Memory (1A)

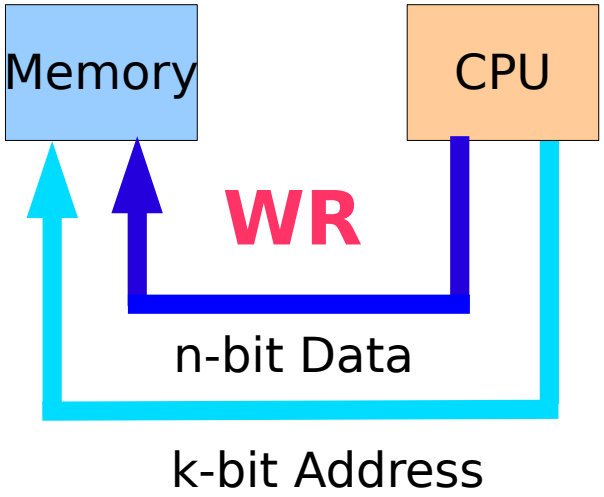
Young Won Lim
4/11/15

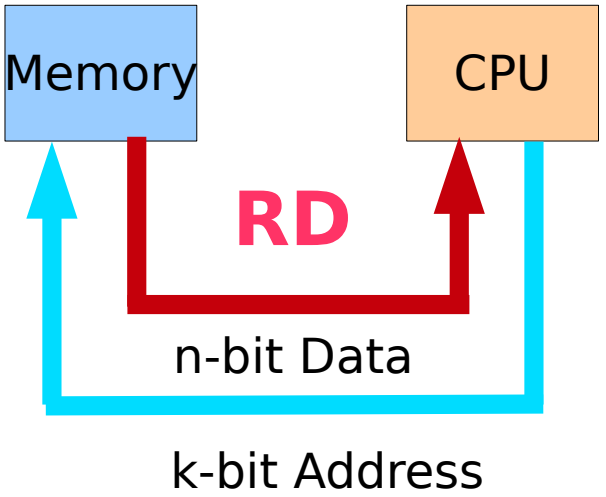
Please send corrections (or suggestions) to youngwlim@hotmail.com.

This document was produced by using OpenOffice.

