Main Memory Design (SRAM)

Nahin Ul Sadad
Lecturer
CSE, RUET

Main Memory in CPU

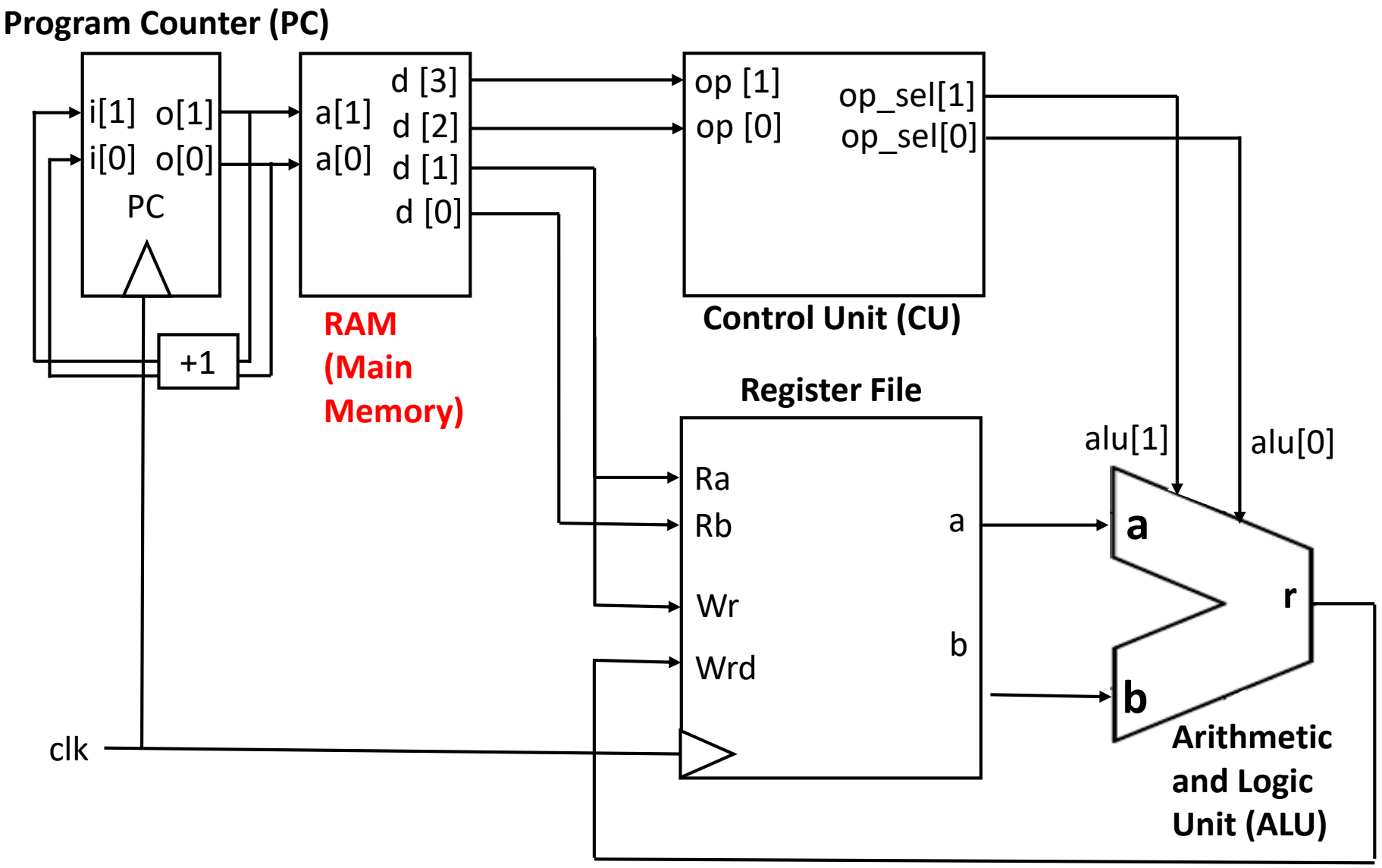


Figure: 1-bit CPU

1. Program Counter will have address of next instruction to be executed in current clock cycle.
2. Address in PC will be sent to **RAM** to retrieve instruction.
3. Instruction will be decoded by control unit and will select registers and/or immediate values.
4. Data within registers and/or immediate values will be sent to Arithmetic and Logic Unit (ALU) to perform operations.
5. ALU will perform operation and result will be sent to the register to be written.
6. Finally, PC will be incremented to point to the next instruction in next clock cycle.

Main Memory

Main memory is where programs and data are kept when the processor is actively using them. When programs and data become active, they are copied from secondary memory into main memory where the processor can interact with them. A copy remains in secondary memory.

Main memory is intimately connected to the processor, so moving instructions and data into and out of the processor is very fast.

Main memory is sometimes called RAM. RAM stands for Random Access Memory. "Random" means that the memory cells can be accessed in any order. However, properly speaking, "RAM" means the type of silicon chip used to implement main memory.

There are two type of RAM. They are: Static Random Access Memory (SRAM) and Dynamic Random Access Memory (DRAM).

SRAM vs DRAM

SRAM stands for Static Random-Access Memory. They store data as do flip-flops where extra 2 transistors are used for controlling the access. It is very fast and its power consumption is also less. As long as power is being supplied to the machine, SRAM will hold data and will lose it as soon as power will be disconnected.

DRAM stands for Dynamic Random-Access Memory. It is the second type of RAM and is manufactured with the help of transistors and capacitors. The function of capacitor is to store data such that the charged capacitor indicates the value 1 and the discharged capacitor indicated a 0.

Capacitor always eventually discharges owing to the fact that charges leak. This phenomenon is shown by the word dynamic. This happens even when power is provided at all times. This is the very reason that DRAM utilizes greater amount of power. If we have to make the data last for a long period of time, there is a constant need to refresh the data. For greater amount of storage, DRAM is used. It is cheaper as only one transistor is needed for one memory block.

Main Memory (SRAM)

Question:

Suppose, you want to store 100 instructions in your main memory (RAM) and size of your ISA is 13 bits, then

1. How many address lines will be needed in RAM?
2. What should be the minimum word size of RAM so that instruction can be stored in one address?

Main Memory (SRAM)

Question:

Suppose, you want to store 100 instructions in your main memory (RAM) and size of your ISA is 13 bits, then

1. How many address lines will be needed in RAM?

   Answer: **No. of address lines** $= log_2(100) = 6.64 \approx 7$ **bits**

2. What should be the minimum word size of RAM so that instruction can be stored in one address?

   Answer: **Since size of ISA is 13 bits, minimum word size of RAM must be 13 bits.**

Main Memory (SRAM)

RAM chip is shown below:

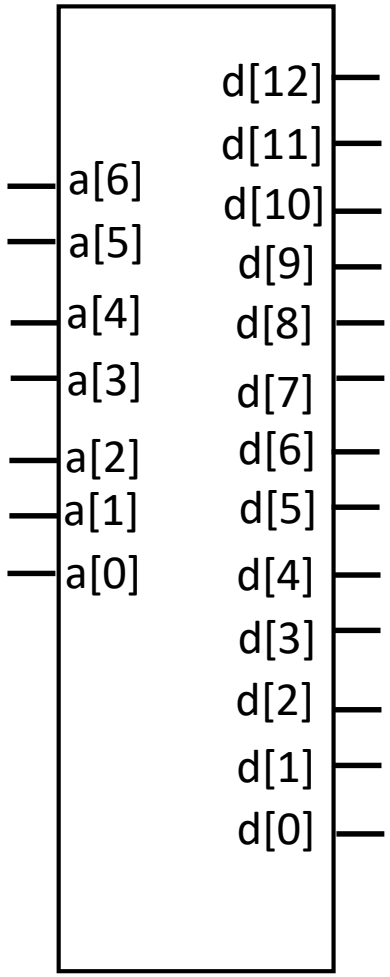


Figure: RAM chip (128x13)

Sample Code

Consider following pseudo code which compares two numbers:

Pseudo Code

`LABEL:`
`R4 = 5`
`JMP LABEL`

Now translate this code to assembly code!

Sample Code

Consider following pseudo code which compares two numbers:

Pseudo Code

```
LABEL:
R4 = 5
JMP LABEL
```

Assembly

Code

```
LABEL:
        XOR R4, R4    ;clearing R4
        ADDI R4, 5    ;R4 = 5
        JMP LABEL     ;JMP to Beginning
```

Sample Code

Consider assembly code with address:

Address	Code
00	`LABEL: XOR R4, R4`
01	`ADDI R4, 5`
02	`JMP LABEL`

Now translate this code to machine code!

