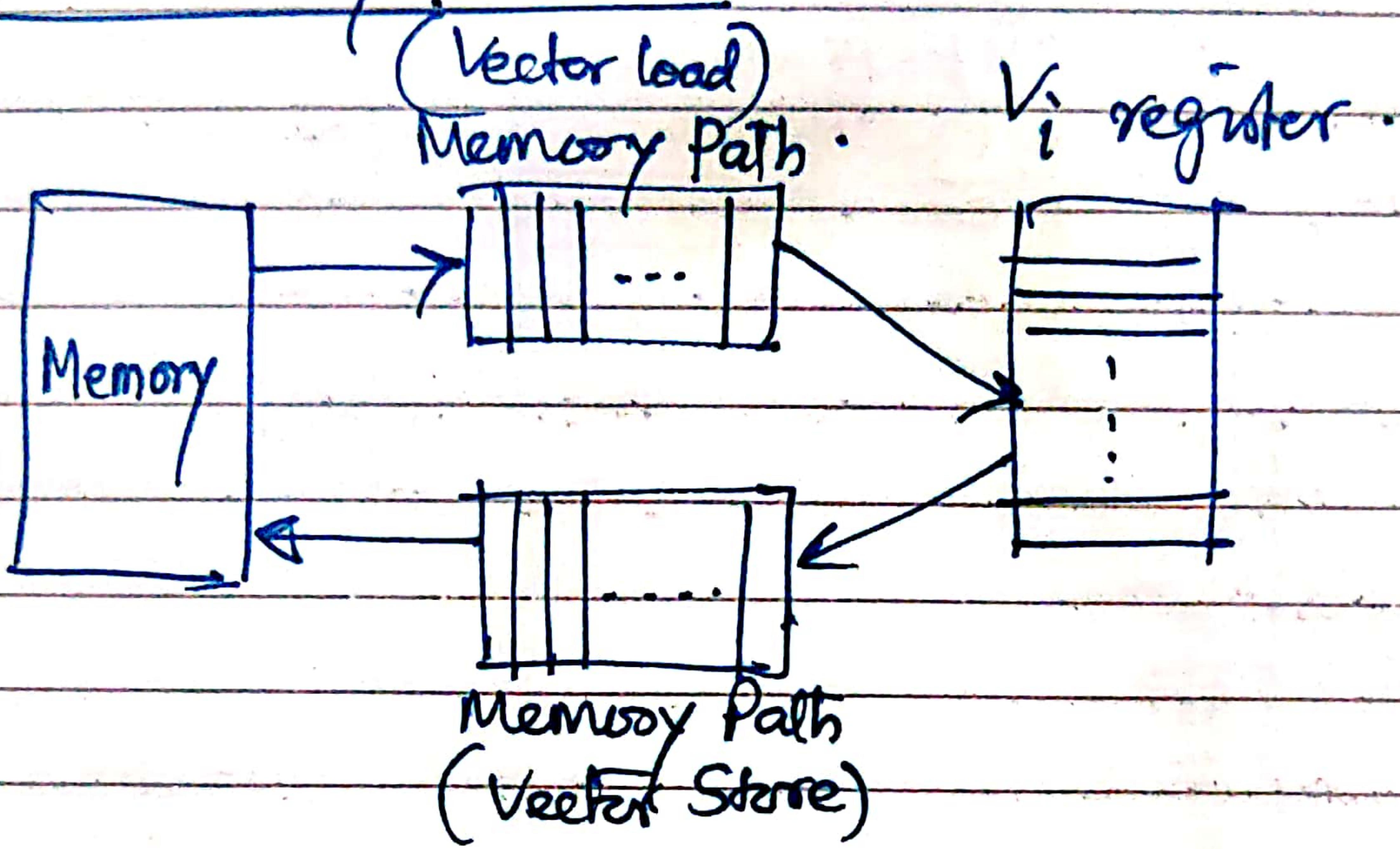


v_i Register.



functional Unit .