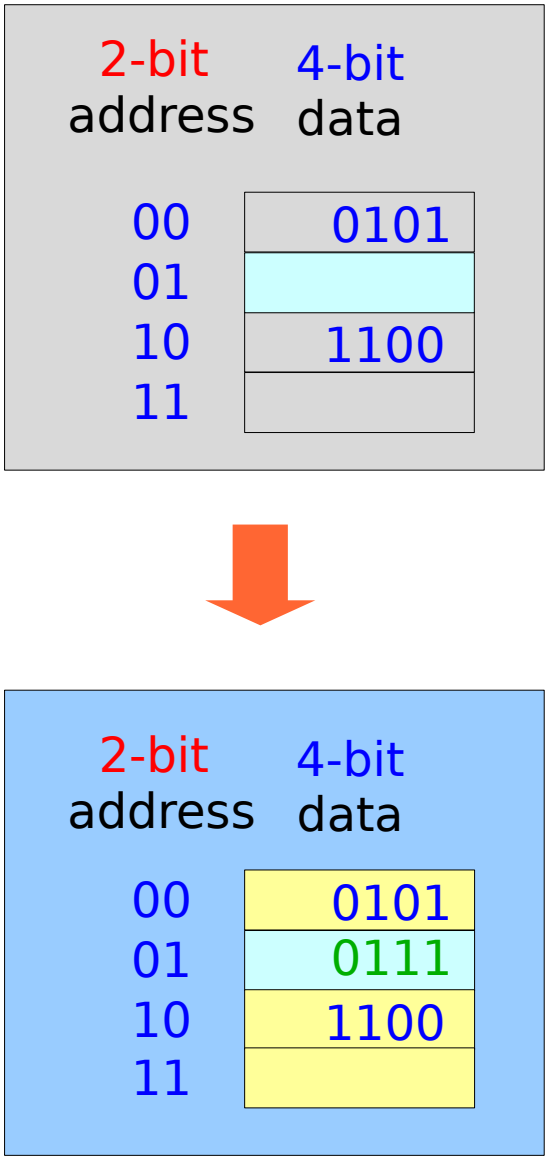
Young Won Lim
4/11/15

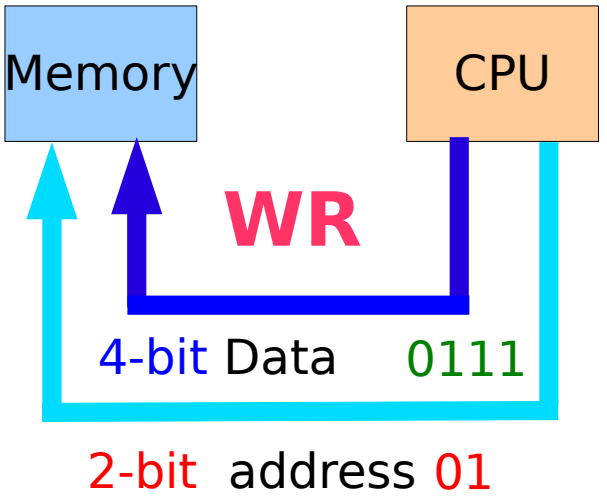
Memory Access Operations





A Memory Write Example



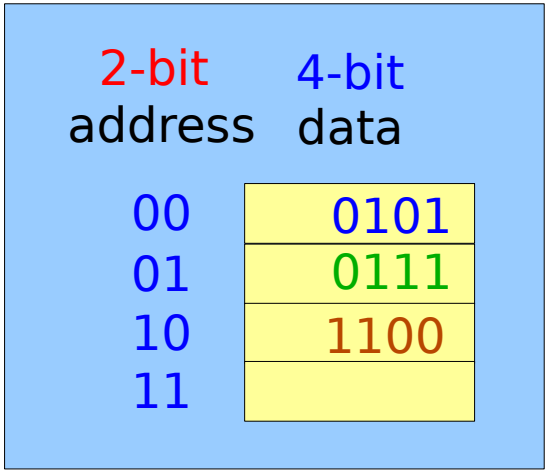


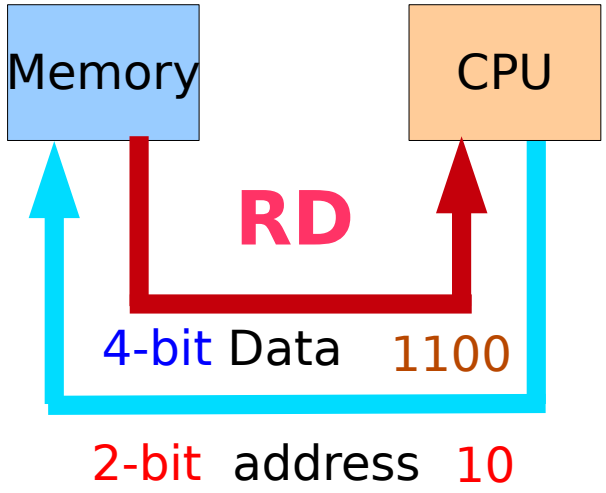
WR

Address: 01

Data: 0111

A Memory Read Example



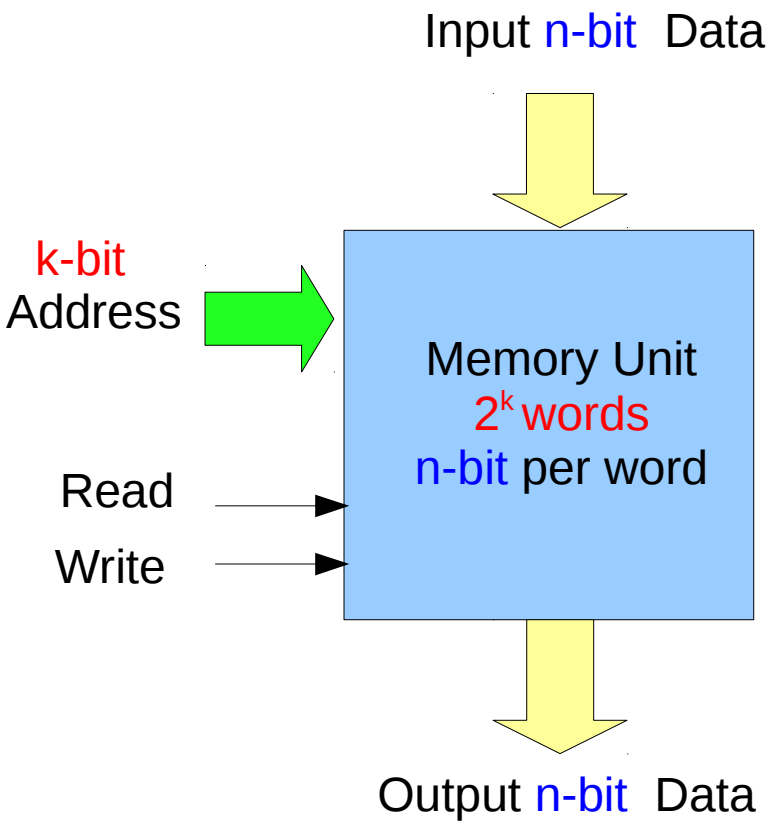


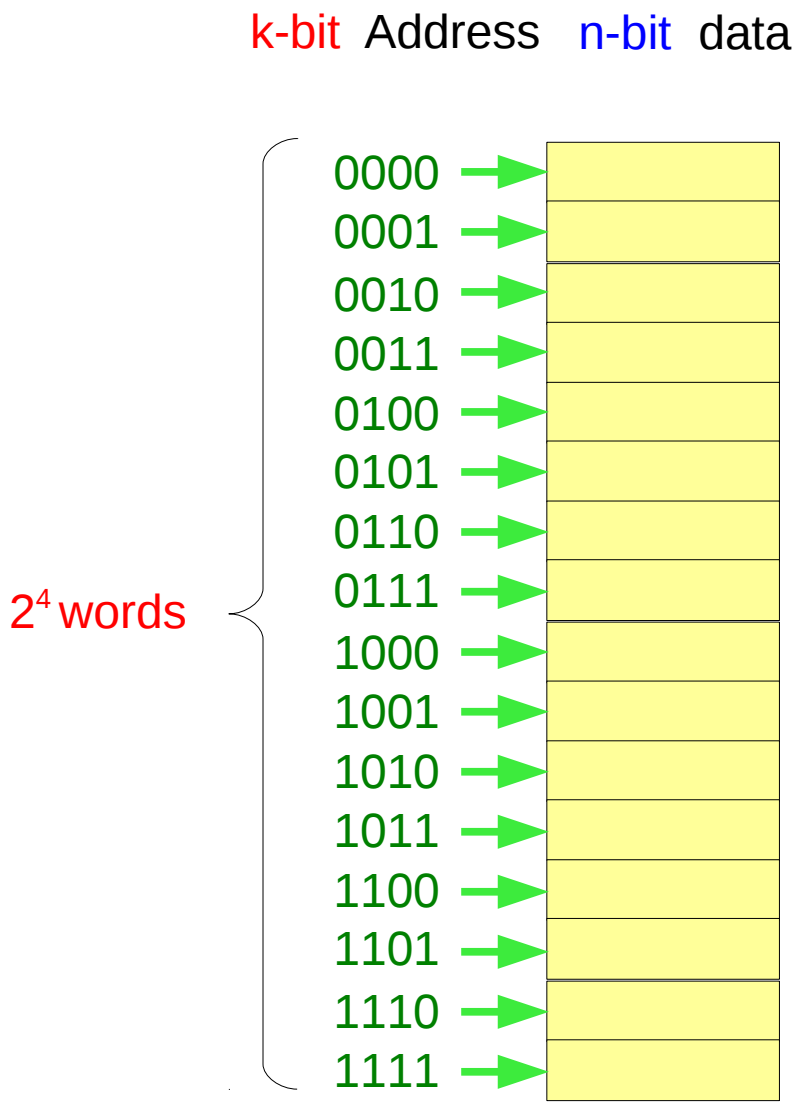
RD

Address: 10

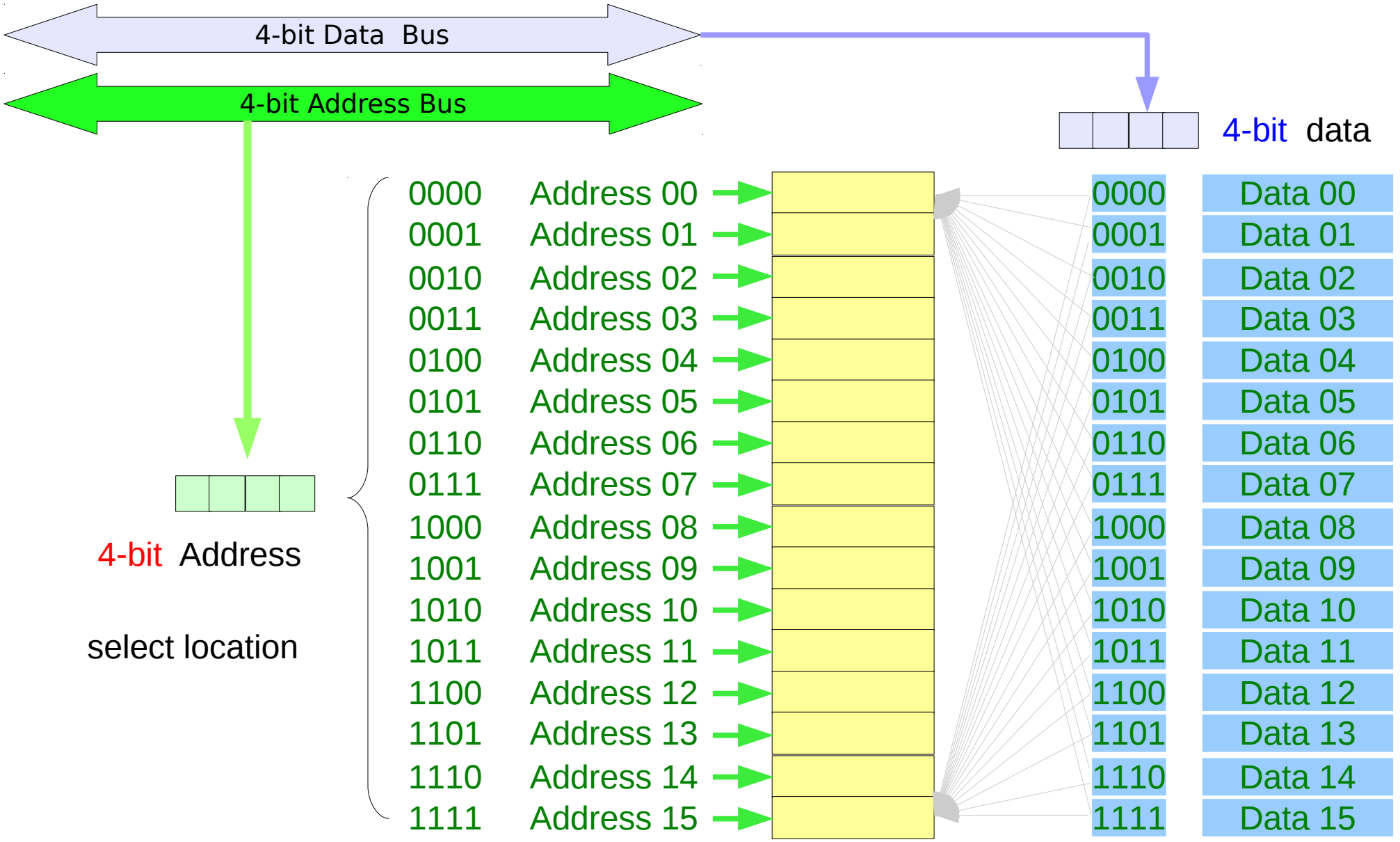
Data: ➡ 1100

A Memory and A Memory Map