Sample Code

Consider assembly code with address:

Address	Code
00	LABEL: XOR R4, R4
01	ADDI R4, 5
02	JMP LABEL

Address	Code	Opcode (2 bits)	Opcode (4 bits)	Register 1	Register 2	Constant	Address	Machine Code											
								12	11	10	9	8	7	6	5	4	3	2	1	0
0000000 (0)	LABEL: XOR R4, R4	00 (A&L Reg)	0010 (XOR)	100 (R4)	100 (R4)	X	X	0	0	0	0	1	0	1	0	0	1	0	0	1
0000001 (1)	ADDI R4, 5	01 (A&L Imm)	0101 (ADD)	100 (R4)	X	0101 (5)	X	0	1	0	1	0	1	1	0	0	0	1	0	1
0000010 (2)	JMP LABEL	10 (Branching)	0000 (JMP)	X	X	X	0000000 (0)	1	0	0	0	0	0	0	0	0	0	0	0	0

Main Memory (SRAM)

- **Suppose, following instructions of a program are stored in RAM after moving it from HDD.**

Address	Instructions												
	12	11	10	9	8	7	6	5	4	3	2	1	0
0000000	0	0	0	0	1	0	1	0	0	1	0	0	1
0000001	0	1	0	1	0	1	1	0	0	0	1	0	1
0000010	1	0	0	0	0	0	0	0	0	0	0	0	0

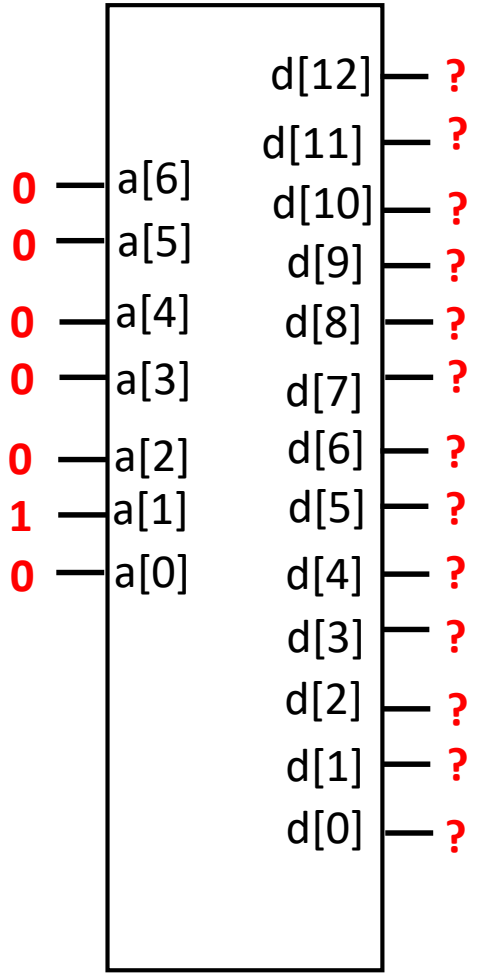


Figure: RAM chip (128x13)

Main Memory (SRAM)

- **Suppose, following instructions of a program are stored in RAM after moving it from HDD.**

Address	Instructions												
	12	11	10	9	8	7	6	5	4	3	2	1	0
0000000	0	0	0	0	1	0	1	0	0	1	0	0	1
0000001	0	1	0	1	0	1	1	0	0	0	1	0	1
0000010	1	0	0	0	0	0	0	0	0	0	0	0	0

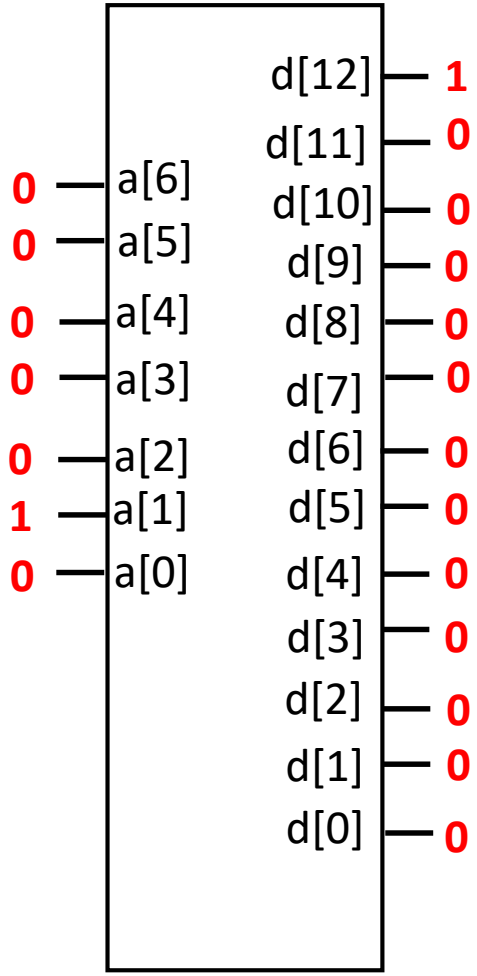


Figure: RAM chip (128x13)

Main Memory (SRAM)

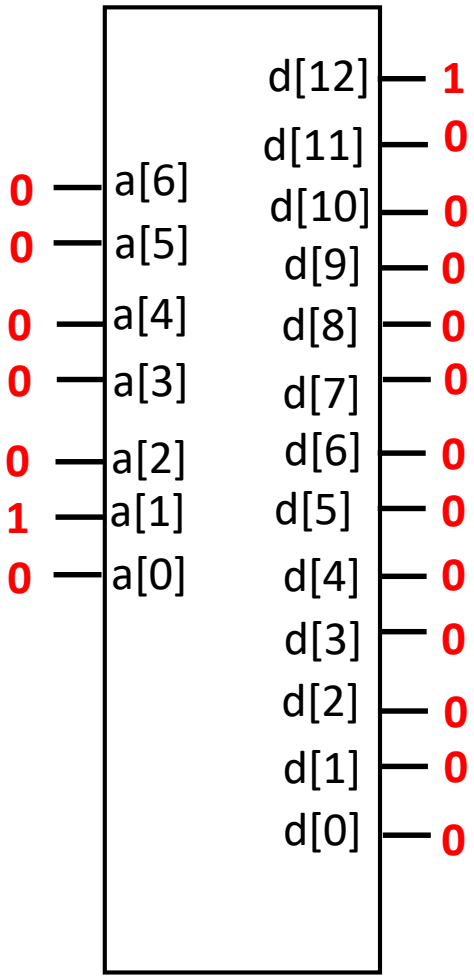
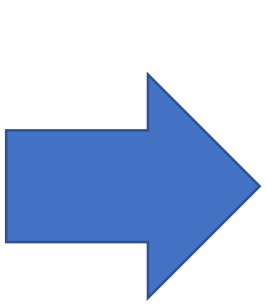


Figure: RAM chip (128x13)



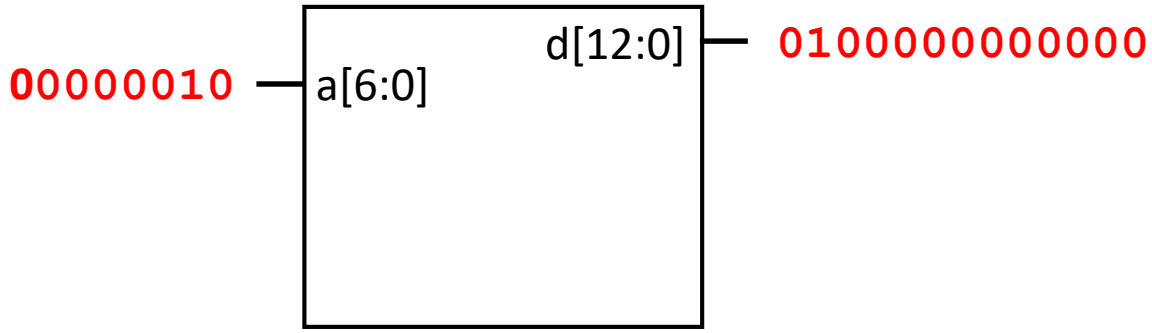


Figure: RAM chip (128x13)

This a Single port SRAM (without write operation)

Single port SRAM (without write operation)

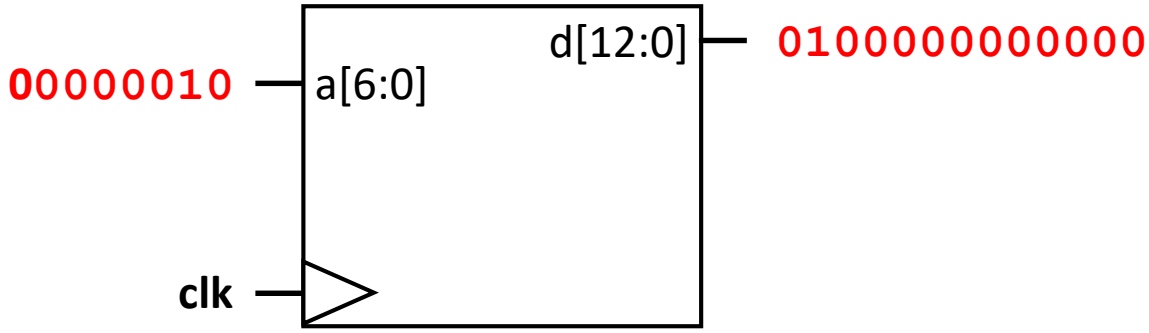


Figure: RAM chip (128x13)

Address	Instructions												
	12	11	10	9	8	7	6	5	4	3	2	1	0
0000000	0	0	0	0	1	0	1	0	0	1	0	0	1
0000001	0	1	0	1	0	1	1	0	0	0	1	0	1
0000010	1	0	0	0	0	0	0	0	0	0	0	0	0

Single port SRAM (without write operation)