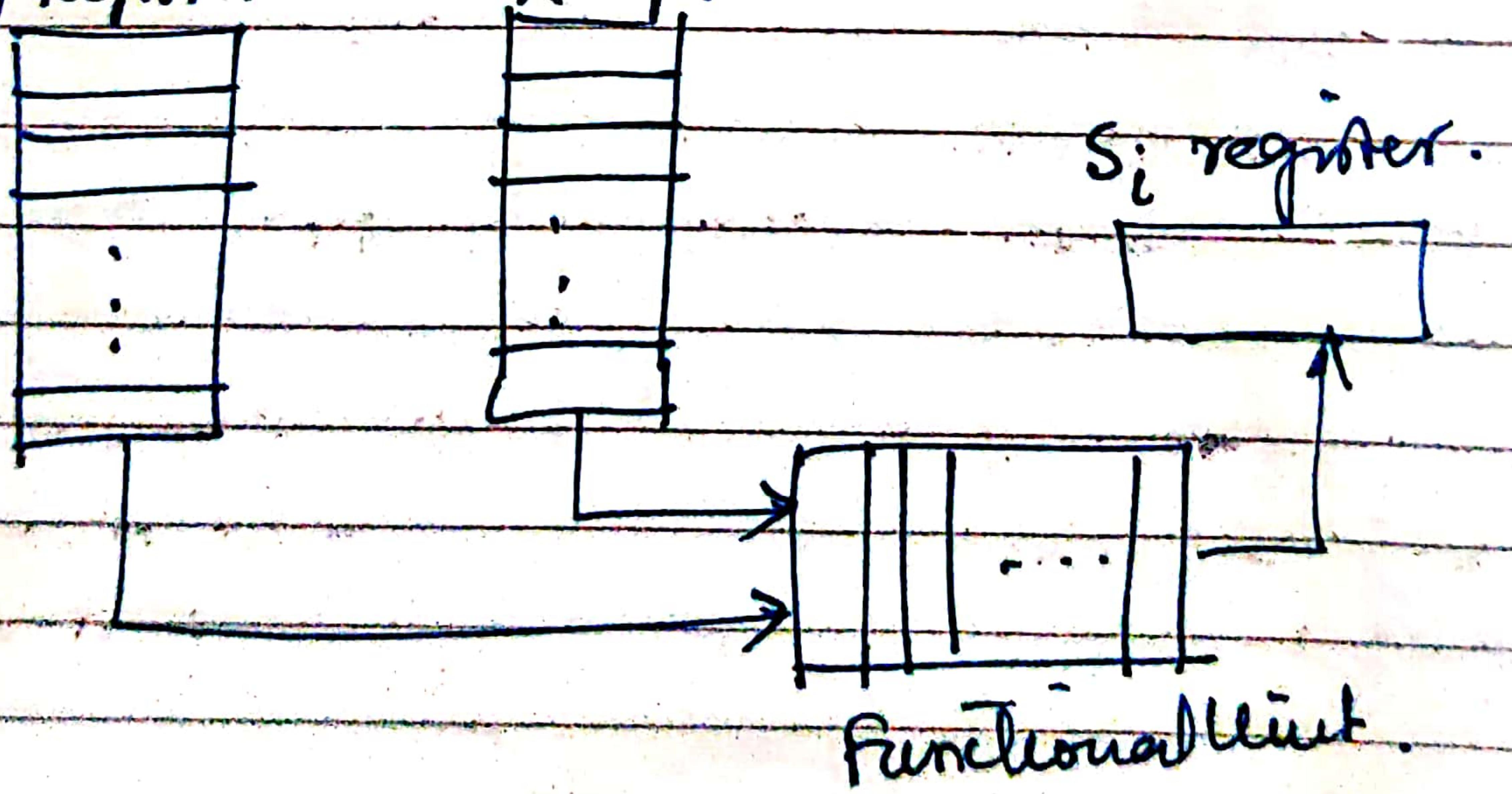
Vector-memory instructions



Vector Reduction instruction

V_j Register V_k Register.

S_i register.



Gather Instructions

Memory

Addr. Content

V_L Register

V₀ Register V₁ Register

100 0 200

4

1

600 *

101

300

2

400 \$

102

400

7

250 #

103

500

0

200 0

104

* 600

A₀

100

105 700

106 100

107 # 250

108 350

Scatter Instructions

Memory

V_L Register

V₀ Reg.