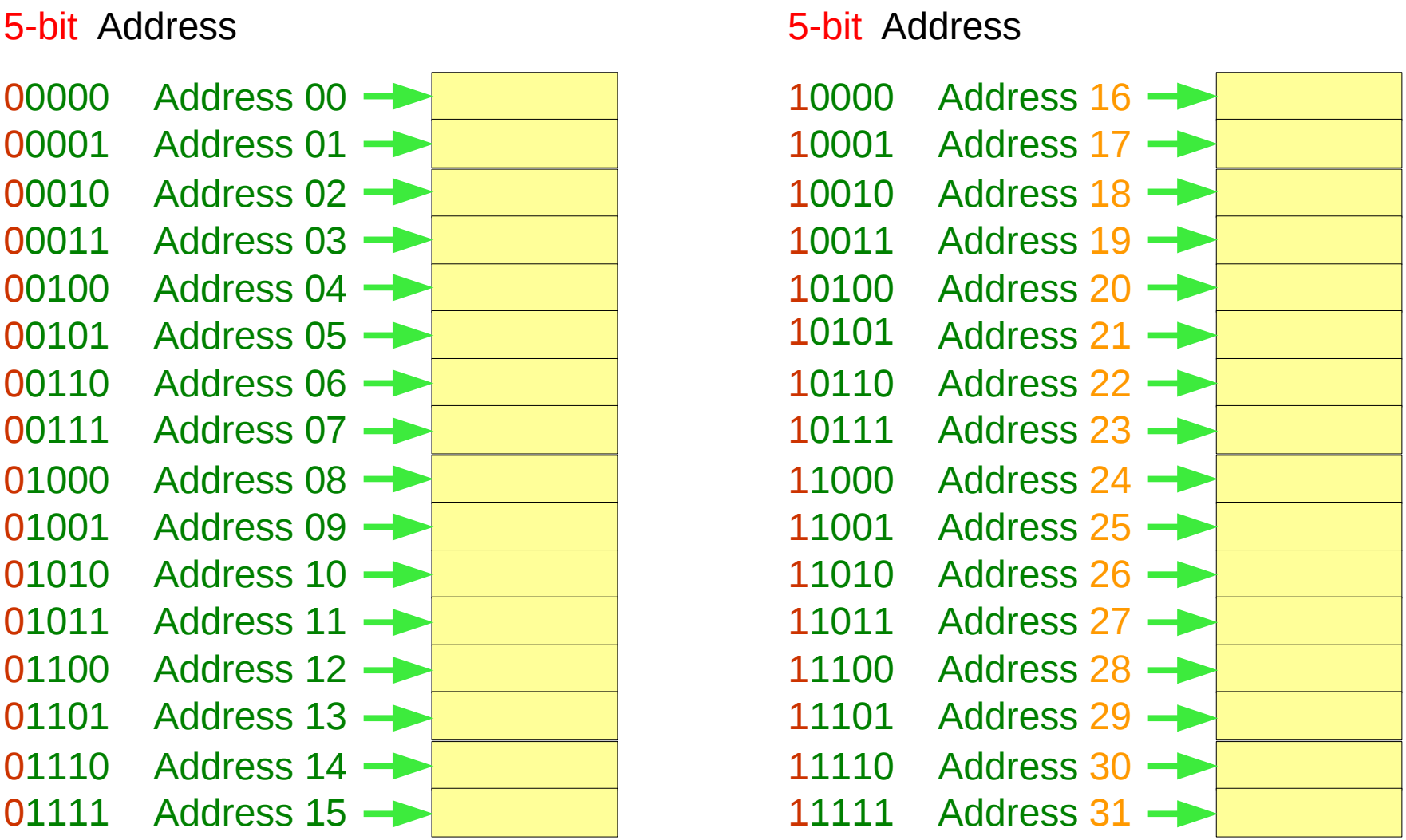




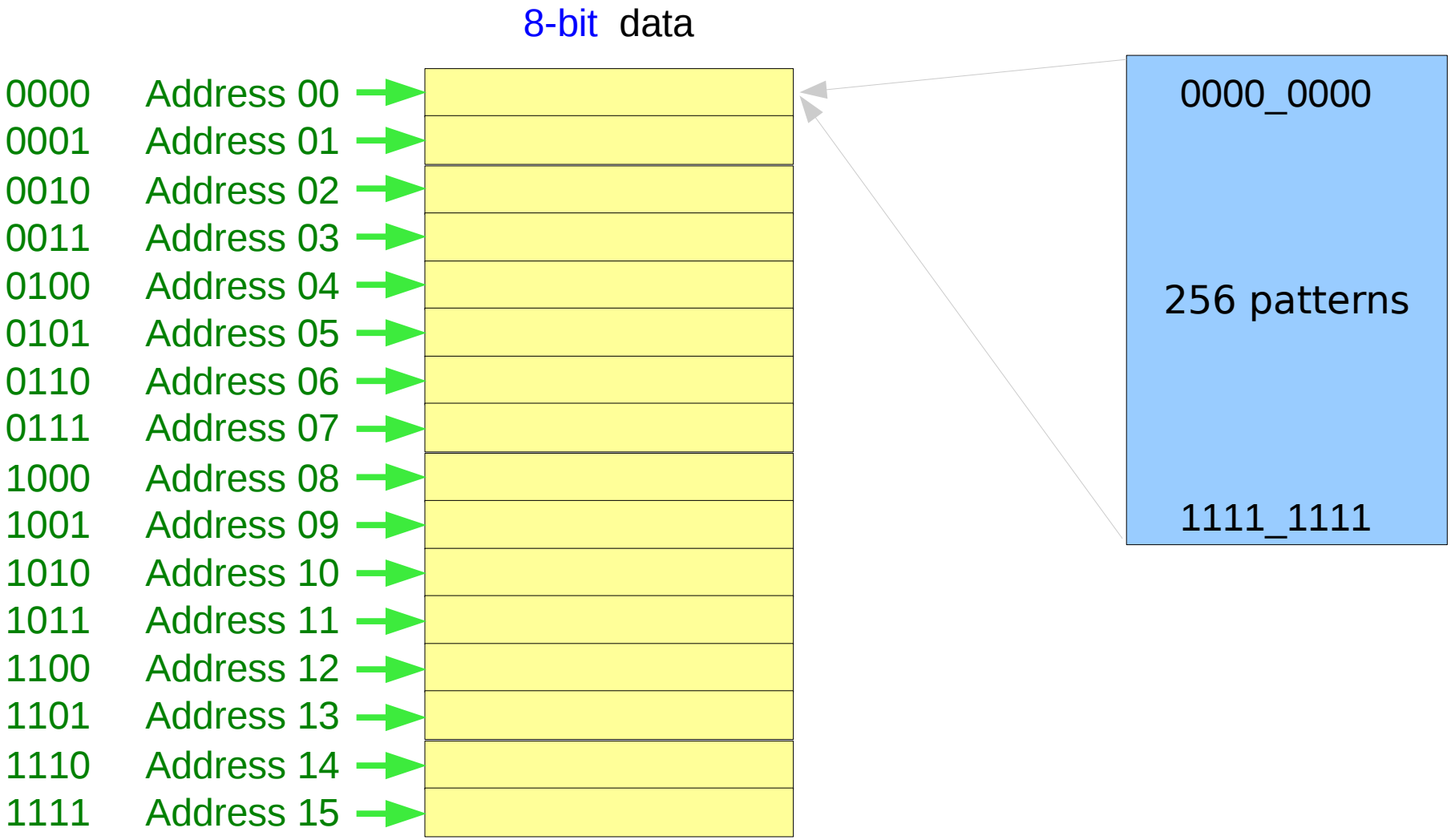
Address Bits and Data Bits



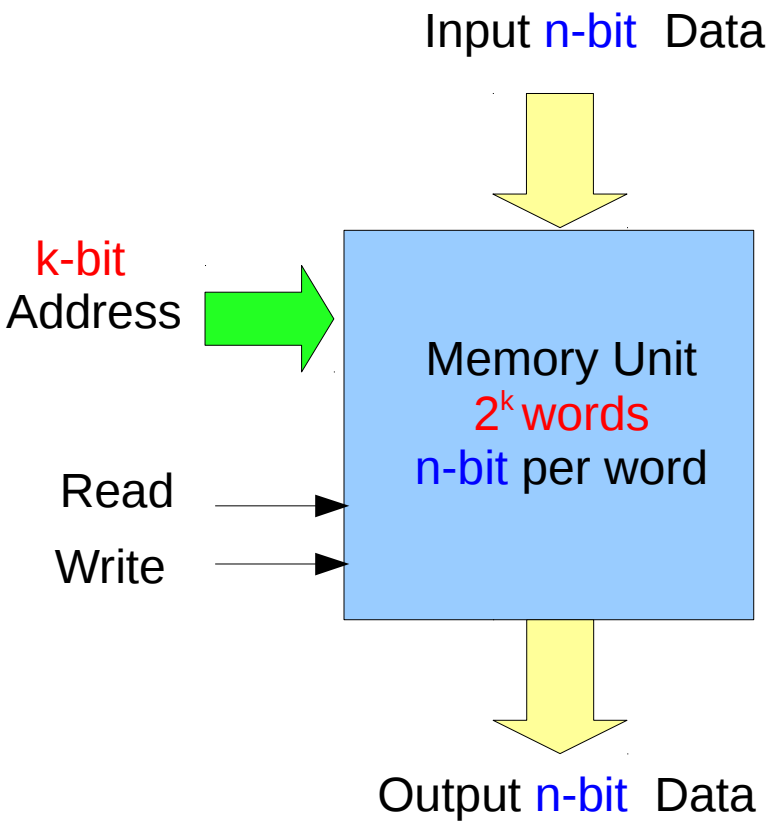
Increasing Memory Locations

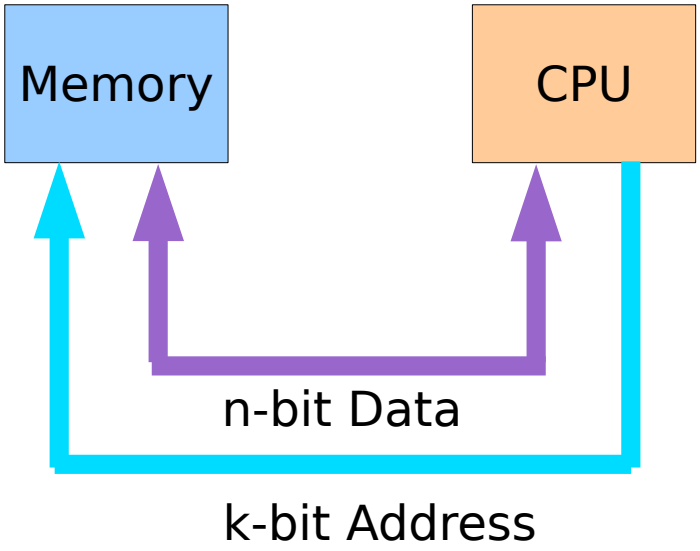


Increasing Memory Width

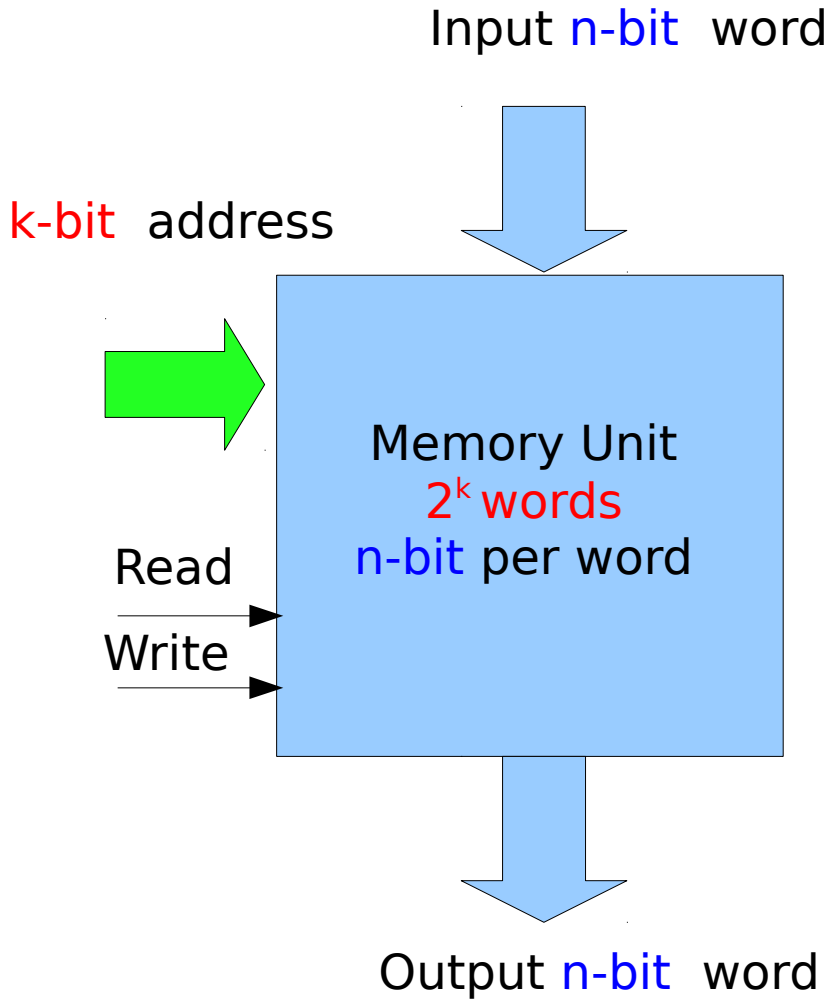


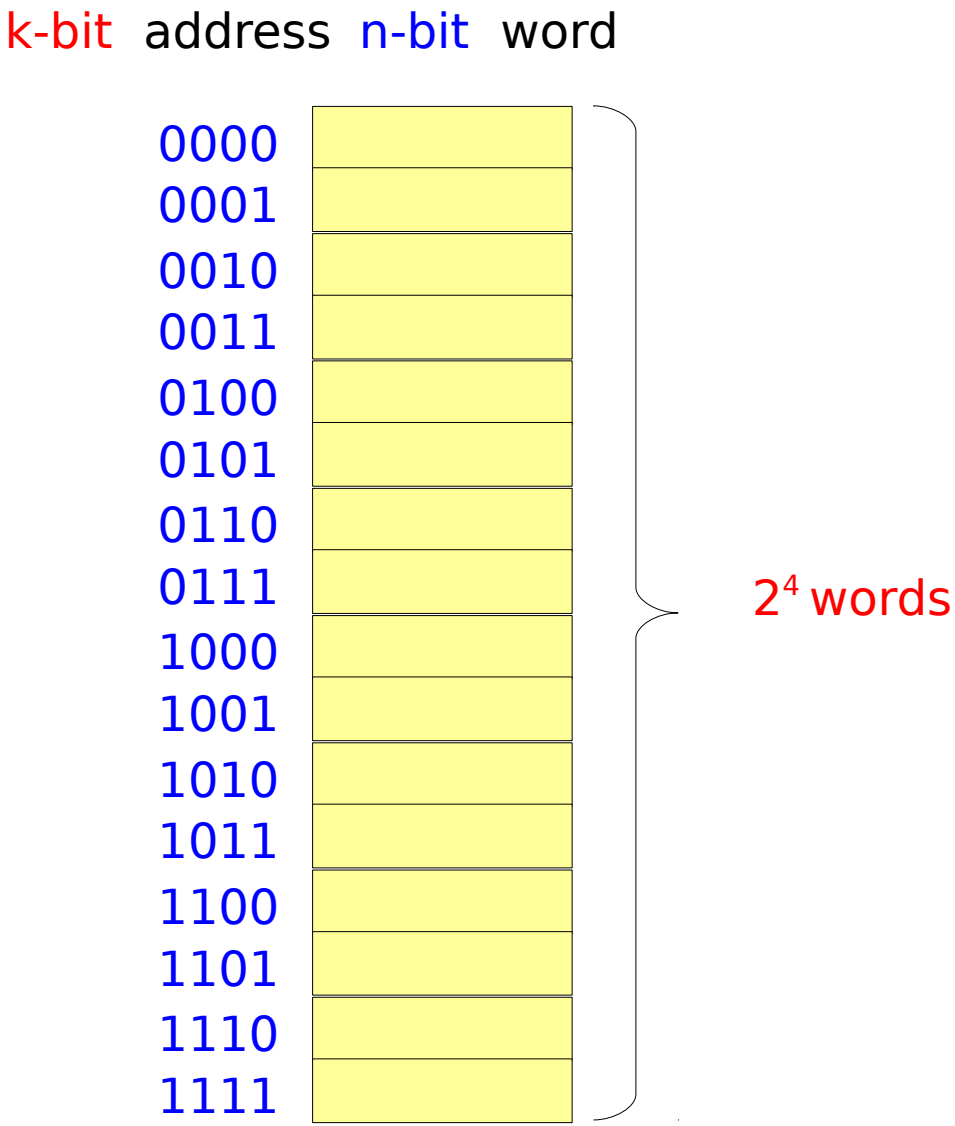
The Inputs and Outputs of a Memory



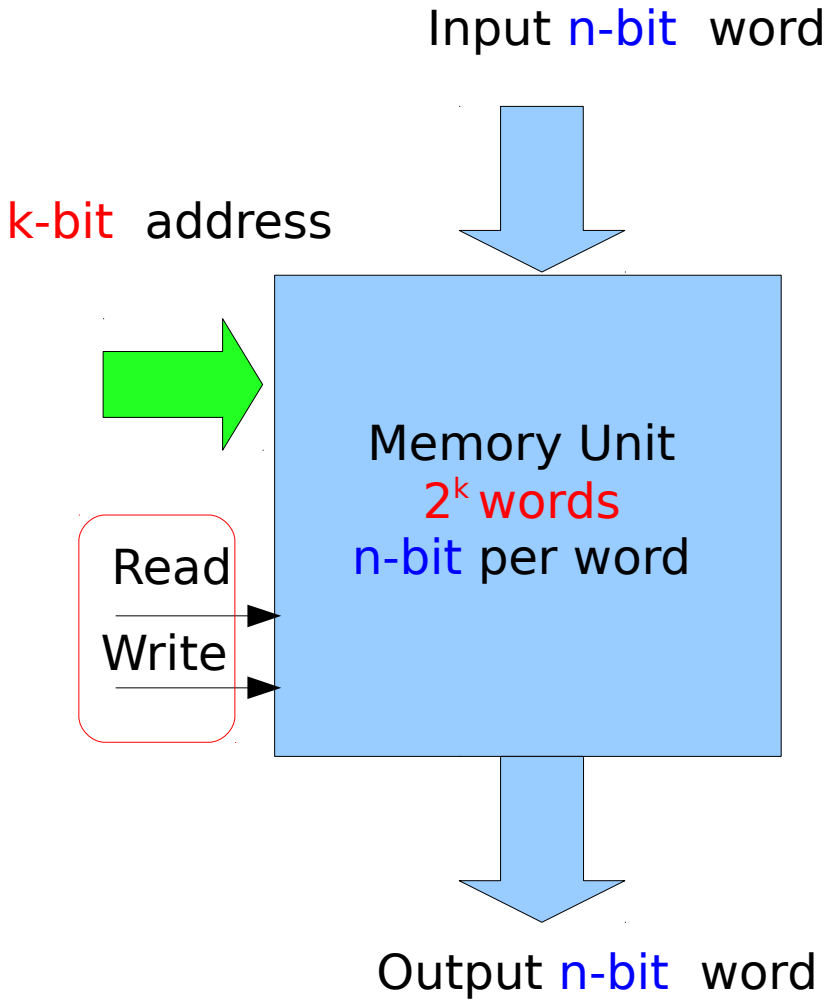


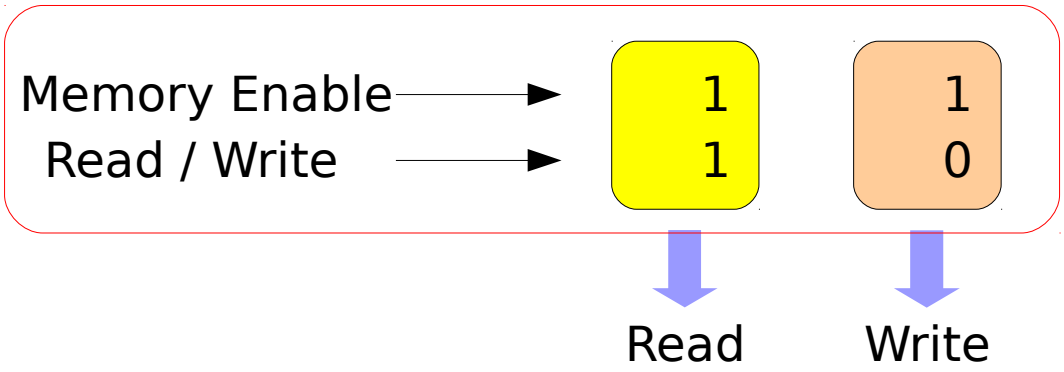
Block Diagram of a Memory Unit



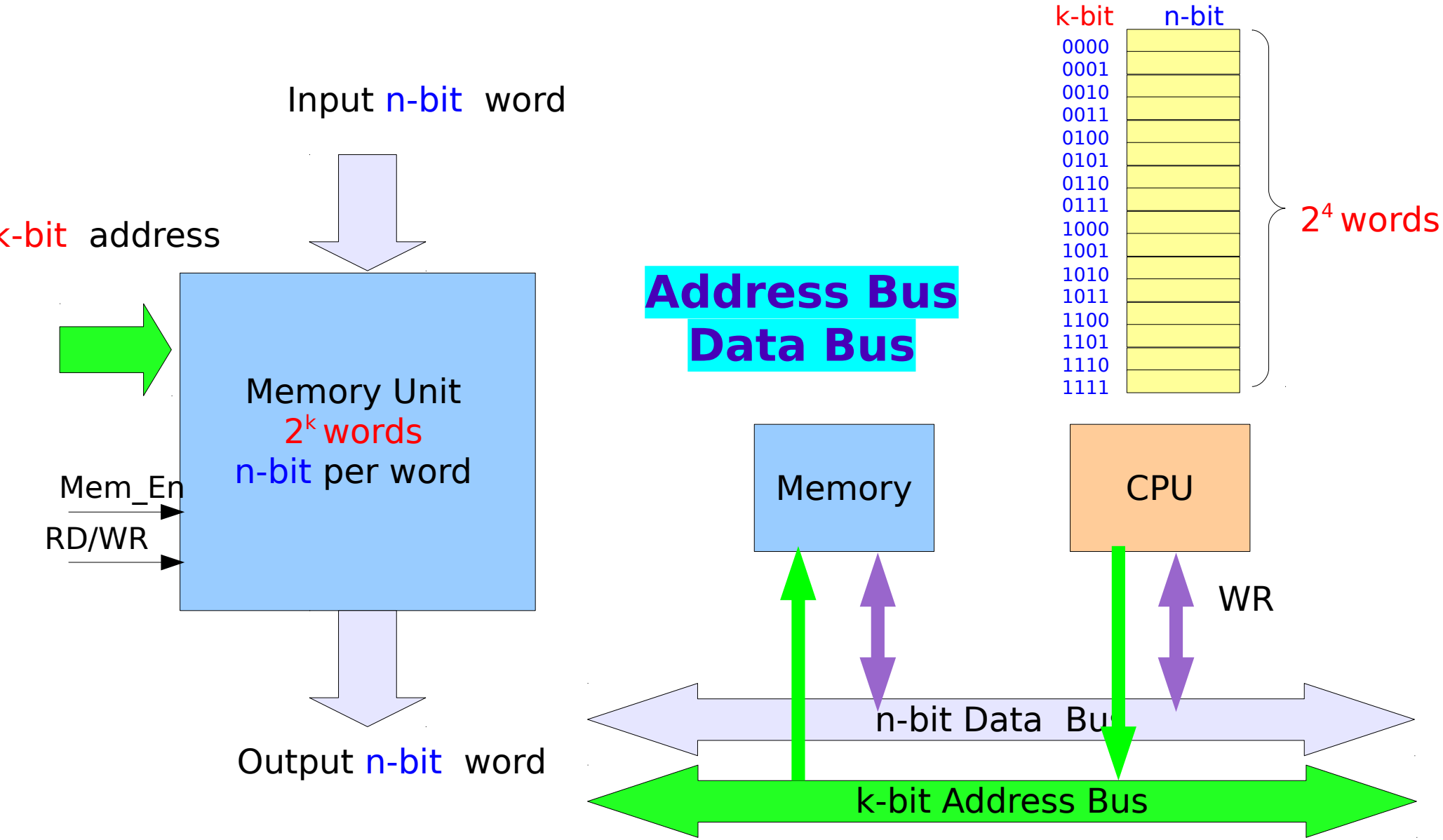