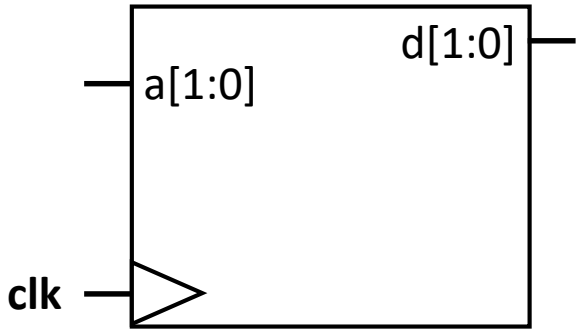


Figure: RAM chip **(?x?)**

Question:

1. What is the size of RAM?
2. What is the word size of RAM?

Single port SRAM (without write operation)

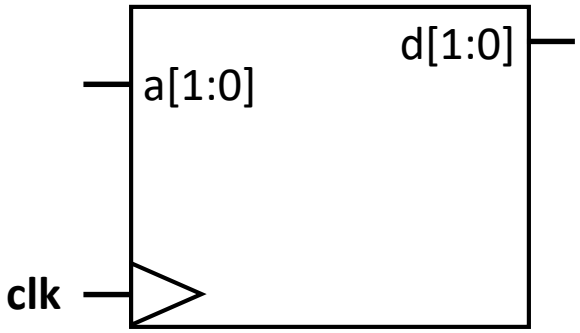


Figure: RAM chip **(4x2)**

Question:

1. What is the size of RAM? **Ans: 4**
2. What is the word size of RAM? **Ans: 2**

Single port SRAM (with write operation)

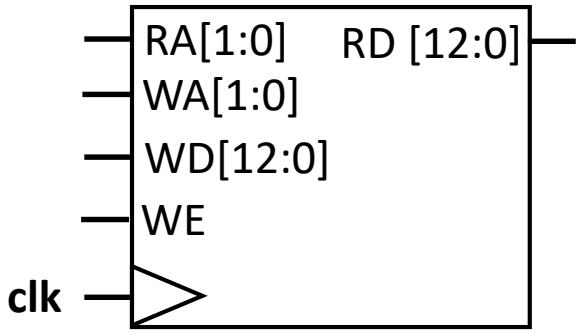


Figure: RAM chip **(4x13)**

Here,

RA = Read Address
RD = Read Data
WA = Write Address
WD = Write Data
WE = Write Enable

Dual port SRAM (with write operation)

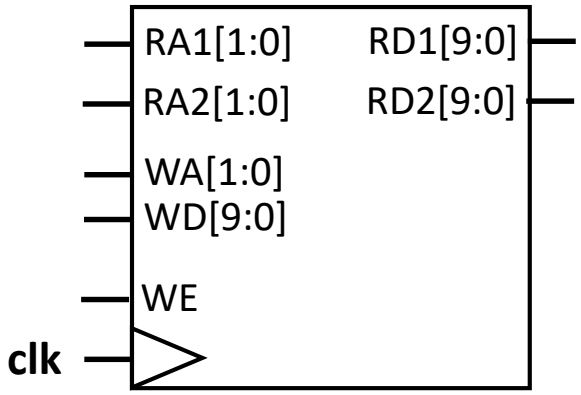


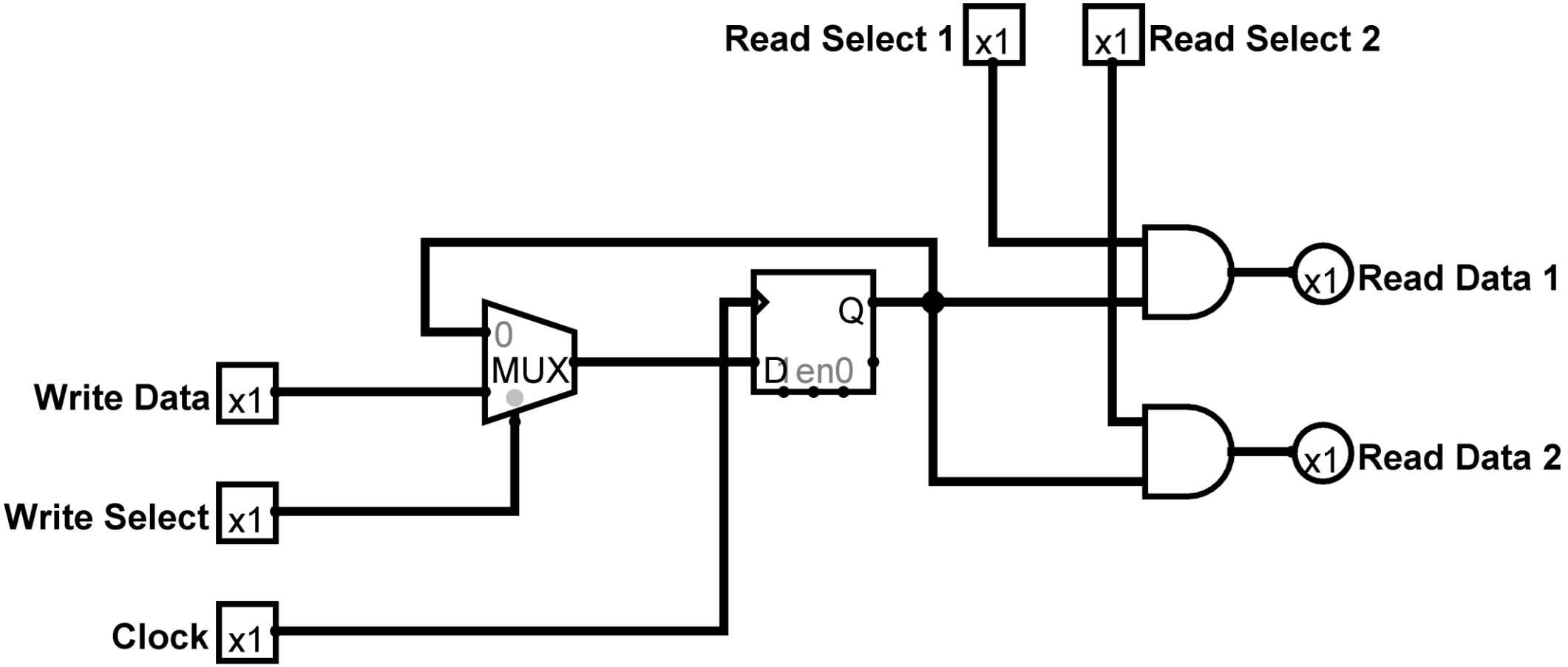
Figure: RAM chip **(4x10)**

Here,

RA = Read Address
RD = Read Data
WA = Write Address
WD = Write Data
WE = Write Enable

1x1 Static RAM (SRAM) Cell

1x1 SRAM



To read data from **Read data 1** of 1x1 SRAM, **Read Select 1** must be equal **1**
To read data from **Read data 2** of 1x1 SRAM, **Read Select 2** must be equal **1**

To write **Write data** to 1x1 SRAM, **Write Select** must be **1** and Clock must be positive edge.

1x1 SRAM Simulation

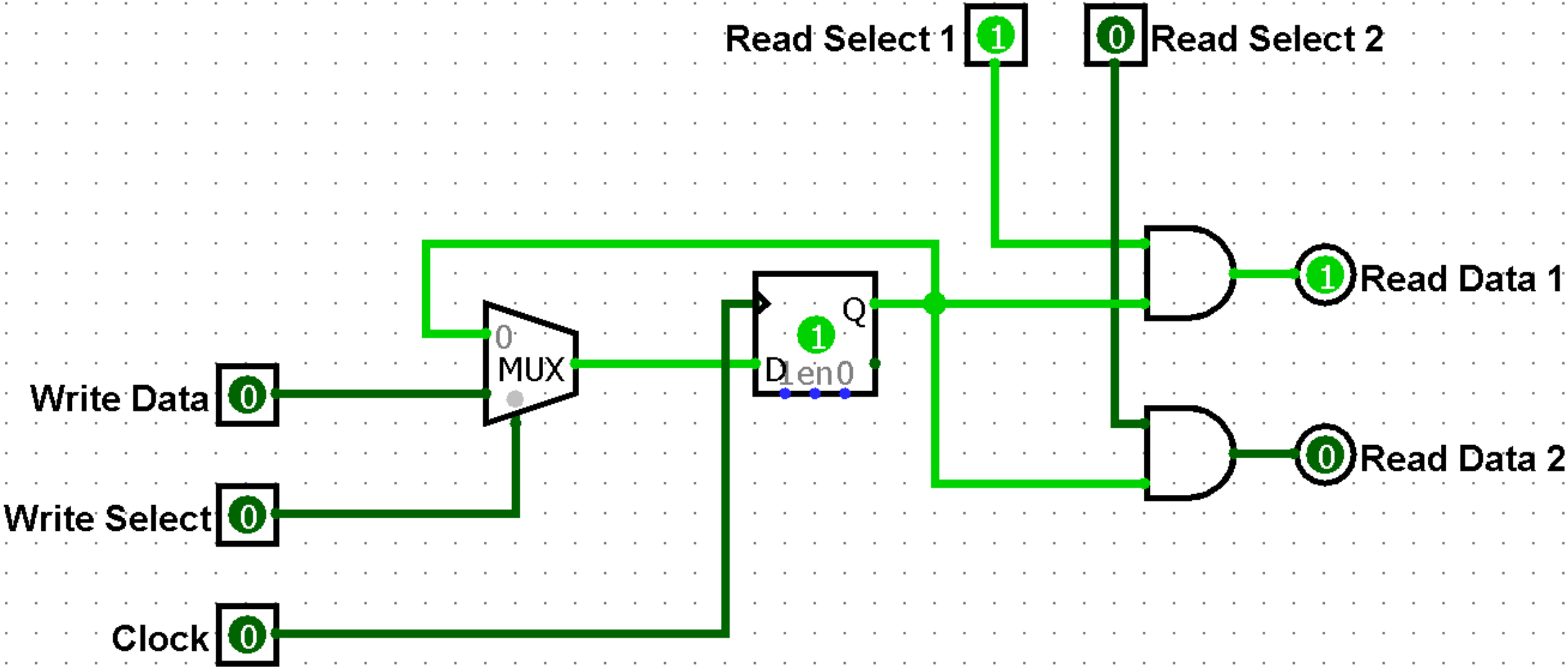


Figure: 1x1 SRAM Simulation.
Reading Data from 1x1 SRAM using `Read Data 1` port.
1x1 SRAM has stored 1. So, `Read Data 1` port shows 1 value.