V₁ Reg

100

500

4

4

200

101

2

300

102

300

7

400

103

0

500

104

200

105

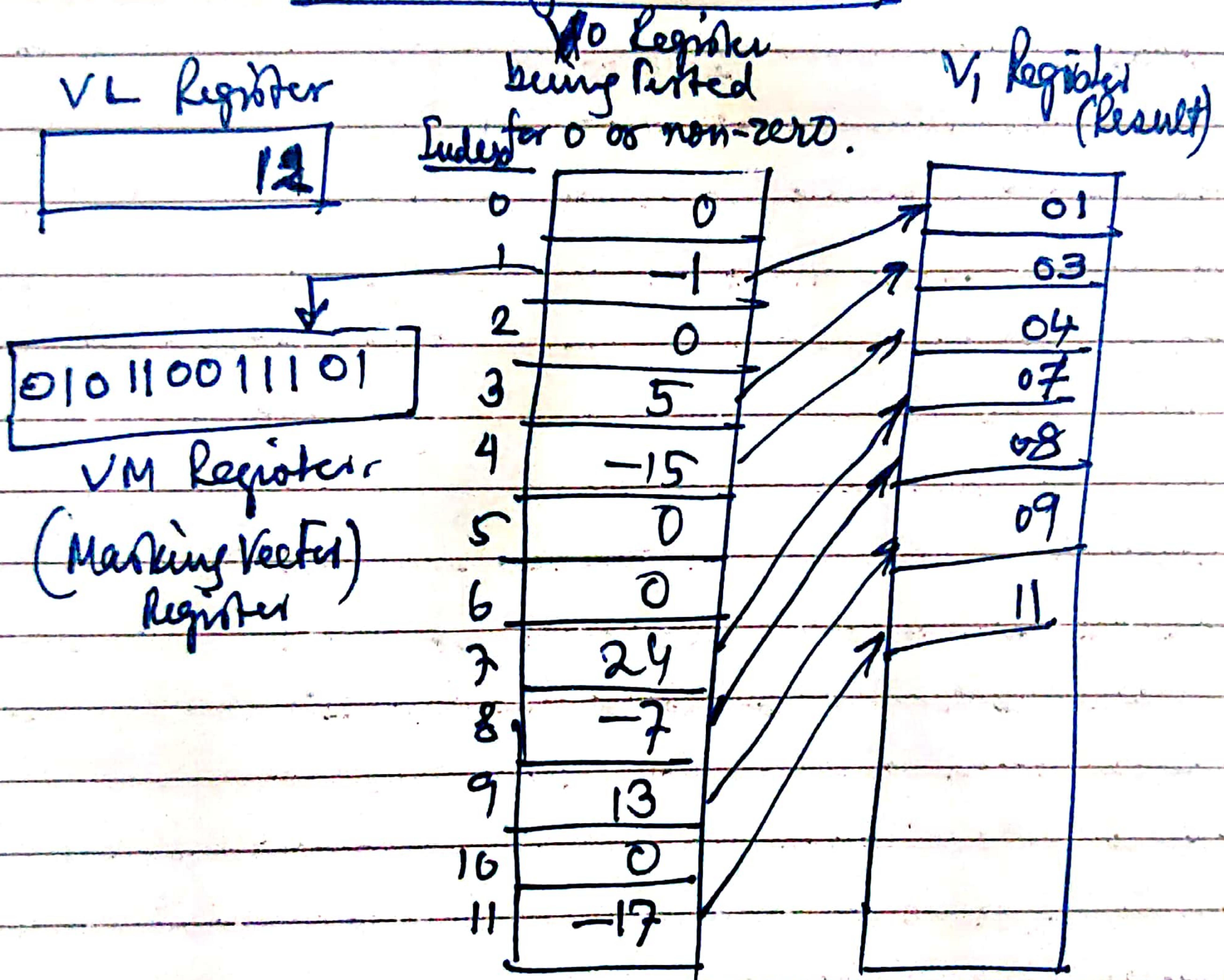
106

107

400

108

Masking Instruction



V_0 is putted for zero or non-zero elements

$VM \rightarrow 0$ indicates zero

\perp indicates non-zero.