Memory Control Lines

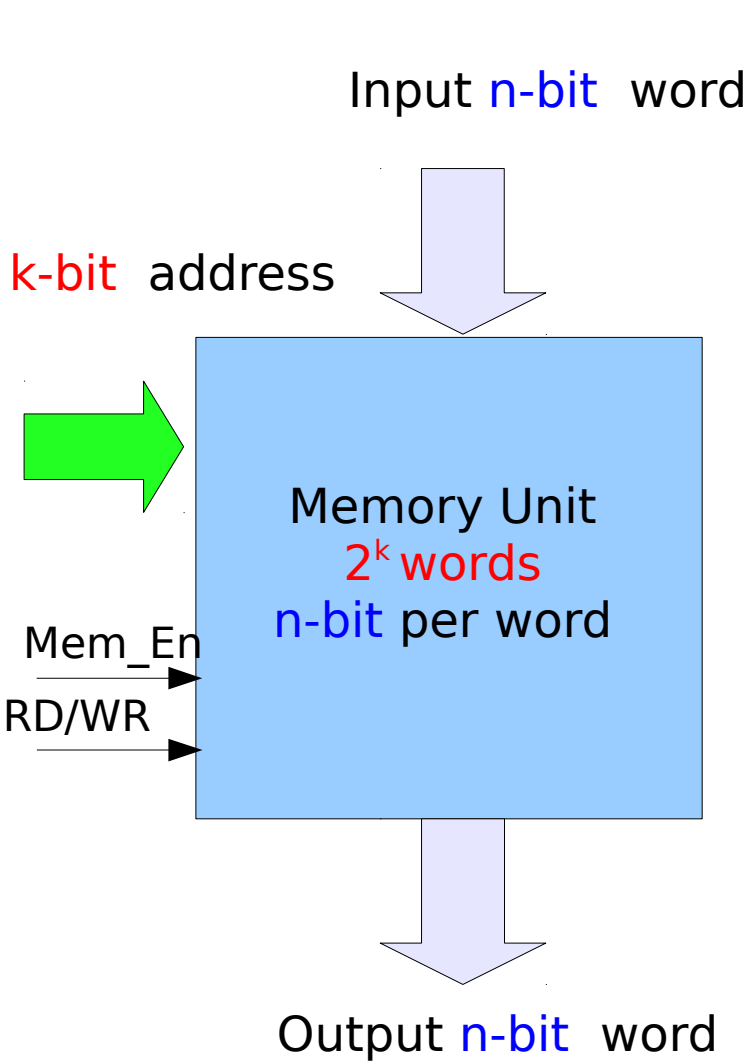


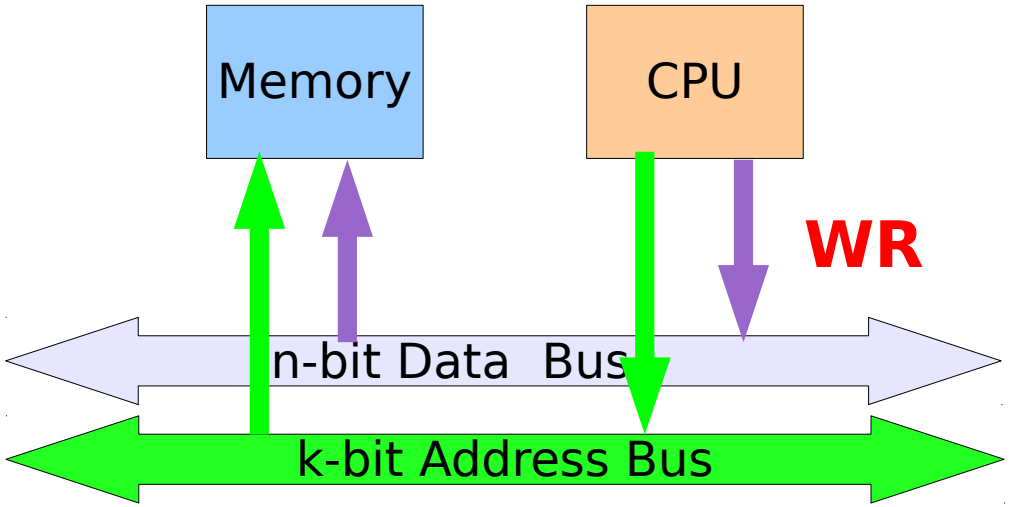


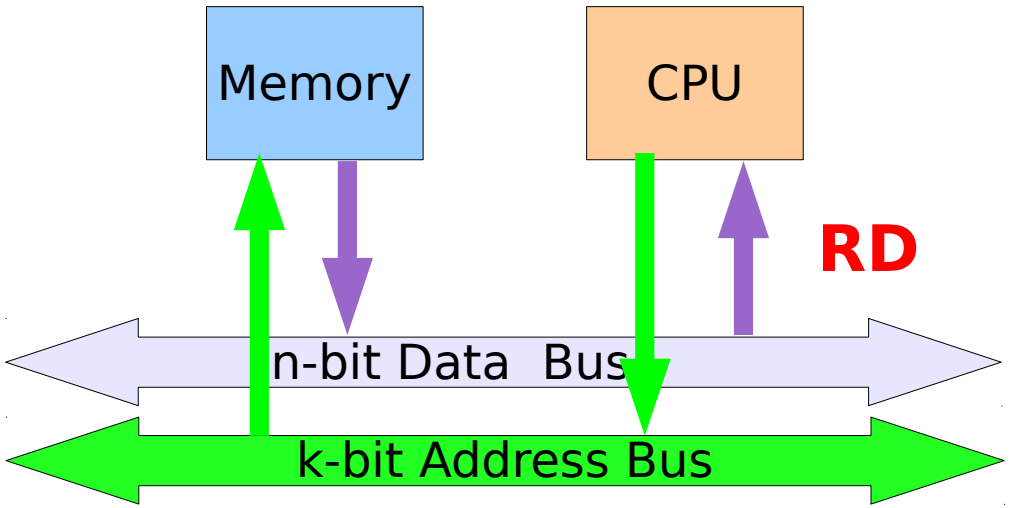
Address & Data Buses



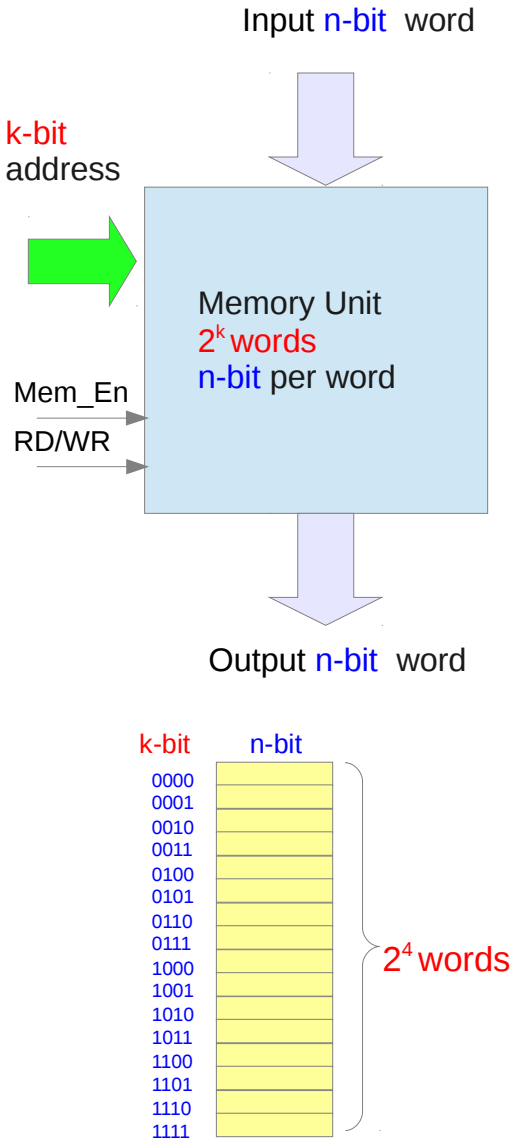
RD & WR Operations

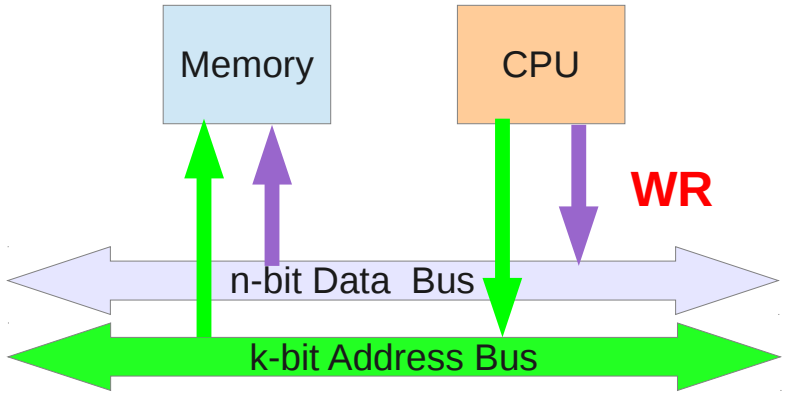




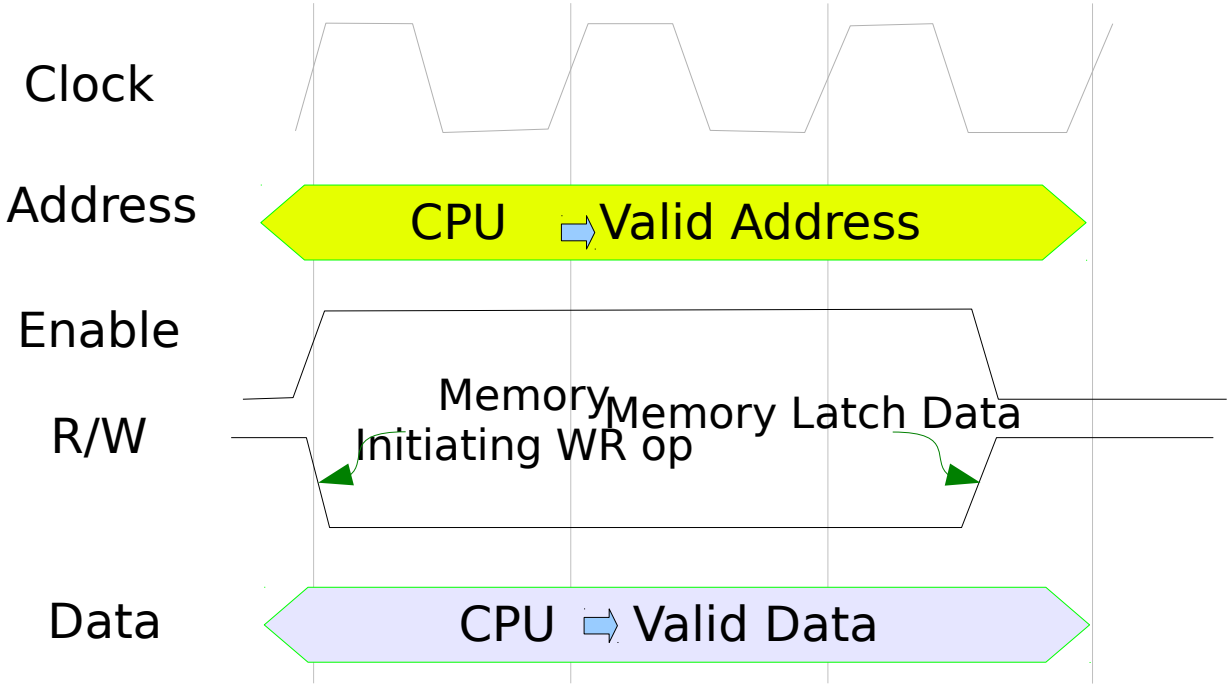


Memory Write Cycle

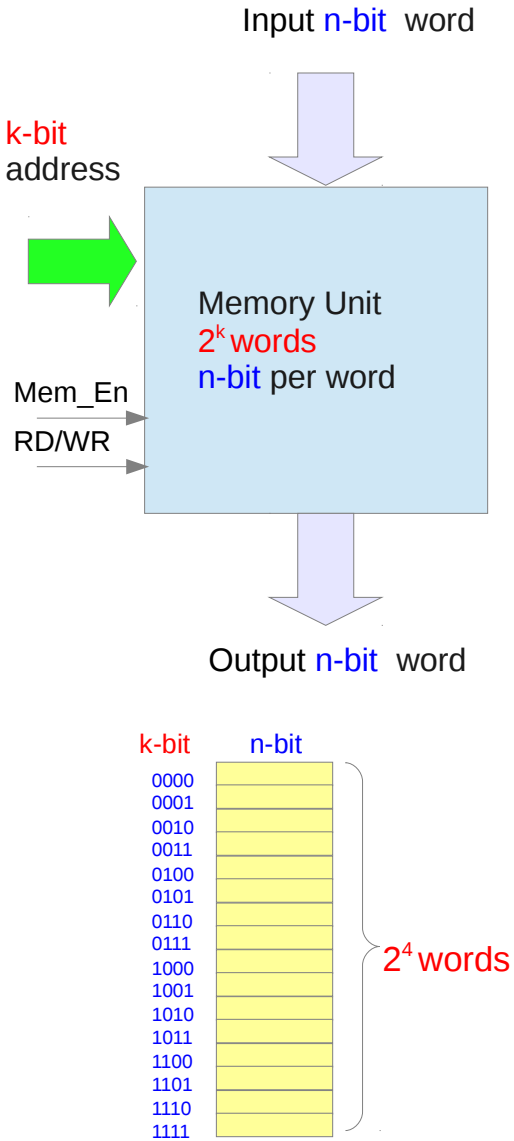


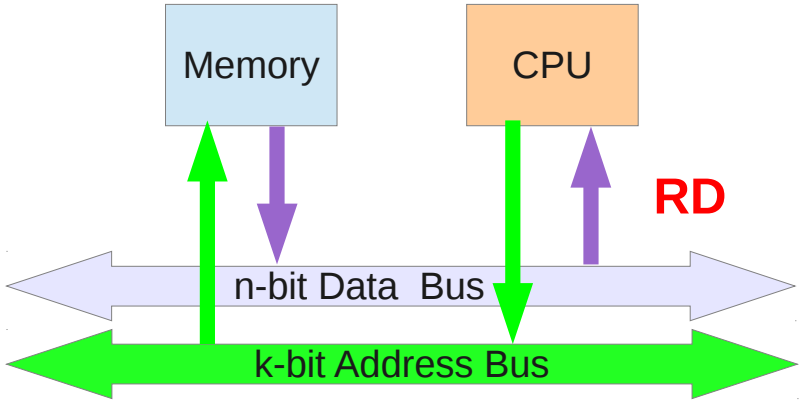


Usually, clock is seldom used. (shown for a reference)