1x1 SRAM Simulation

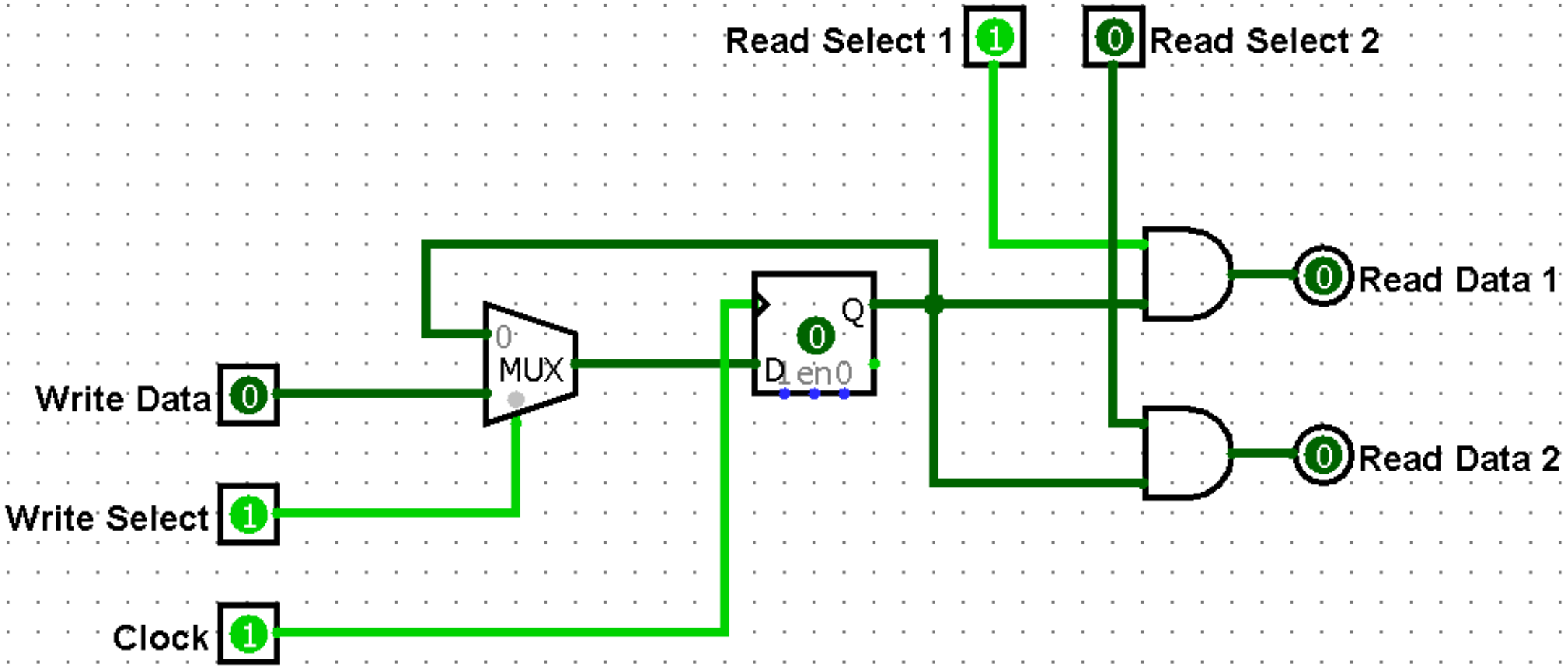


Figure: 1x1 SRAM Simulation.

Reading Data from 1x1 SRAM using `Read Data 1` port because `Read Select 1` is 1.

`Write data` 0 is written to 1x1 SRAM because `Write Select` is 1.

So, `Read Data 1` port shows 0 value.

1x1 SRAM

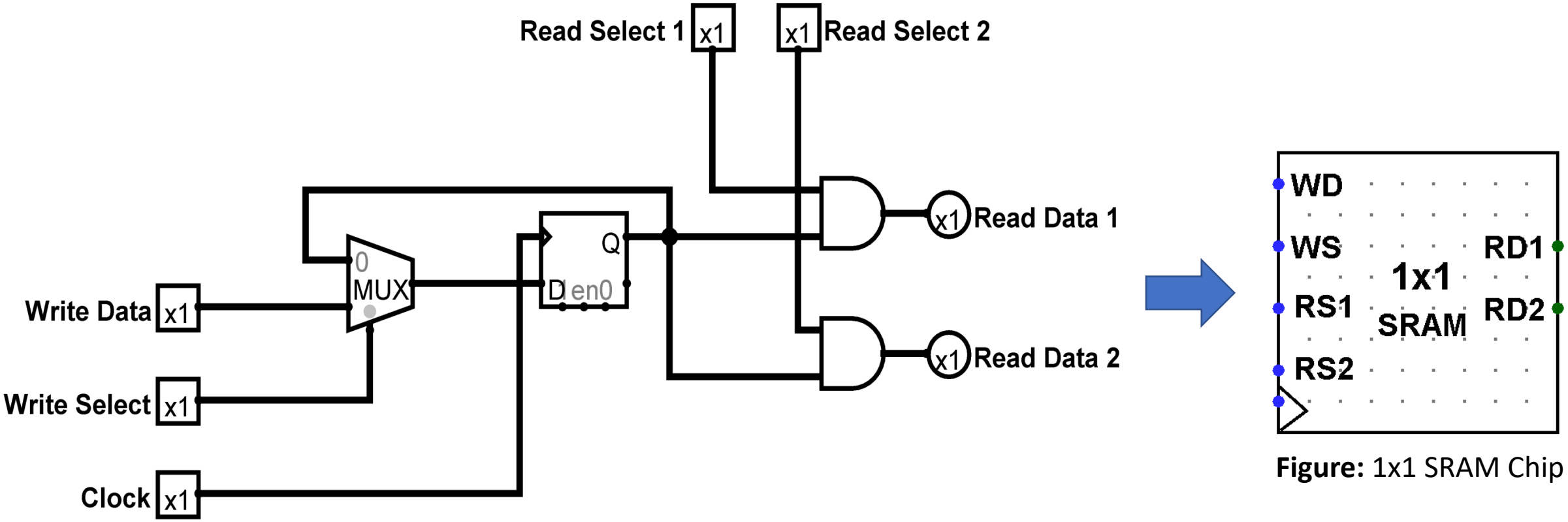
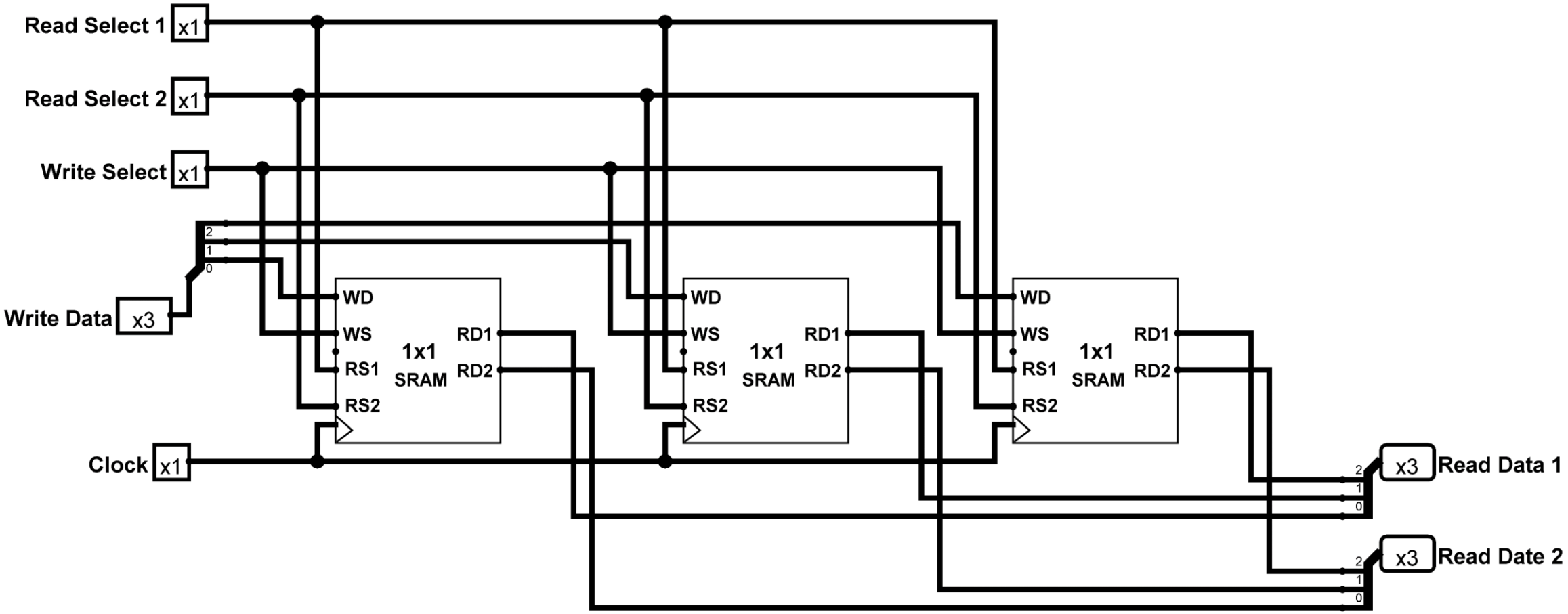