$VL \rightarrow$ length of vector being tested

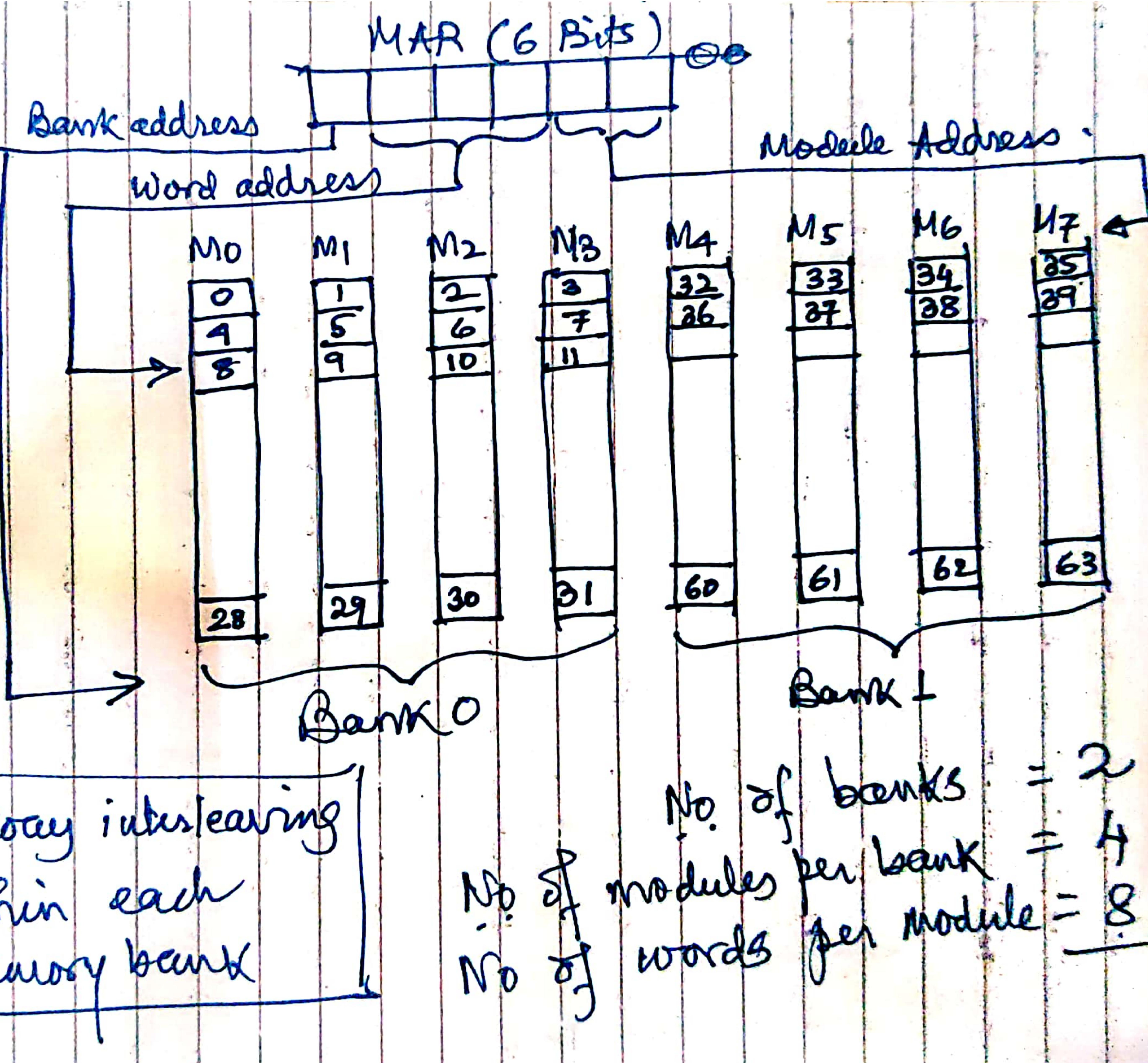
$V_1 \rightarrow$ Index of non-zero elements.