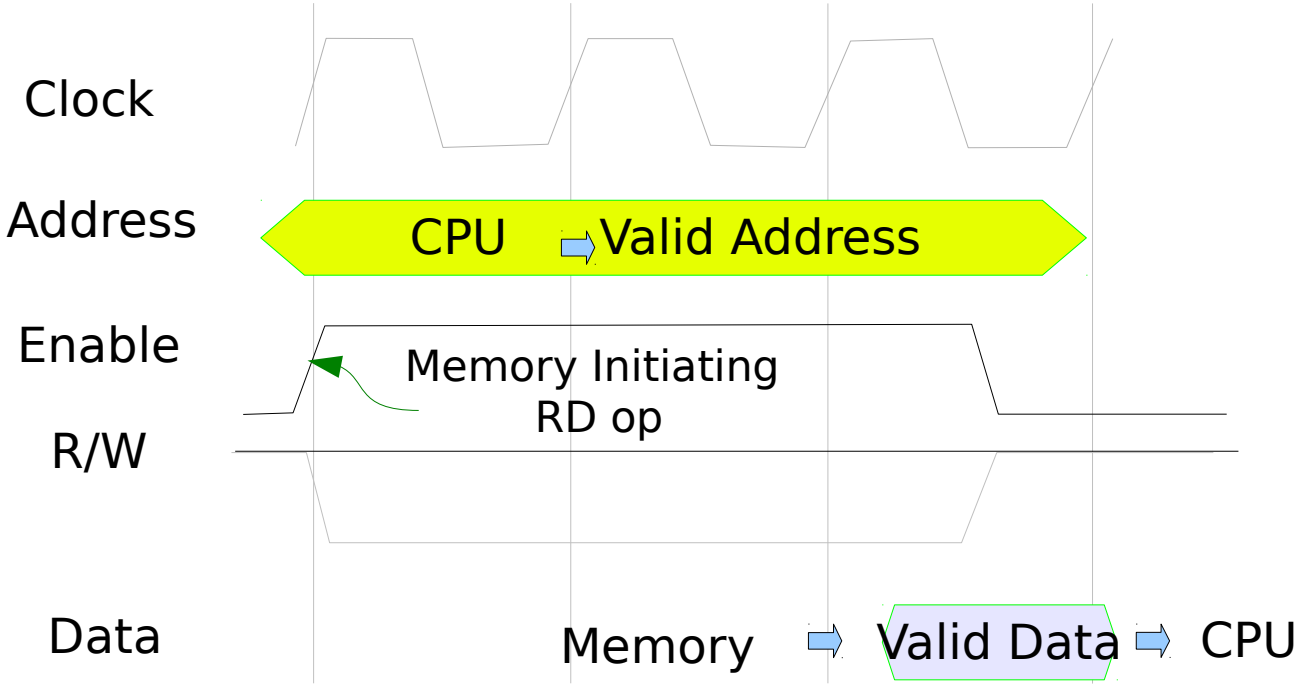


Memory Read Cycle

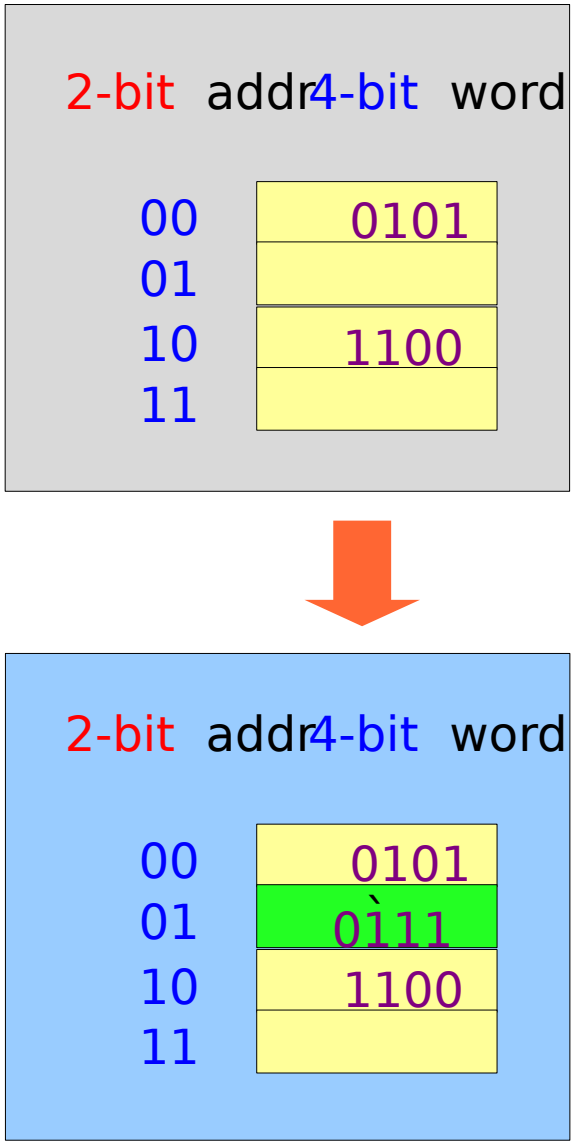


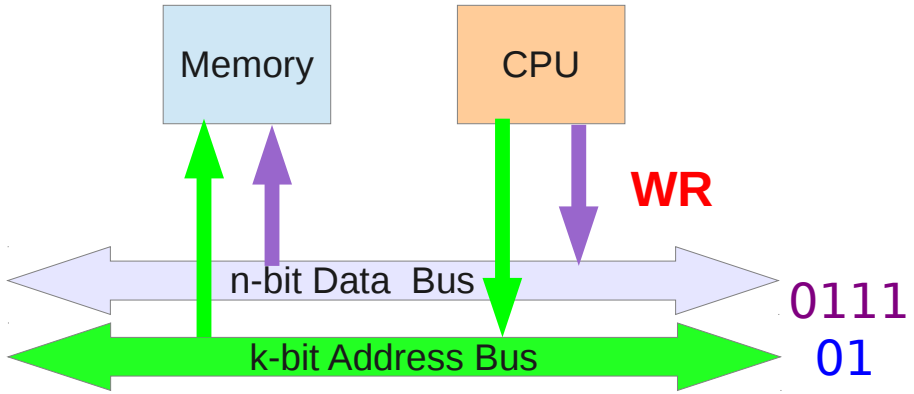


Usually, clock is seldom used. (shown for a reference)

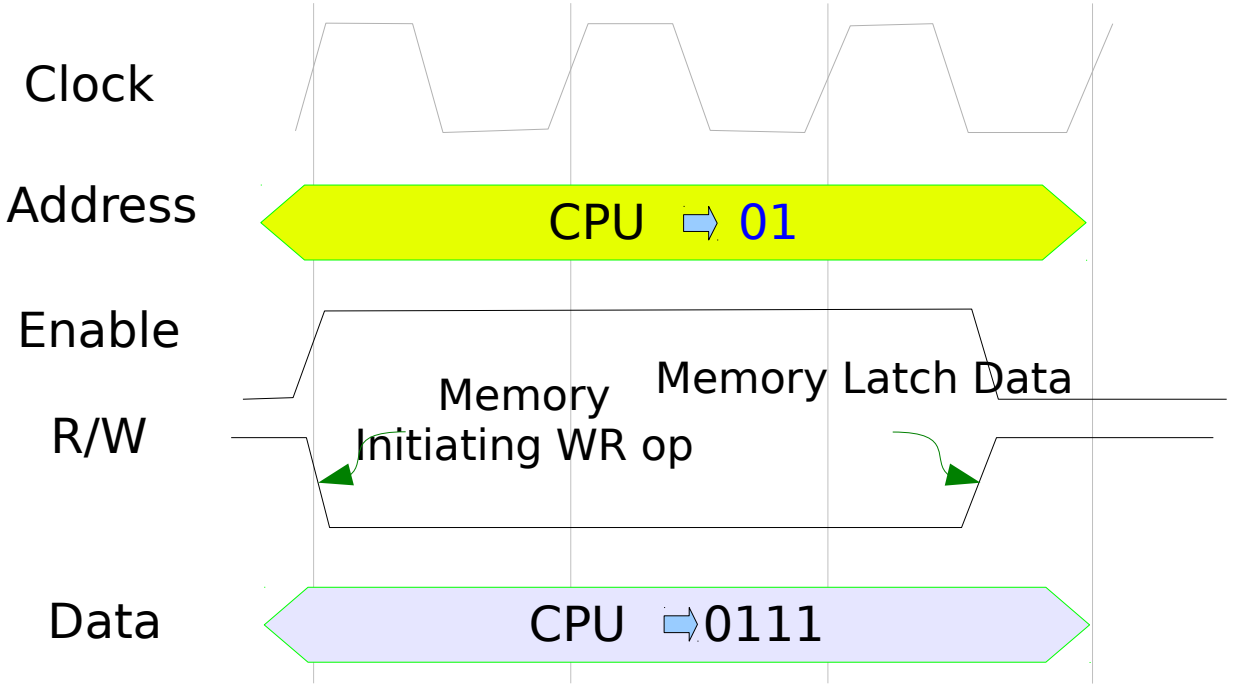


Memory Write Example

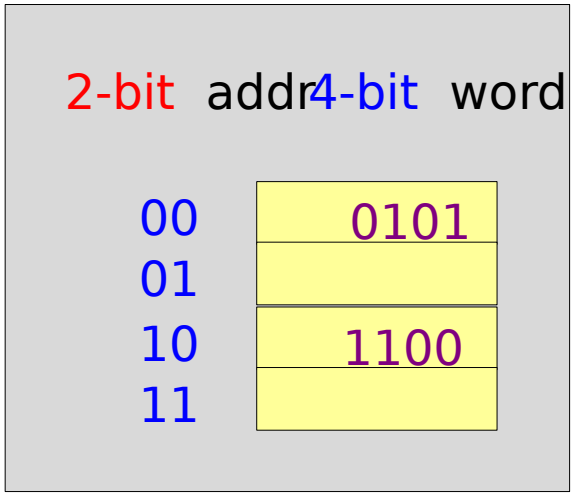


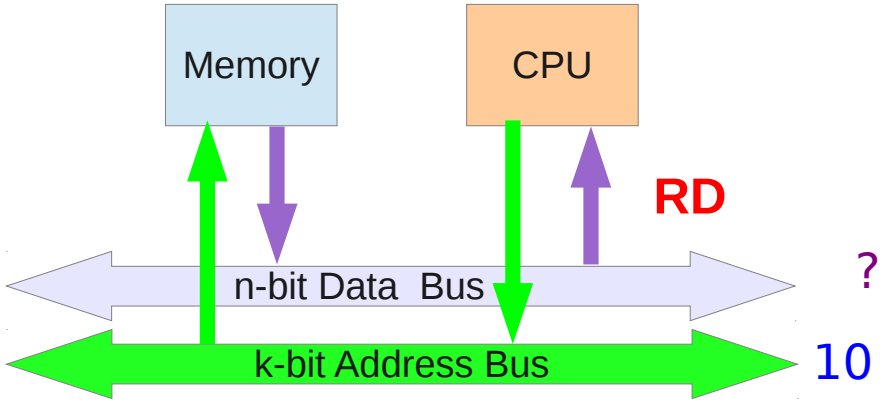


Usually, clock is seldom used. (shown for a reference)

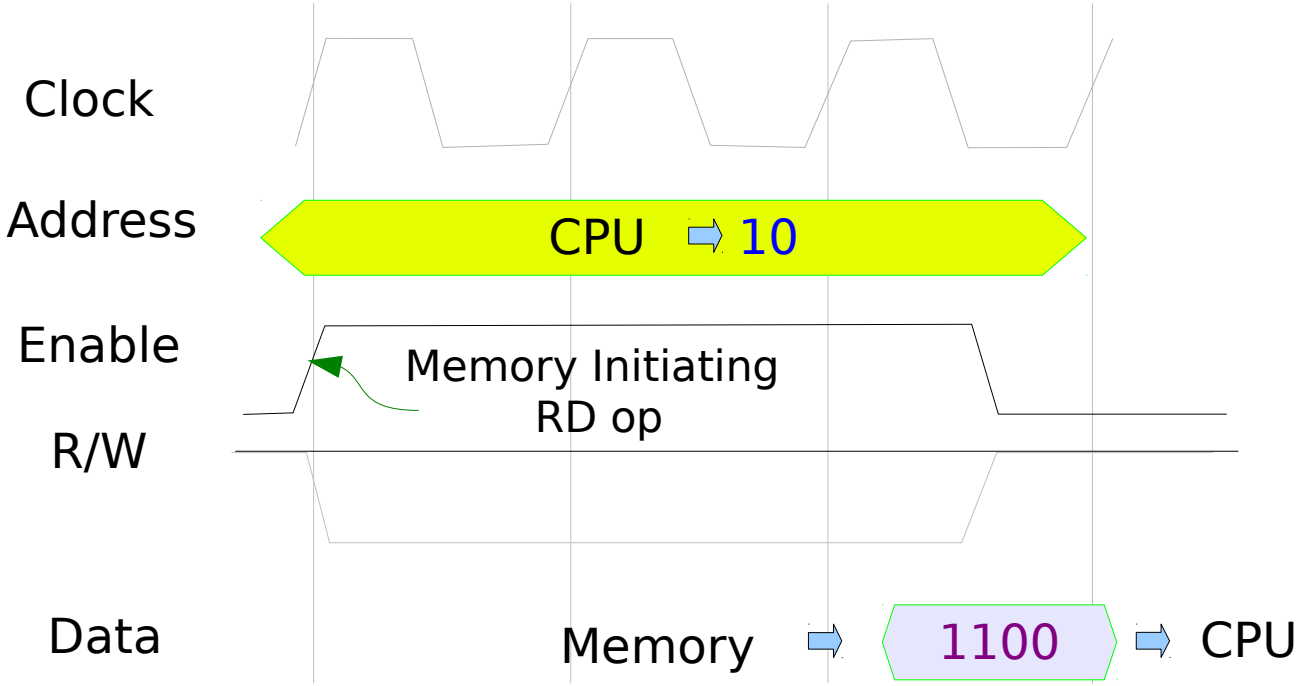


Memory Read Example

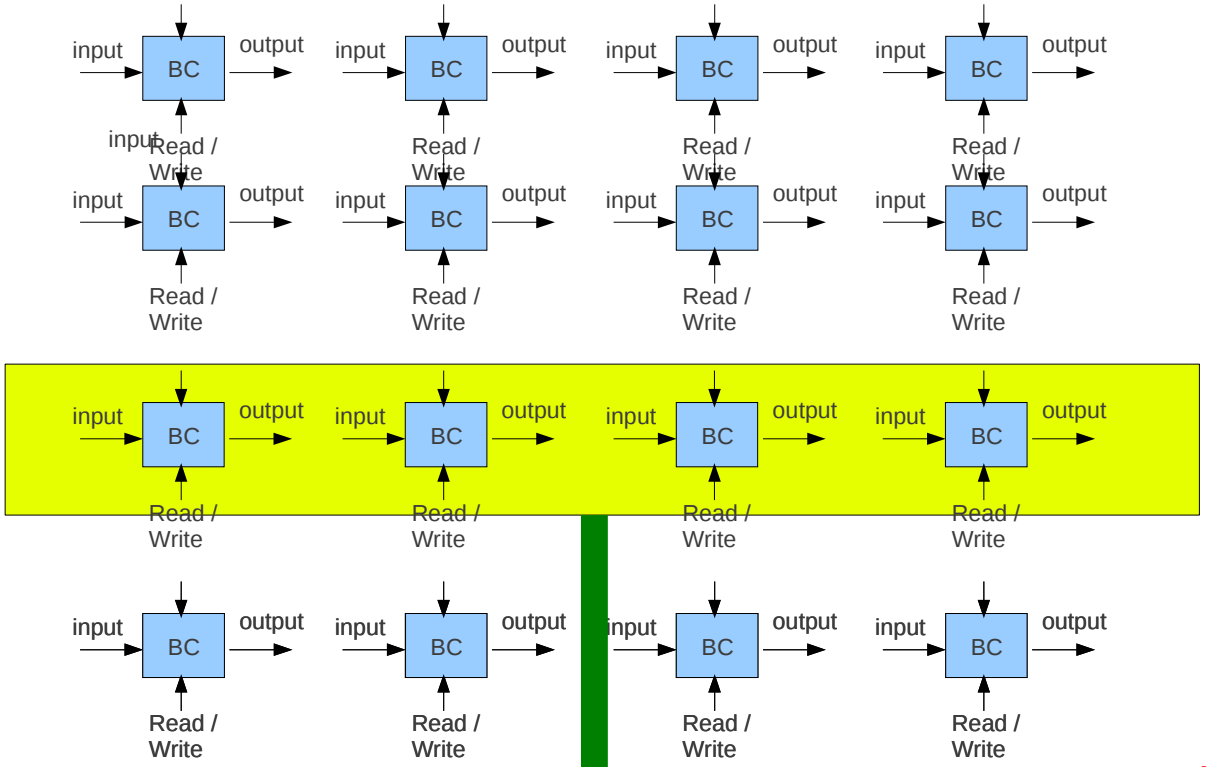




Usually, clock is seldom used. (shown for a reference)