Figure: 1x1 SRAM Chip

1x3 Static RAM (SRAM) Cell

1x3 SRAM


Read Select 1
x1
Read Select 2
x1
Write Select
x1
Write Data
x3
2
1
0
Clock
x1
WD
WS
RS1
RS2
1x1
SRAM
RD1
RD2
WD
WS
RS1
RS2
1x1
SRAM
RD1
RD2
WD
WS
RS1
RS2
1x1
SRAM
RD1
RD2
x3
Read Data 1
x3
Read Date 2

1x3 SRAM Simulation

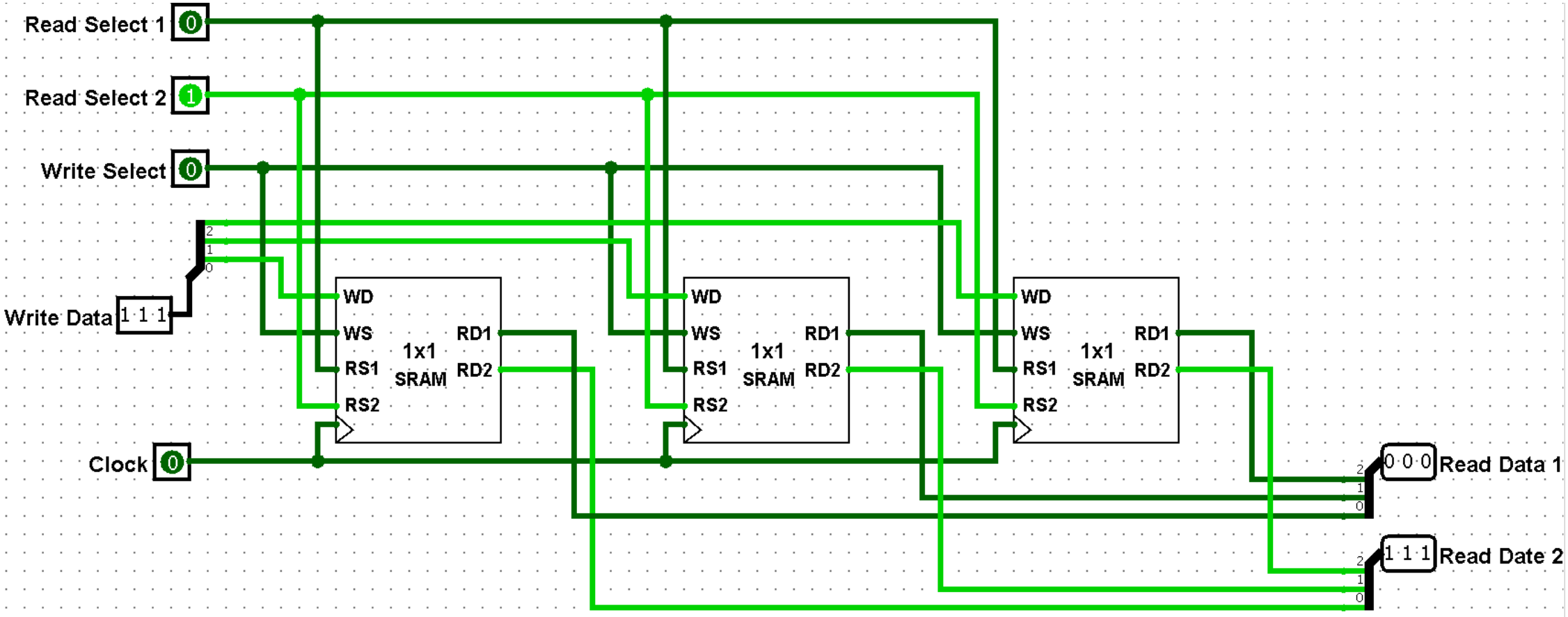


Figure: 1x3 SRAM Simulation.

Reading Data from 1x3 SRAM using `Read Data 2` port because `Read Select 2` is `1`.

1x3 SRAM stores `111`. So, `Read Data 2` port shows `111` value.