Vector Stride

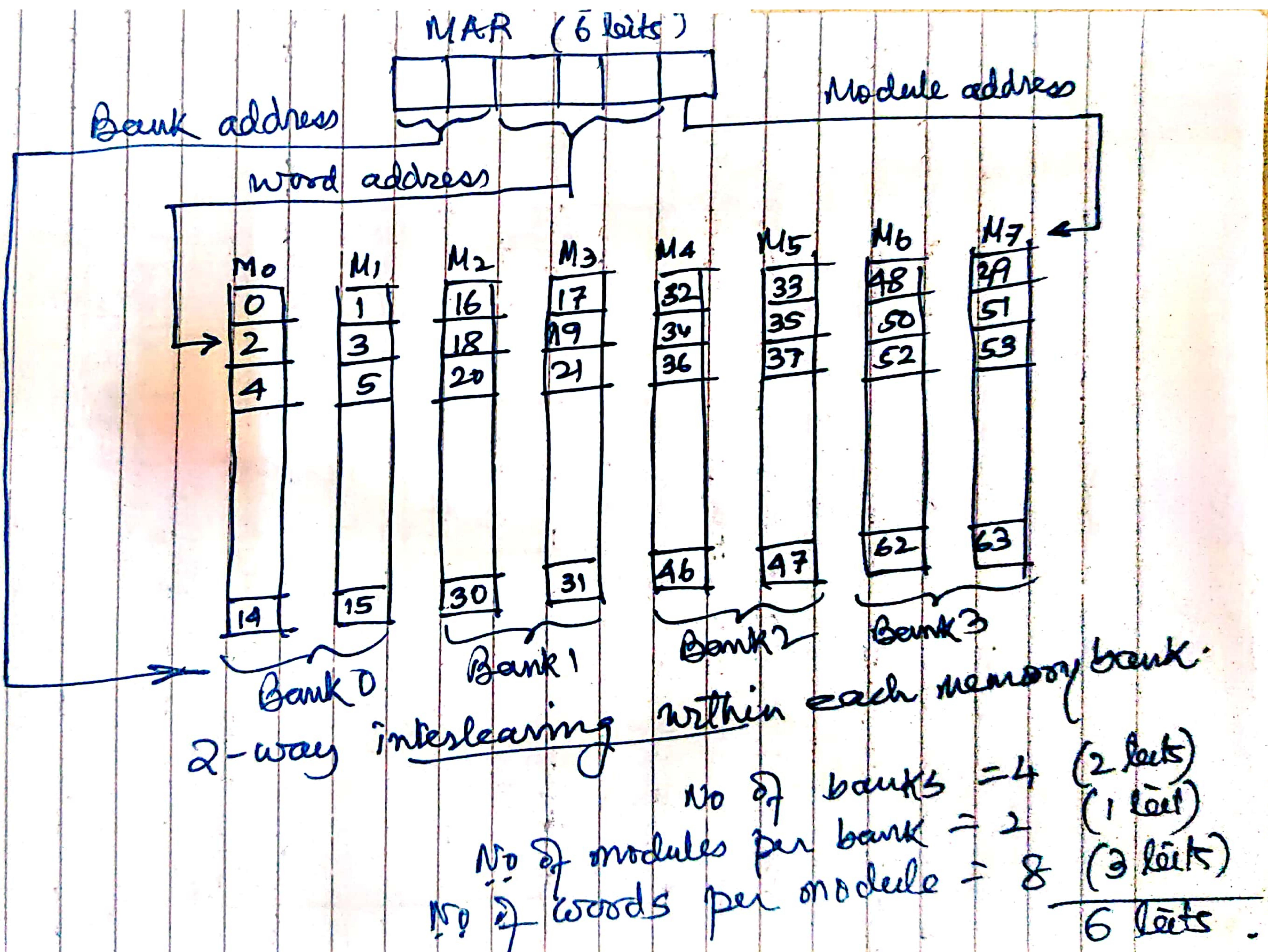
1	2	3	1
4	5	6	1
7	8	9	

Row major order: 1 2 3 4 5 6 7 8 9

Column major order: 1 4 7 2 5 8 3 6 9.