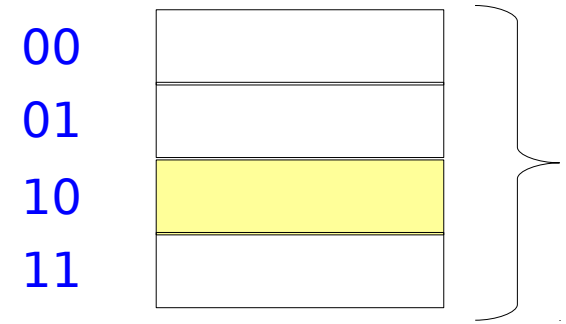


Selecting a Word

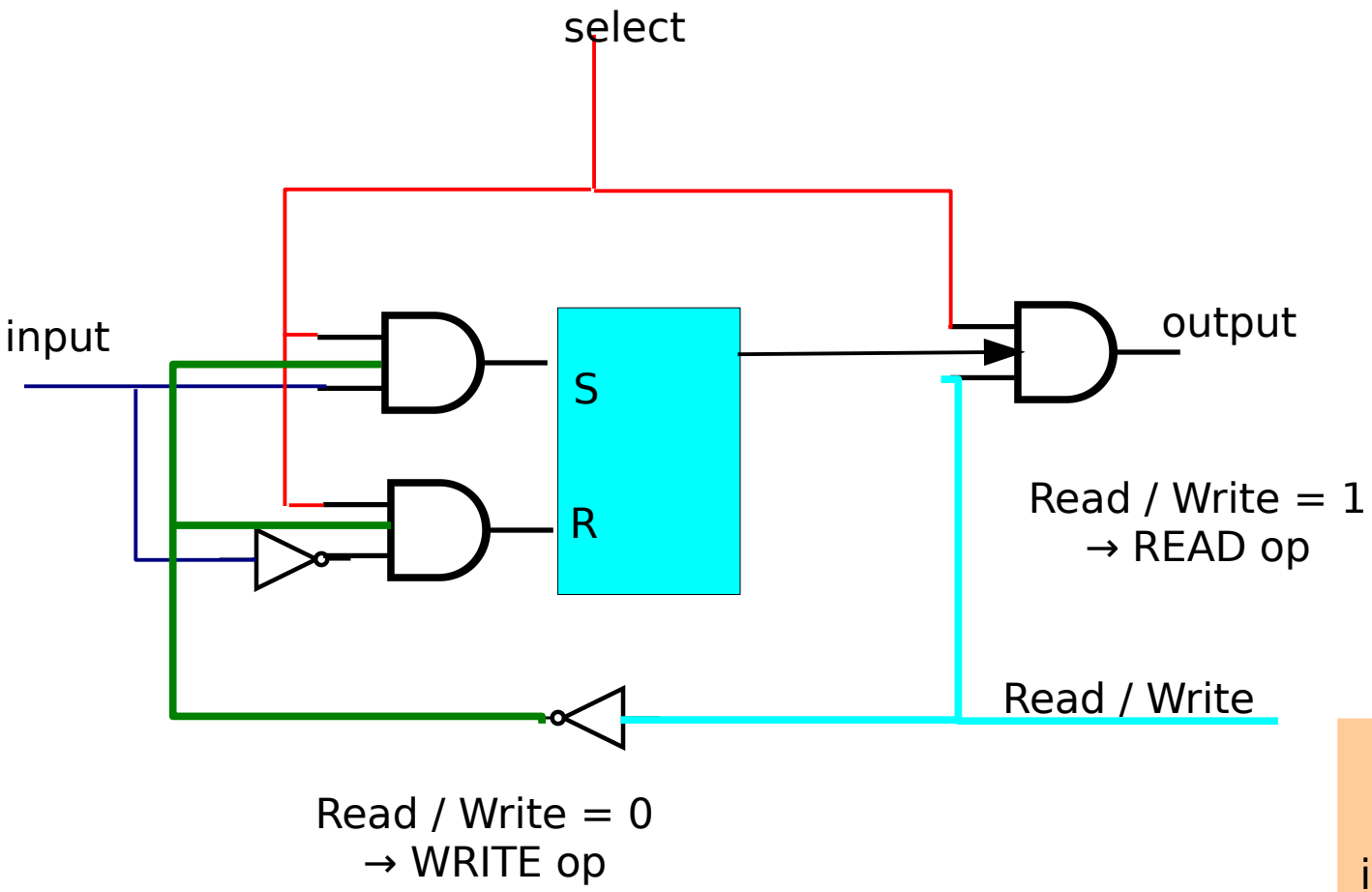


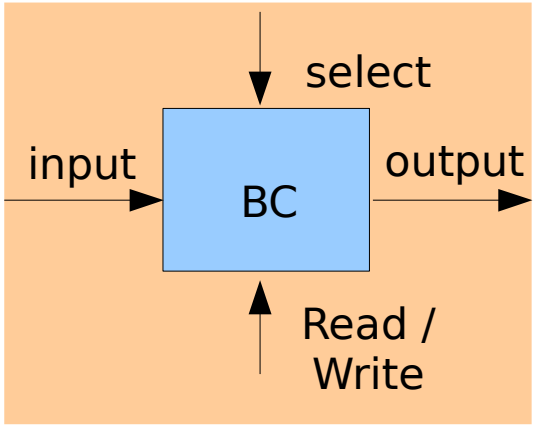
selected 4-bit word

2-bit address 4-bit word

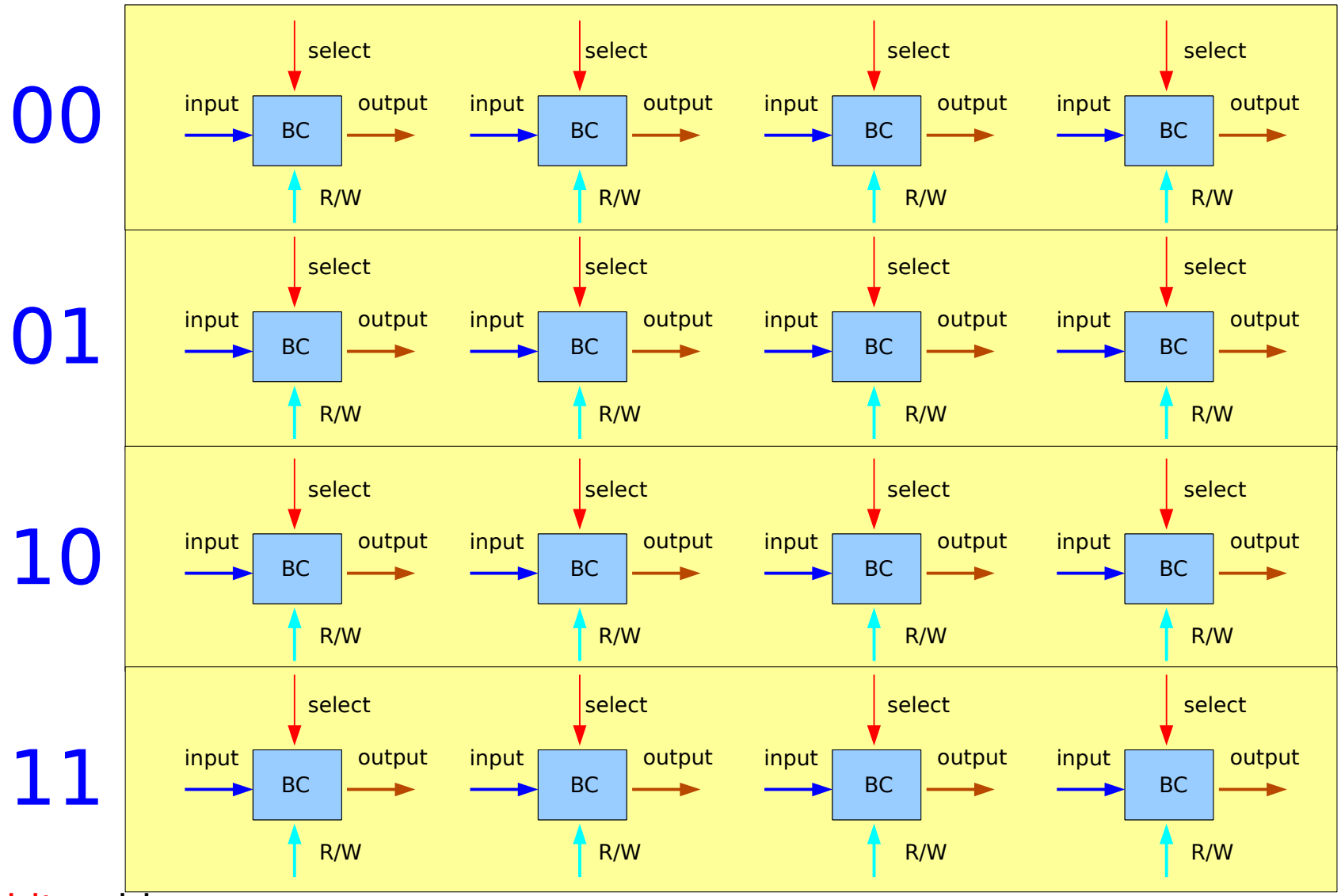


Selecting a Word

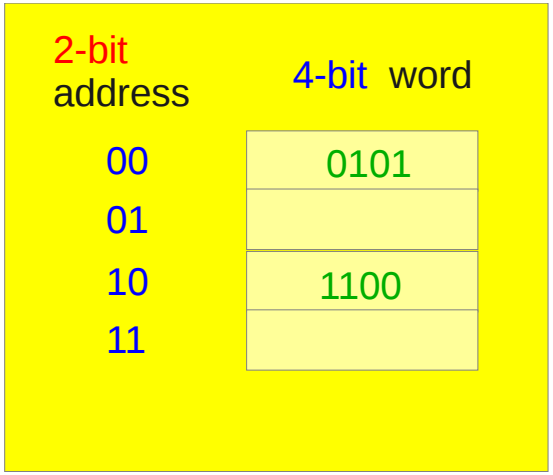


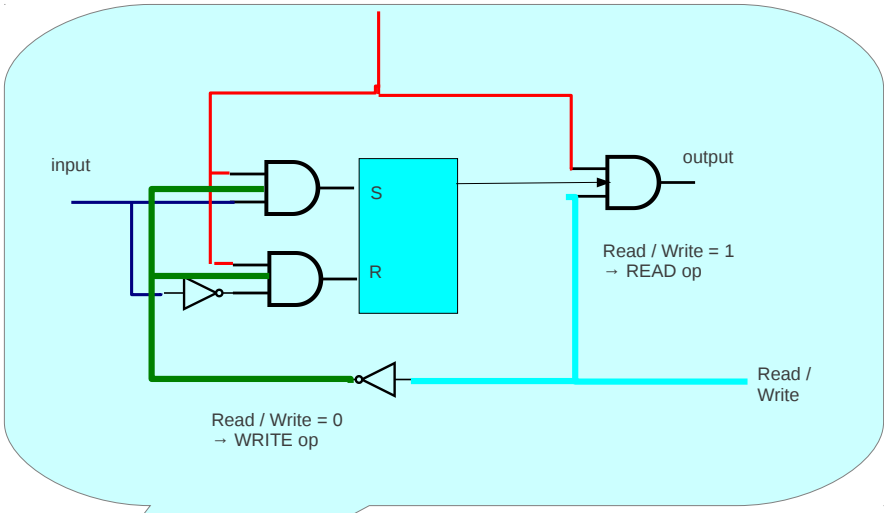


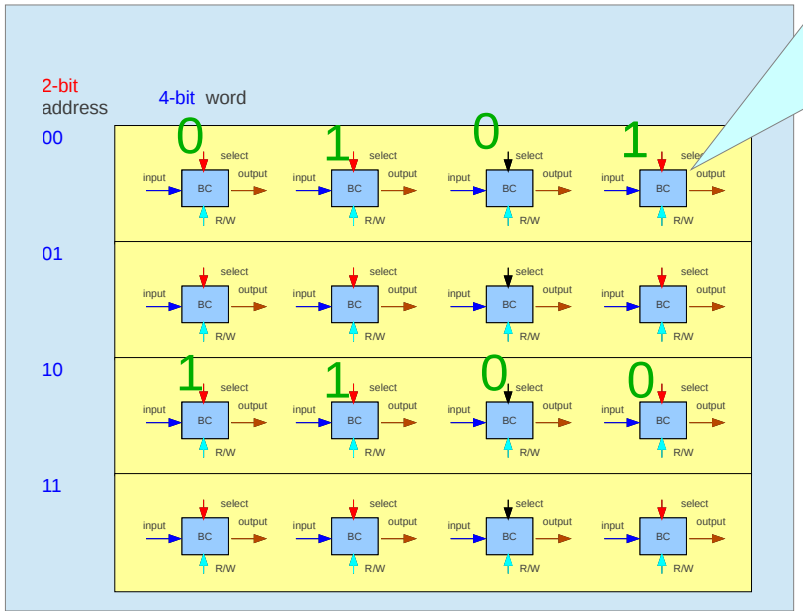
A Memory Map


00
01
10
11
select
input
output
BC
R/W
2-bit addr
Bit 3
Bit 2
Bit 1
Bit 0

4x4 Memory


2-bit
address
4-bit word
00
01
10
11
0101
1100


input
S
R
output
Read / Write = 1
→ READ op
Read /
Write
Read / Write = 0
→ WRITE op