1x3 SRAM Simulation

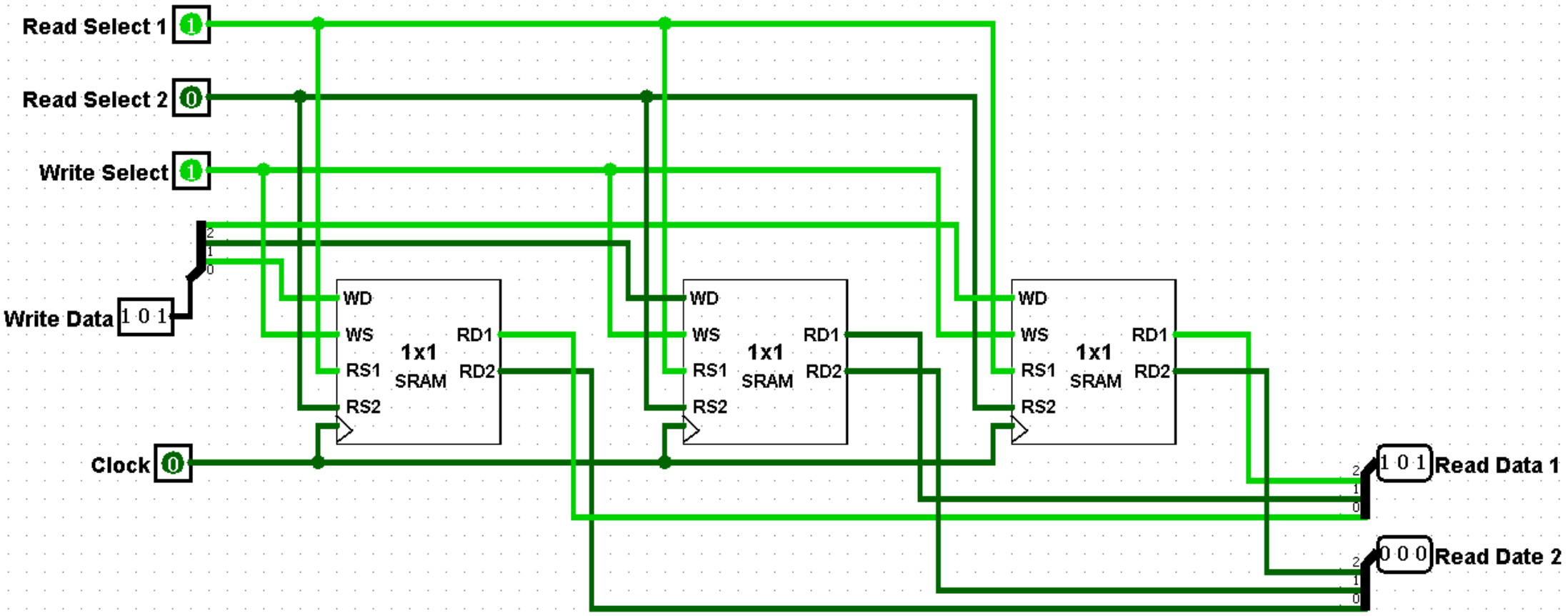


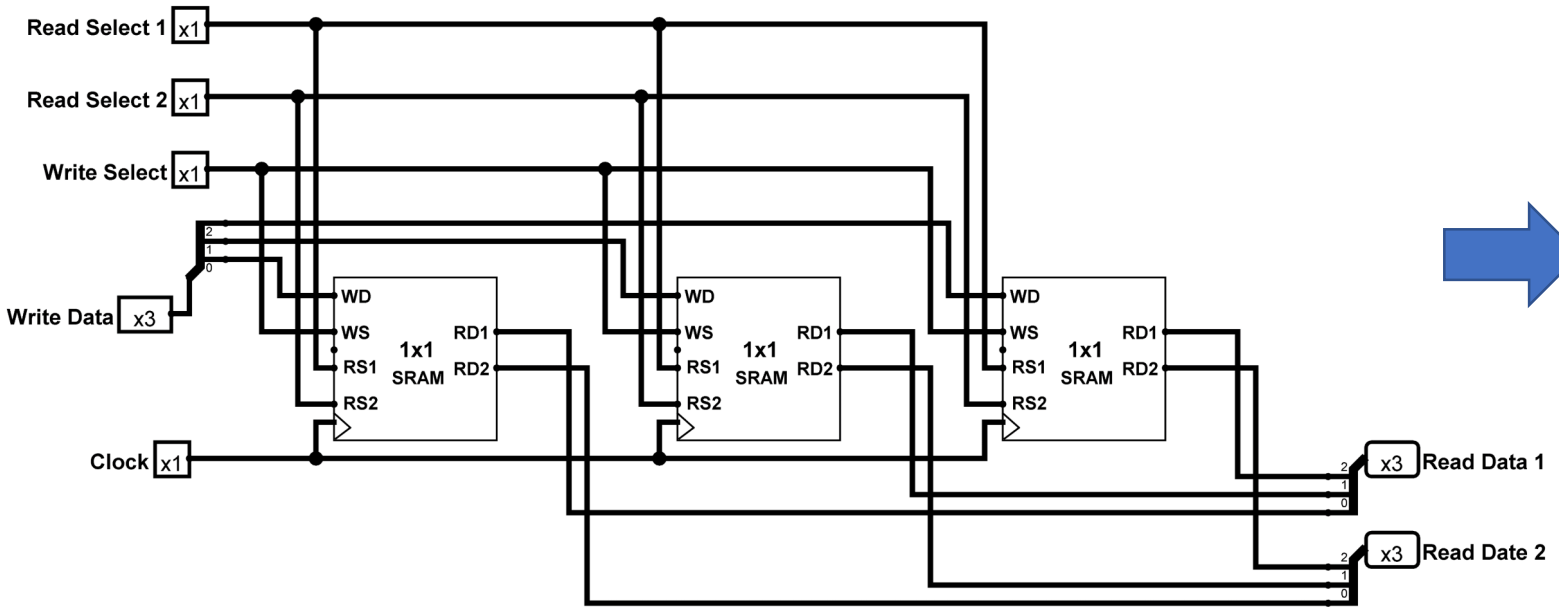
Figure: 1x3 SRAM Simulation.

`Write data` 101 is written to 1x3 SRAM because `Write Select` is `1`.

Reading Data from 1x3 SRAM using `Read Data 1` port because `Read Select 1` is 1.

So, `Read Data 1` port shows `101` value.

1x3 SRAM



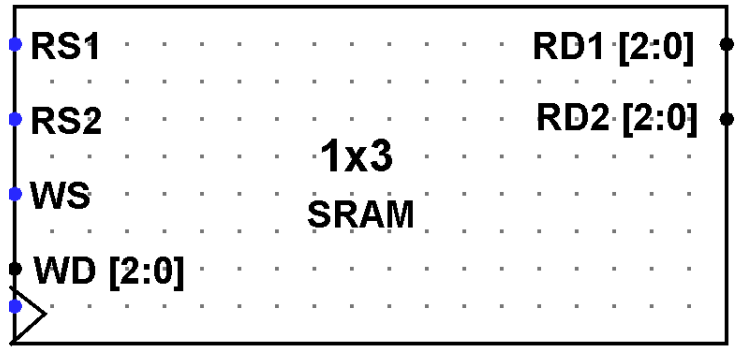
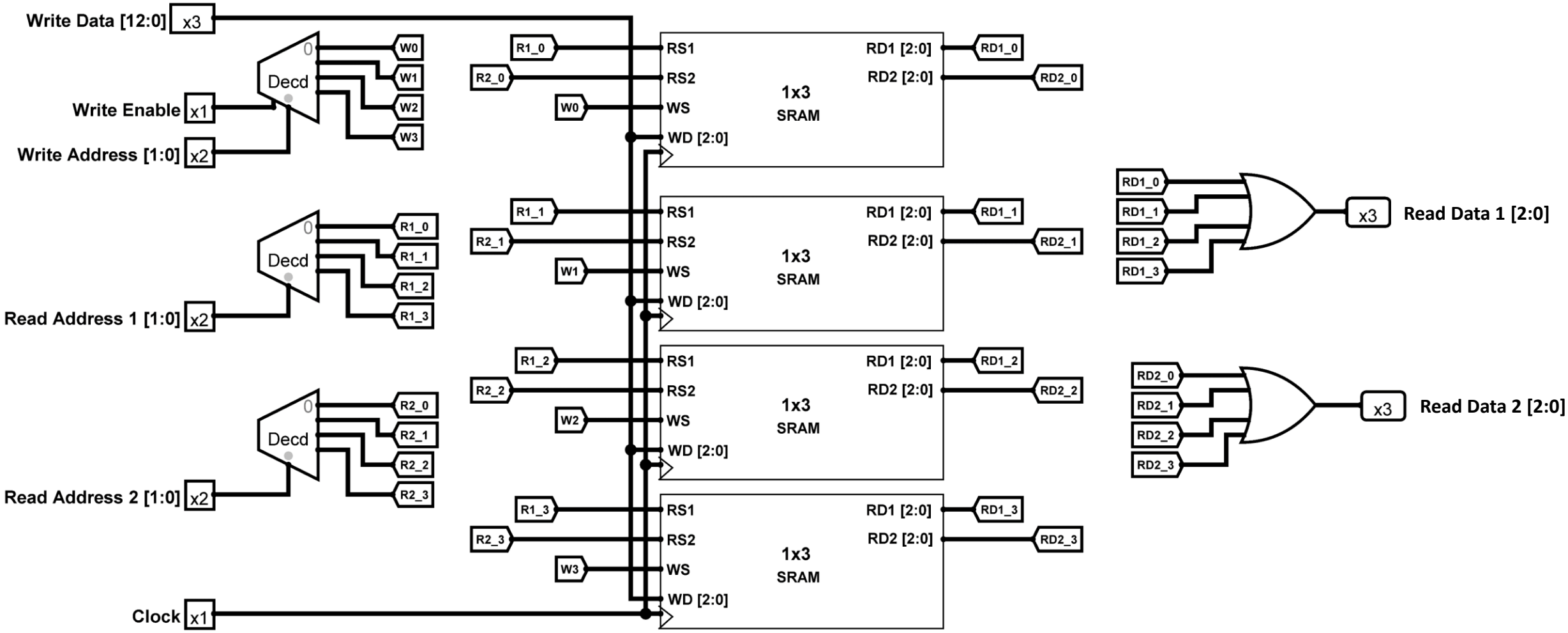


Figure: 1x3 SRAM Chip

4x3 Static RAM (SRAM)

4x3 SRAM


Write Data [12:0]
x3
Write Enable
x1
Write Address [1:0]
x2
Decd
0
W0
W1
W2
W3
Read Address 1 [1:0]
x2
Decd
0
R1_0
R1_1
R1_2
R1_3
Read Address 2 [1:0]
x2
Decd
0
R2_0
R2_1
R2_2
R2_3
Clock
x1
R1_0
R2_0
W0
RS1
RS2
WS
WD [2:0]
1x3
SRAM
RD1 [2:0]
RD2 [2:0]
RD1_0
RD2_0
R1_1
R2_1
W1
RD1_1
RD2_1
R1_2
R2_2
W2
RD1_2
RD2_2
R1_3
R2_3
W3
RD1_3
RD2_3
RD1_0
RD1_1
RD1_2
RD1_3
x3
Read Data 1 [2:0]
RD2_0
RD2_1
RD2_2
RD2_3
x3
Read Data 2 [2:0]