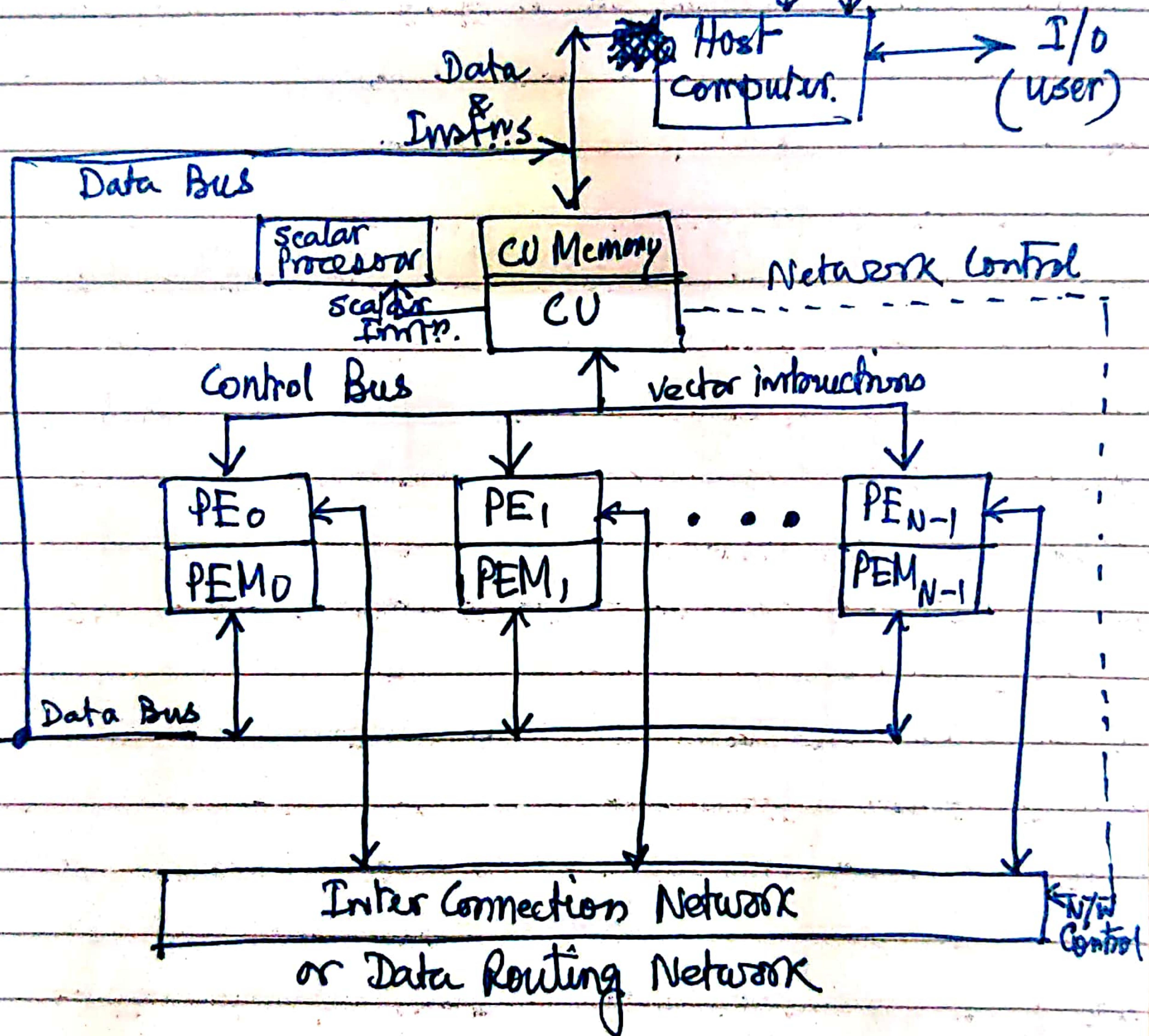


(1 bit)
(2 bits)
(3 bits)
6 bits

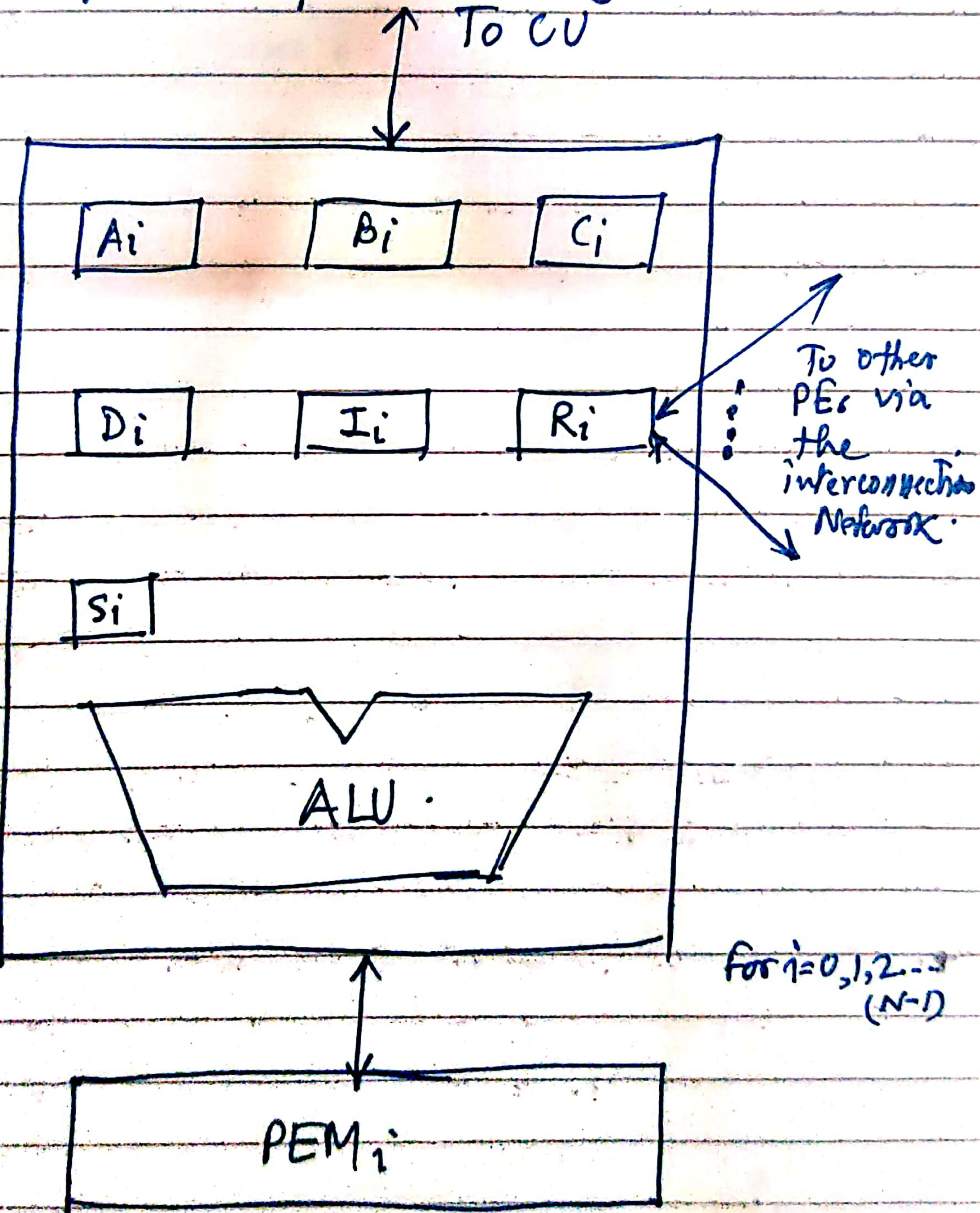


Configuration of Illiac IV

~~Magnetic Storage~~

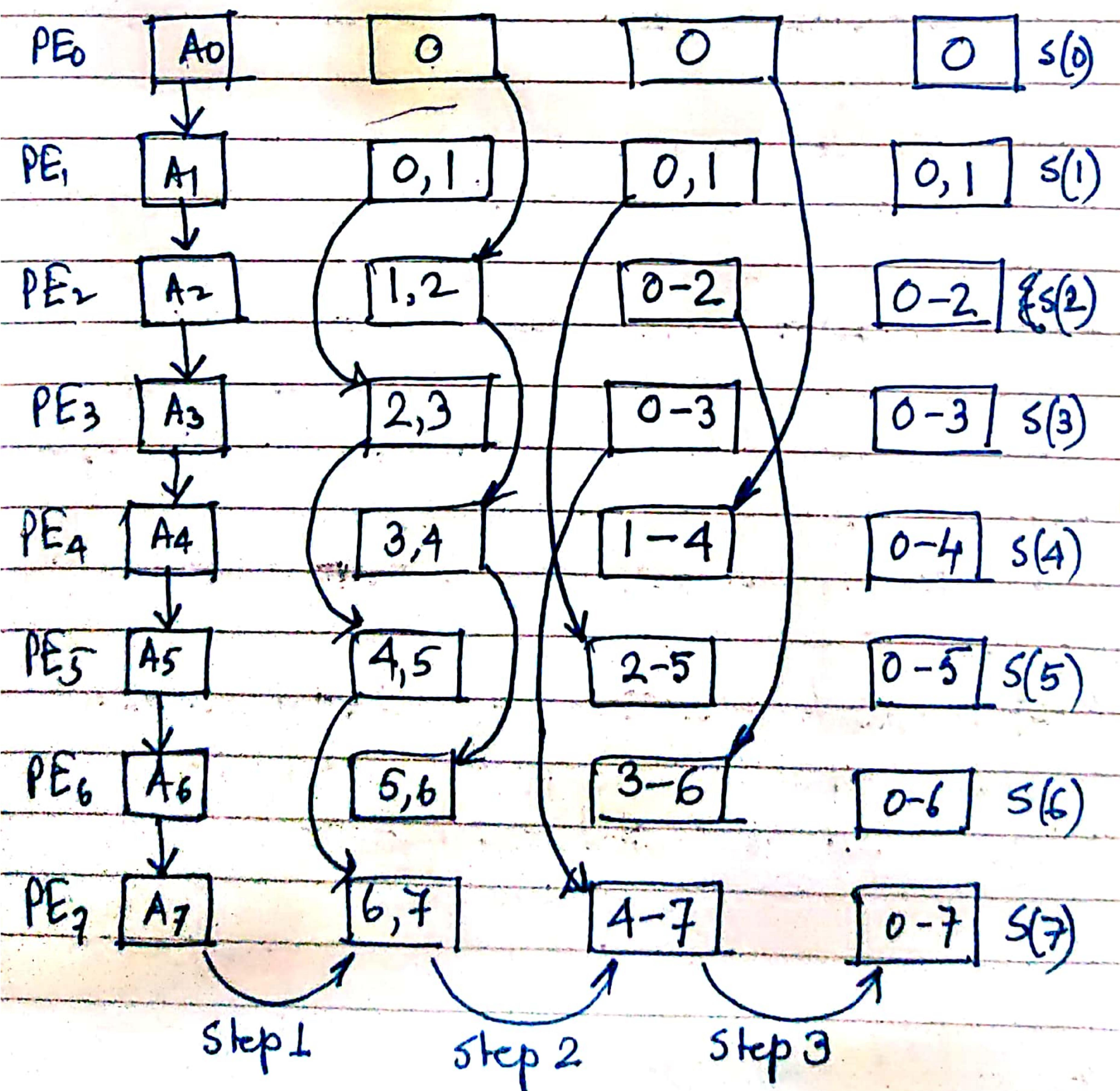


Components of a Processing Element



Calculation of $S(k) = \sum_{i=0}^k A_i$, $k=0, 1, \dots, 7$

in an SIMD Machine



Vector Stride / strip mining / vector chaining / 2018 [5(c)]

SIMD : Matrix Mult in ILLIAC-IV

Data routing logic of SIMD to compute $\sum_{i=0}^k A_i$.

Why masking mechanism in SIMD may pre-

2018 [4(c)] Mem-Mem Arch. not possible in VP or
array
Rake