2-bit
address
4-bit word
00
01
10
11
0
1
0
1
1
1
0
0
select
input
BC
output
R/W

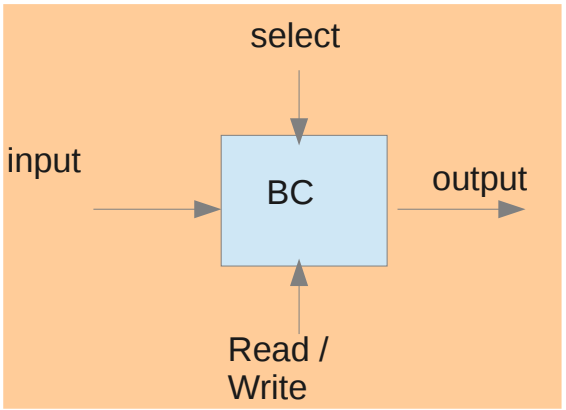
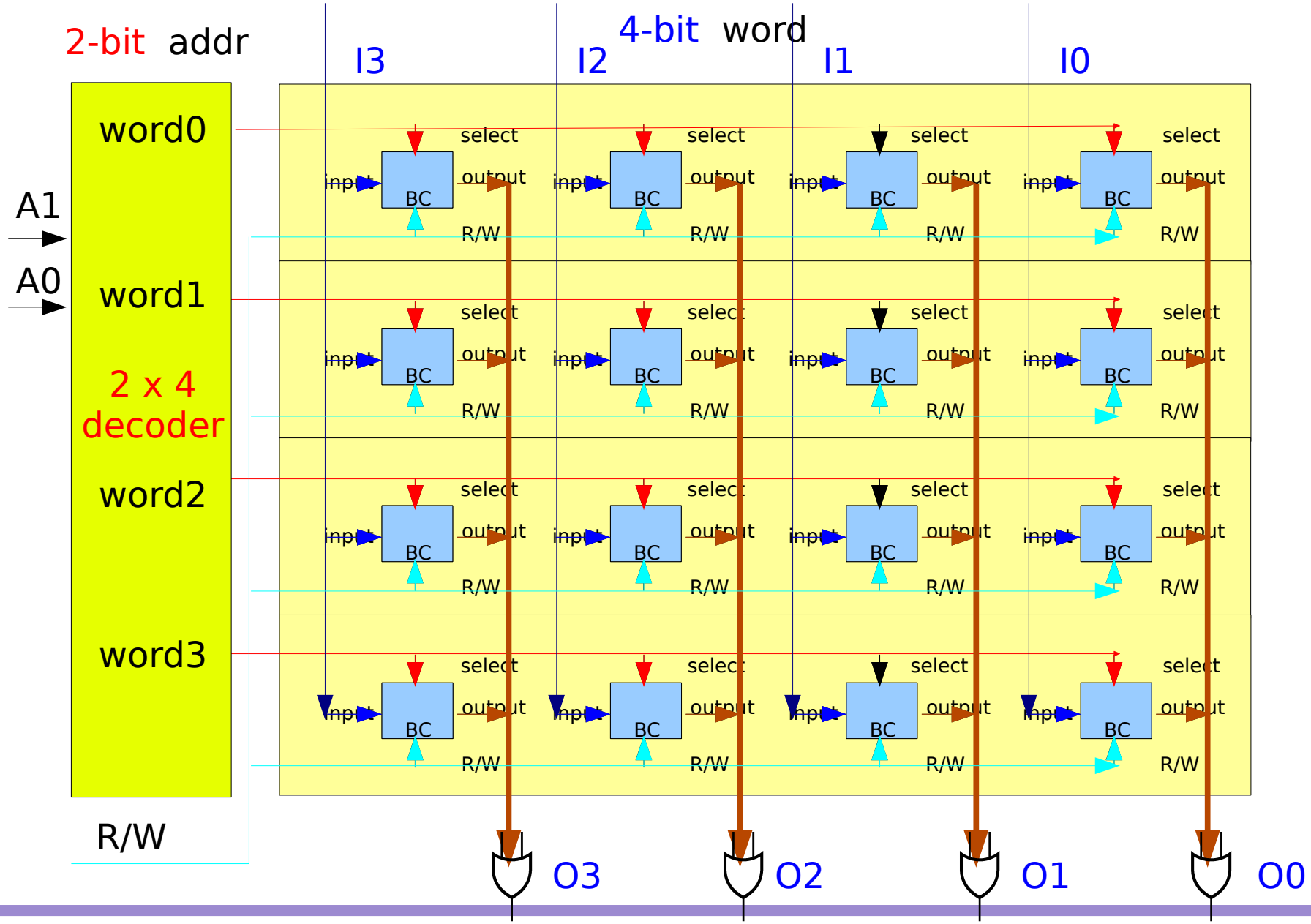
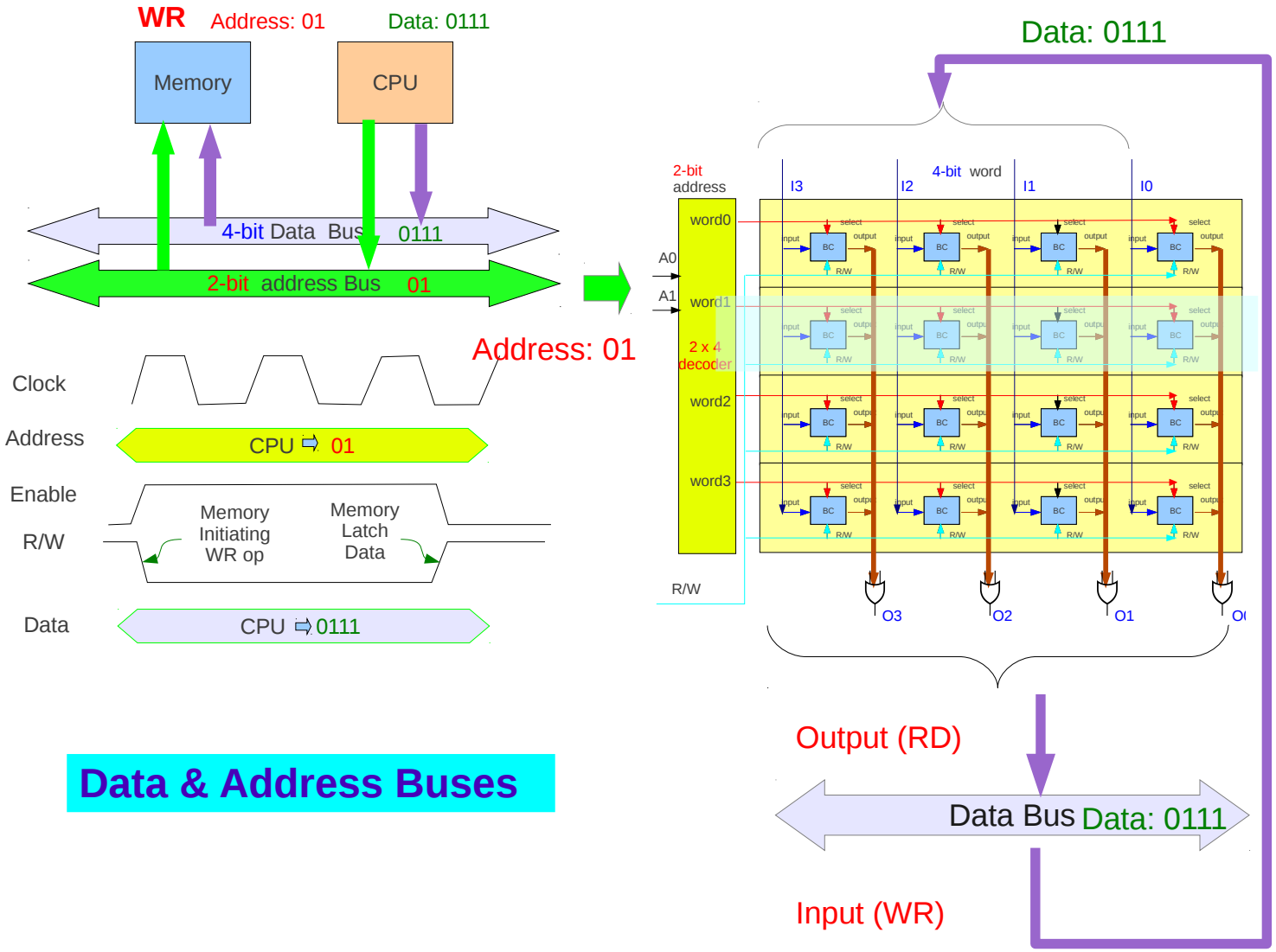