4x3 SRAM Simulation

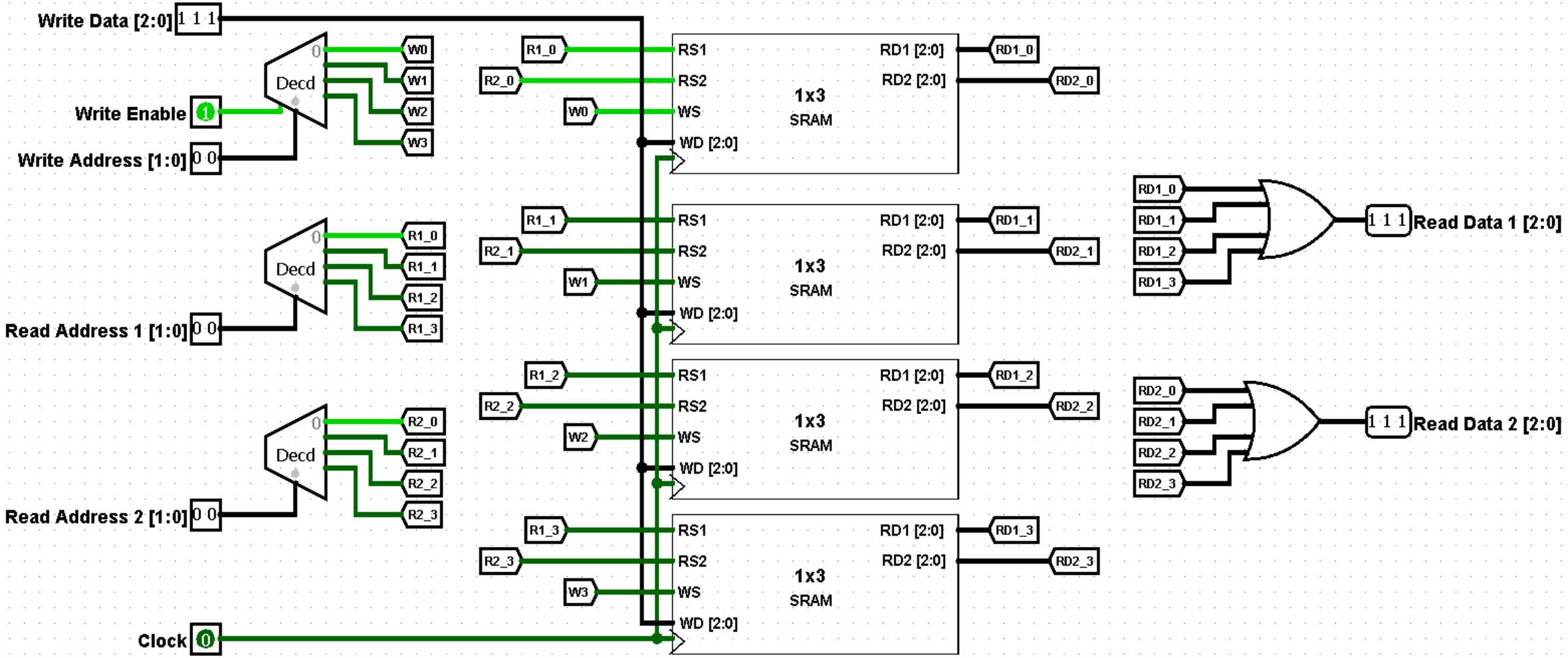


Figure: 4x3 SRAM Simulation.

Reading Data from 4x3 SRAM at address 00 using `Read Data 1` port because `Read Address 1` is `00`.

Reading Data from 4x3 SRAM at address 00 using `Read Data 2` port because `Read Address 2` is `00`.

`Write Data` `111` is written to 4x1 SRAM at `Write Address` `00` which is enabled by `Write Enable` port.

So, `Read Data 1` and `Read Data 2` port shows 111 value.

4x3 SRAM

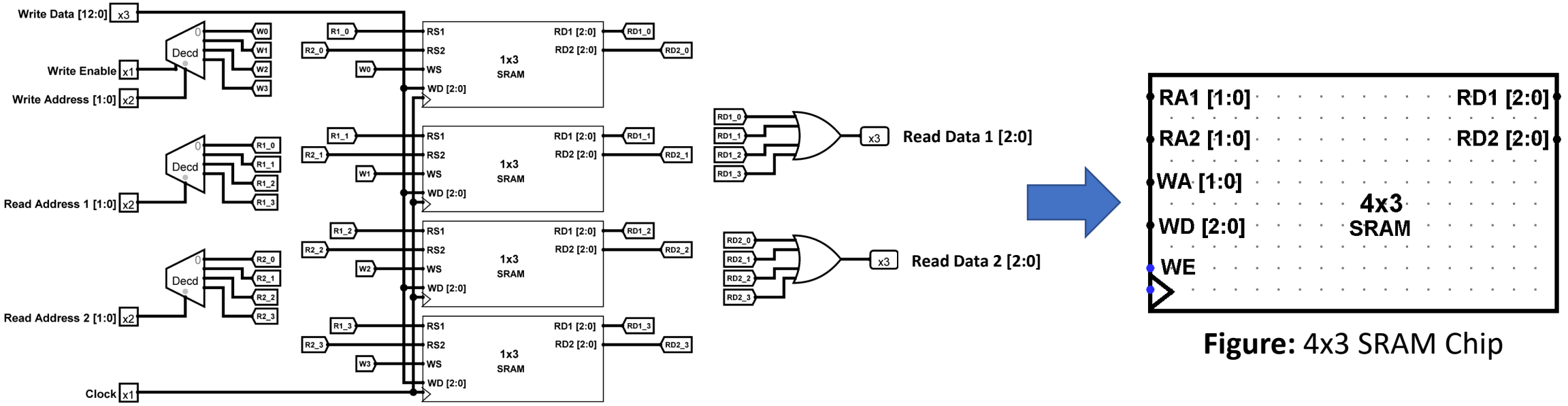


Figure: 4x3 SRAM Chip

Example: Main Memory

Question: Draw 4x3 Main Memory having Dual ports with write operation.

Answer:

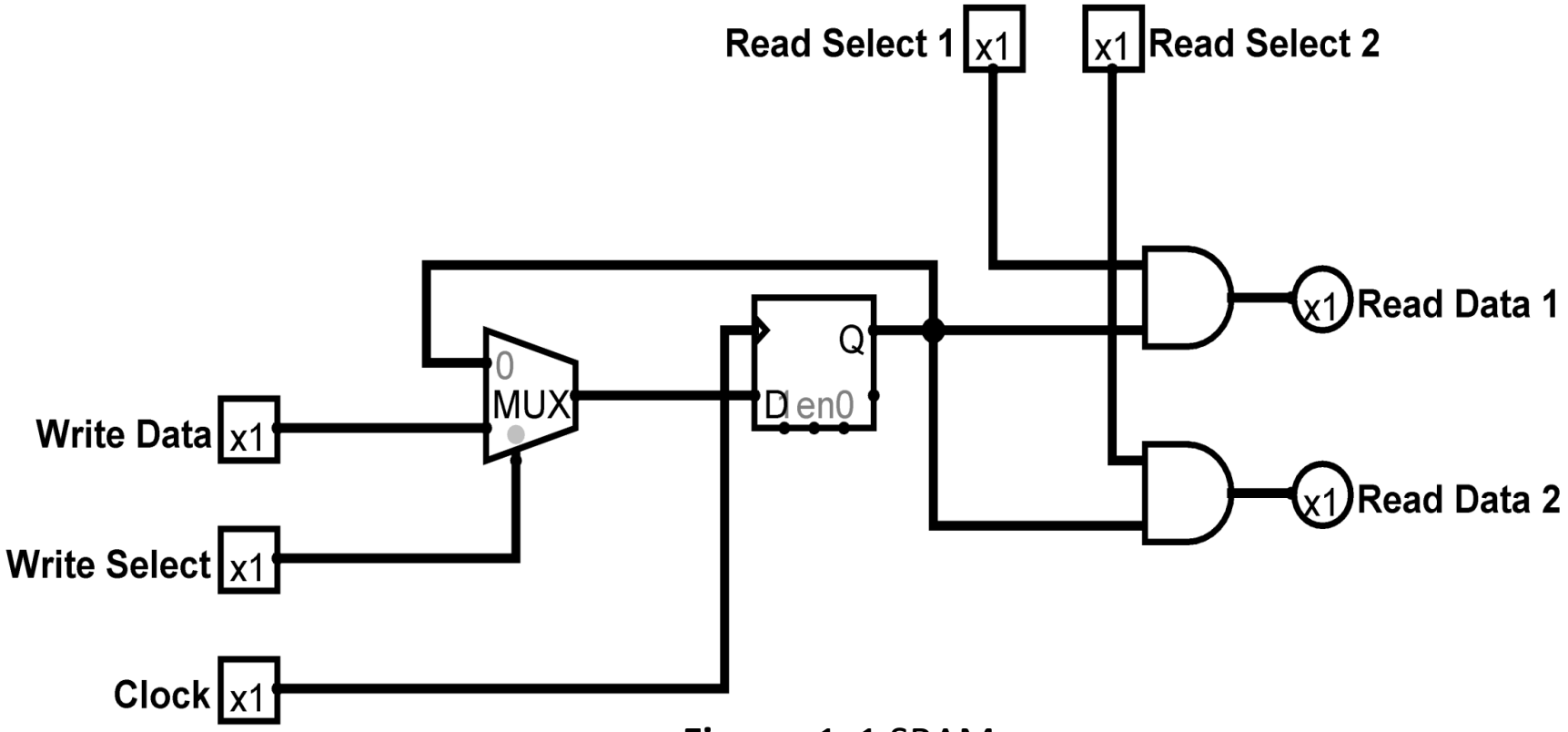


Figure: 1x1 SRAM

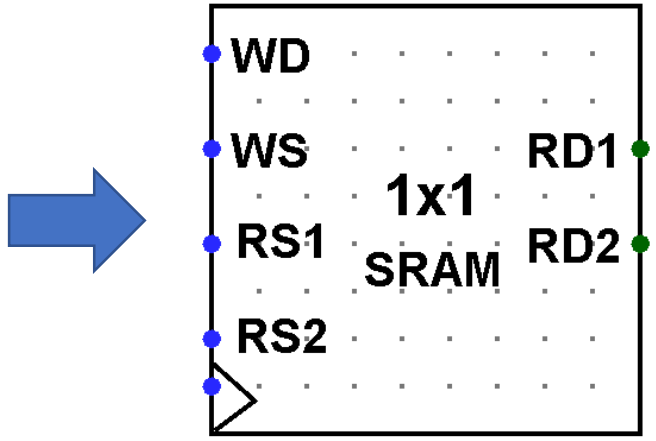


Figure: 1x1 SRAM Chip

Example: Main Memory

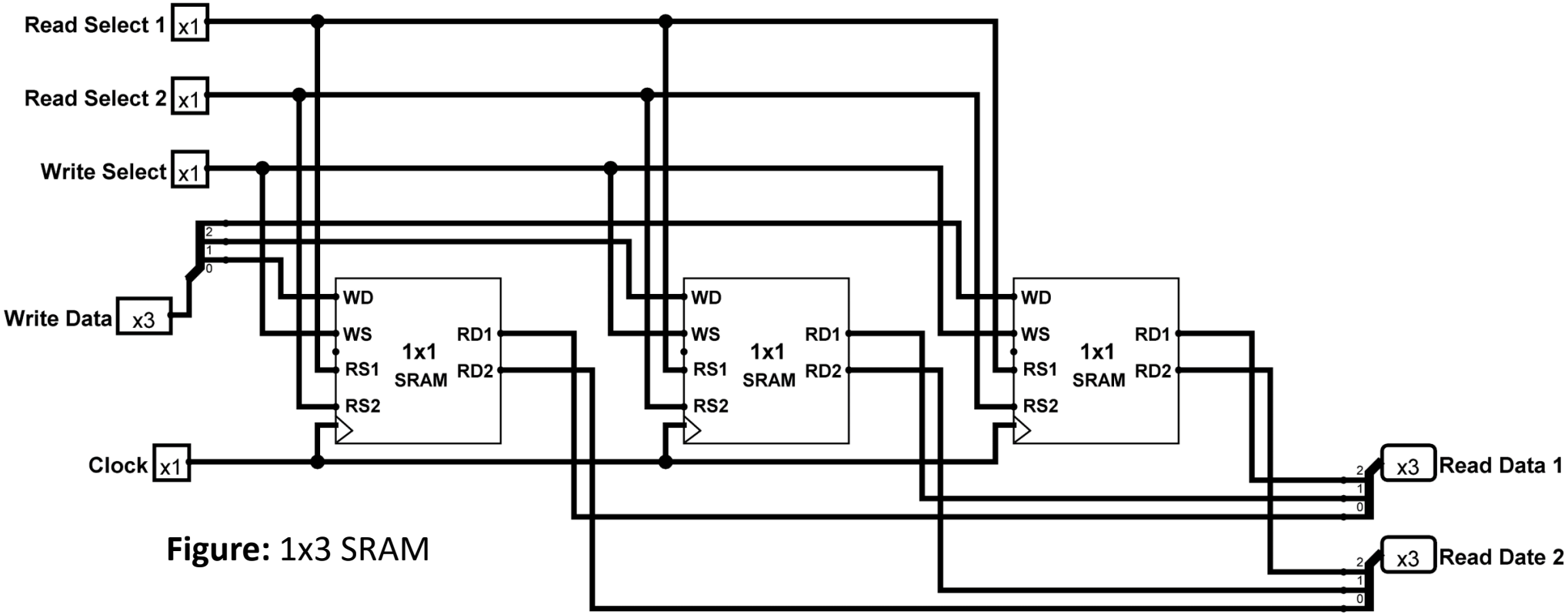


Figure: 1x3 SRAM

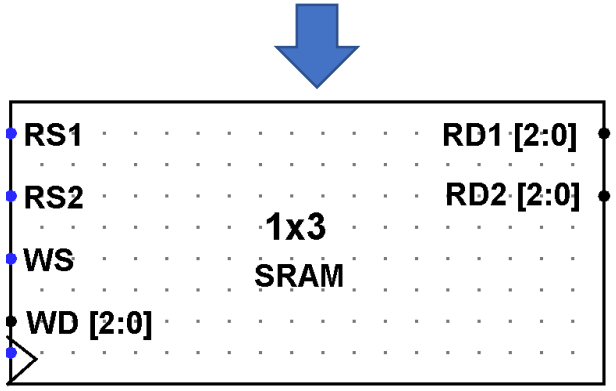


Figure: 1x3 SRAM Chip

Example: Main Memory

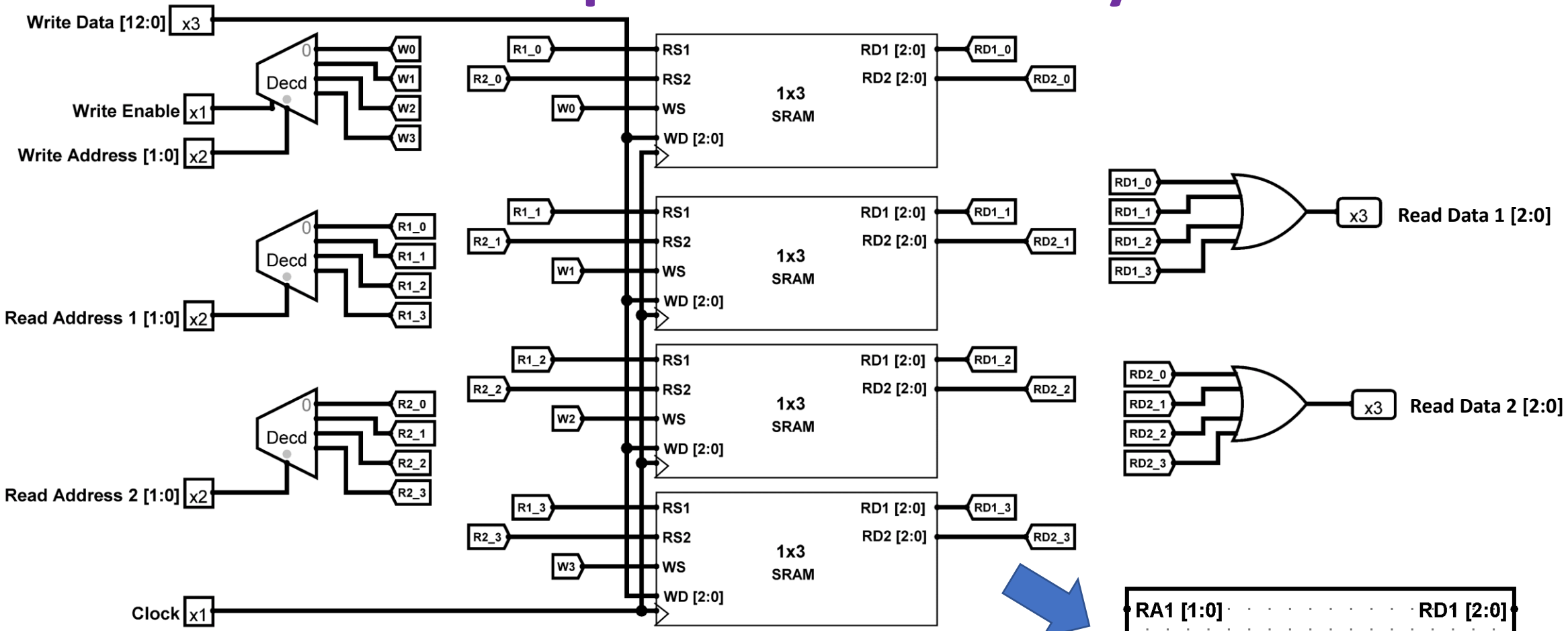


Figure: 4x3 SRAM

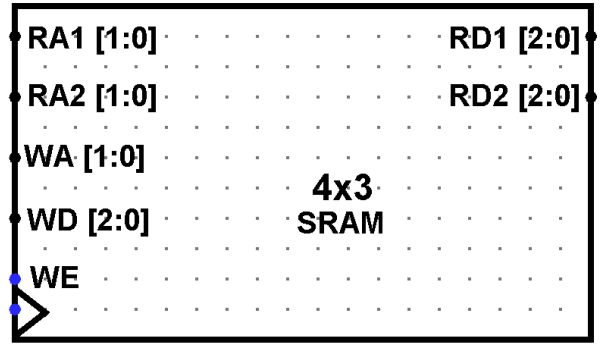


Figure: 4x3 SRAM Chip

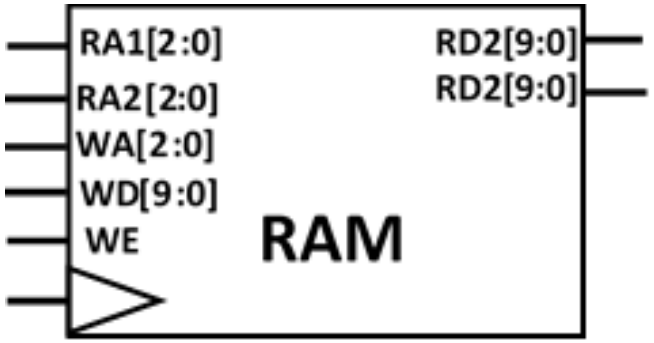
Exercises

1. Suppose, you want to store 1000 instructions in your main memory (RAM) and size of your ISA is 32 bit then

i. How many address lines will be needed in RAM?

ii. What should be the minimum word size of RAM so that instruction can be stored in one address?

2. Consider the following RAM chip and its contents:



Address	Instructions									
	9	8	7	6	5	4	3	2	1	0
000	0	1	0	1	0	0	1	0	0	1
001	1	0	1	1	0	0	0	1	0	1
010	0	0	0	0	0	0	0	0	0	0

i. What is the size and word size of RAM?

ii. If RA1 = 001 and RA2 = 000, then what will the value of RD1 and RD2?

iii. If WA = 000, WD = 111110000, WE = 1, RA1 = 001, RA2 = 000, then what will be the value of RD1 and RD2 after next clock pulse?

iv. Design this RAM chip.

Exercises

3. Design/Implement 10x10 RAM based on following configurations:

i. Single port
ii. Dual ports
iii. Single port with write operation
iv. Dual port with write operation.

Thank You ☺