select
input
BC
output
Read /
Write

Diagram for a 4x4 Memory

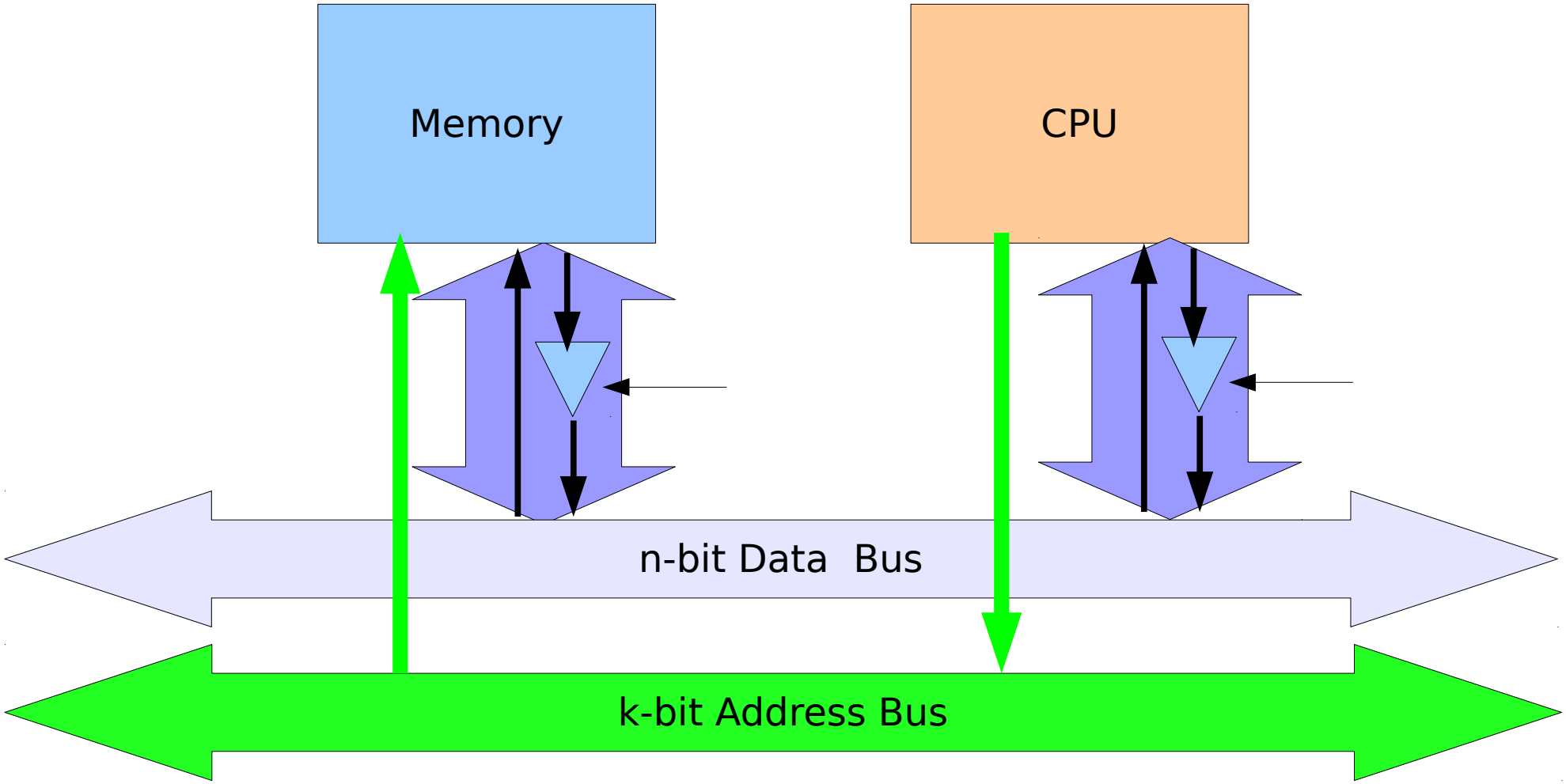


Write Cycle Example for a 4x4 Memory

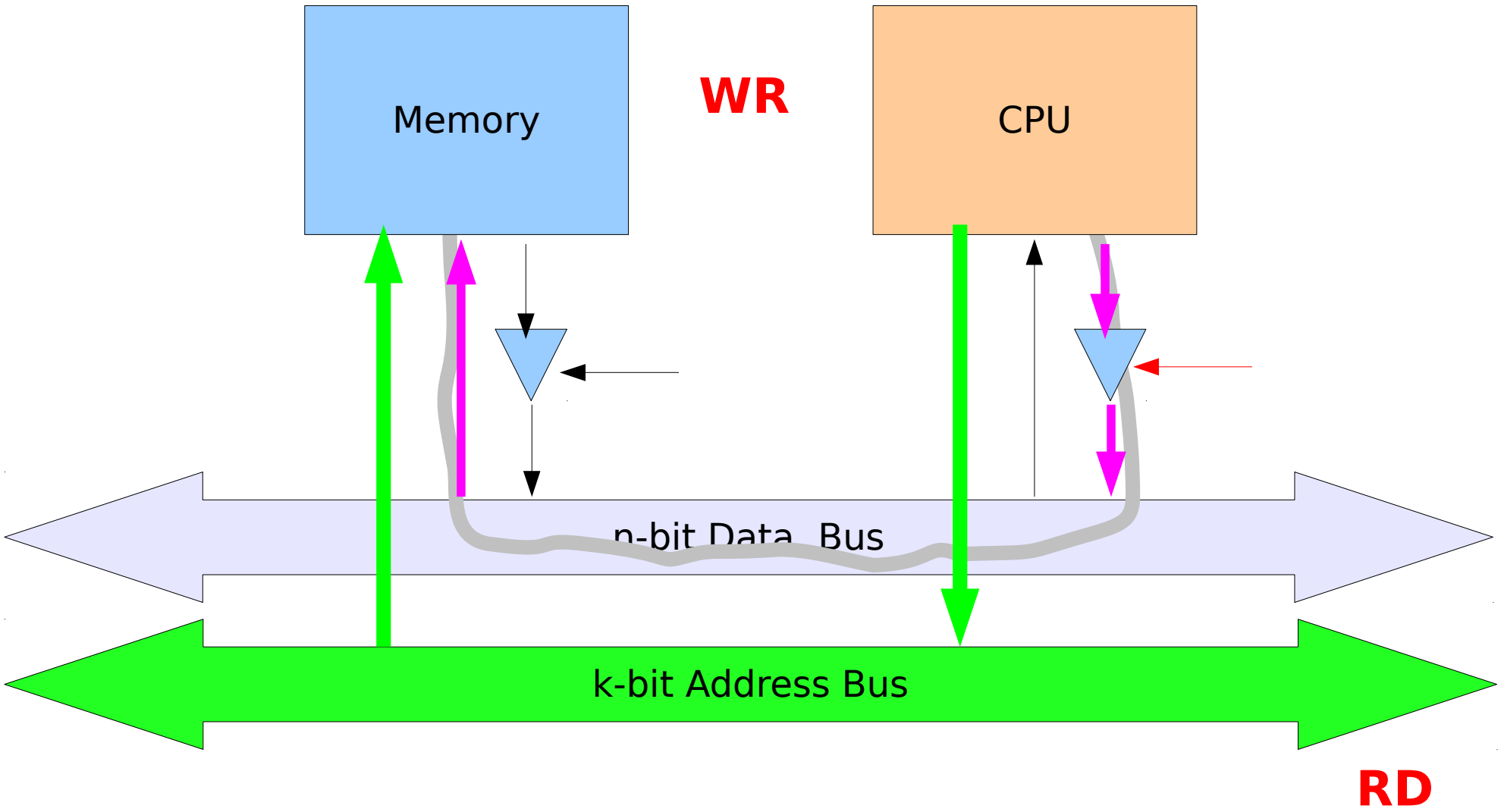


Data & Address Buses

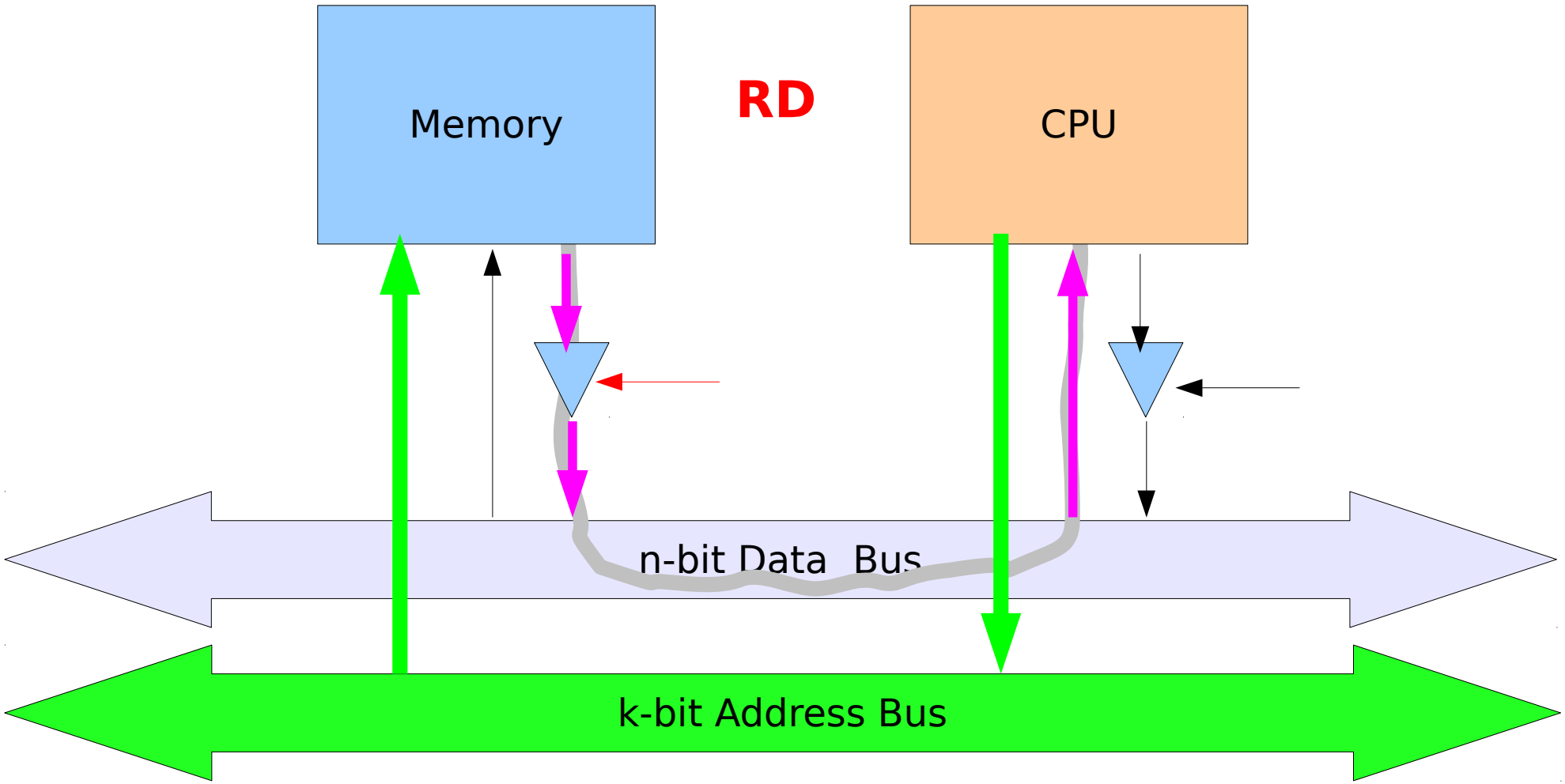
Bi-directional Data Bus



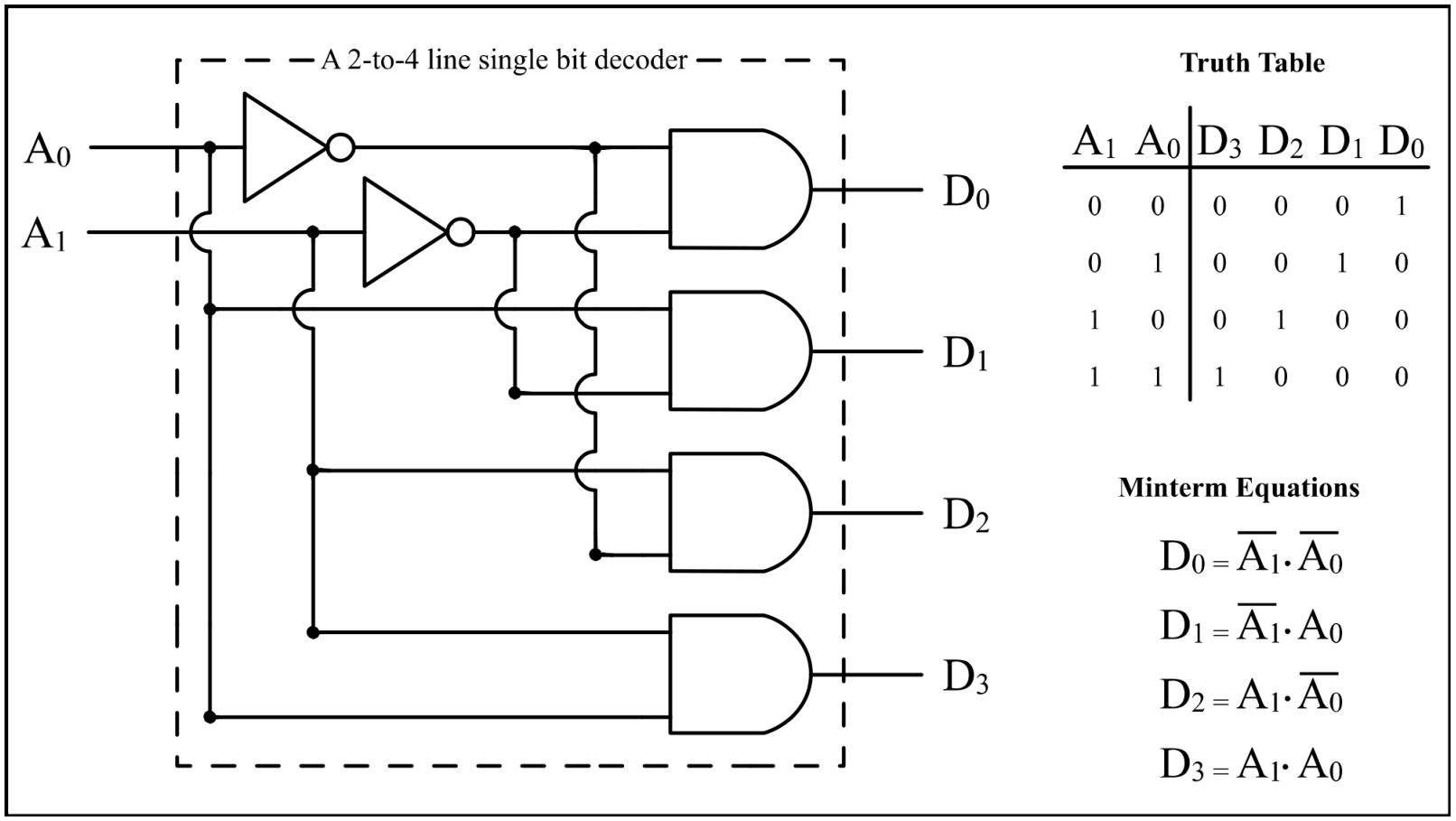
From CPU to Memory



From Memory to CPU



2 to 4 Decoder



Truth Table

A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

Minterm Equations

$$D_0 = \overline{A_1} \cdot \overline{A_0}$$

$$D_1 = \overline{A_1} \cdot A_0$$

$$D_2 = A_1 \cdot \overline{A_0}$$

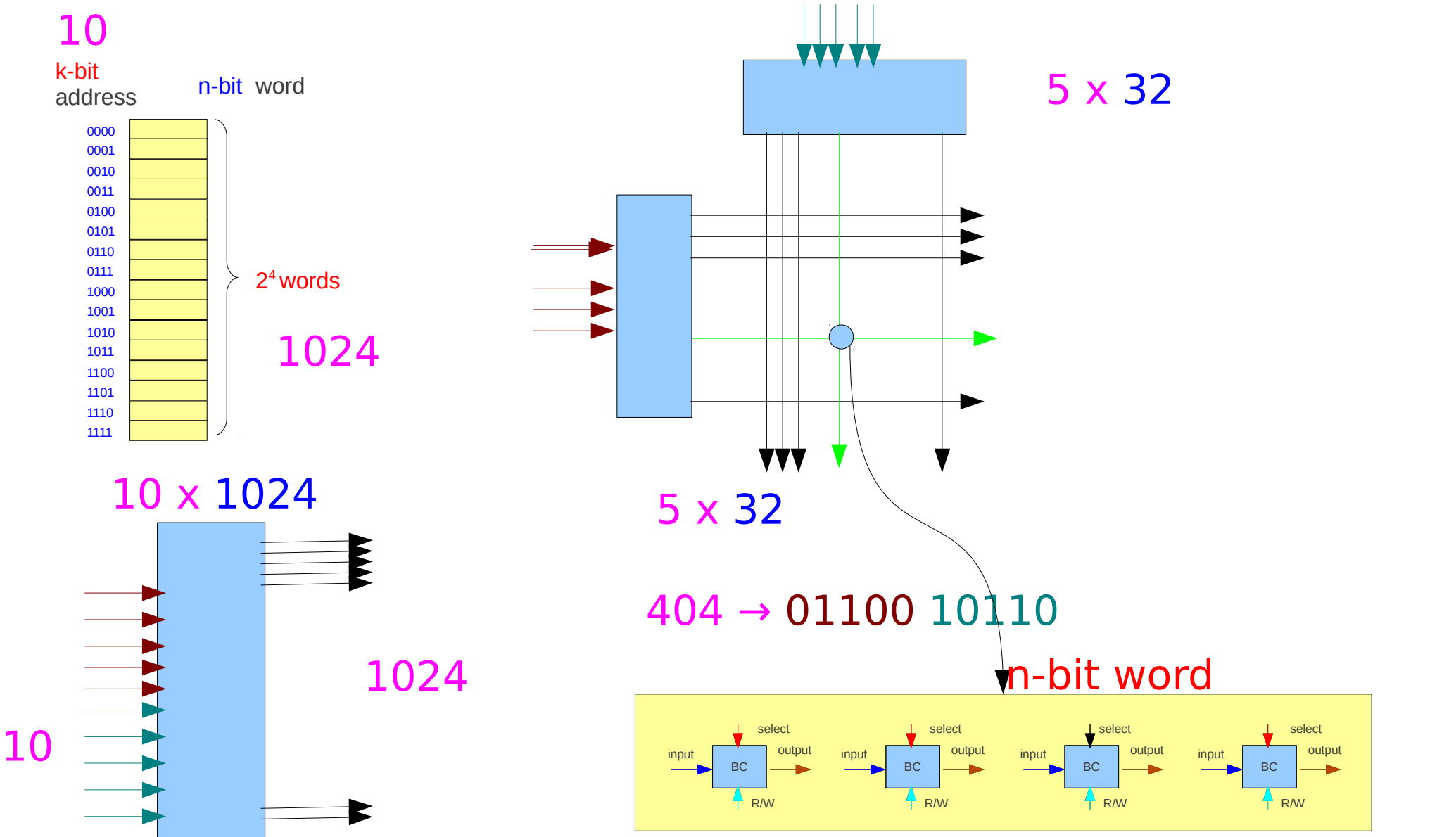
$$D_3 = A_1 \cdot A_0$$

2 x 4 decoder : 4 AND gates

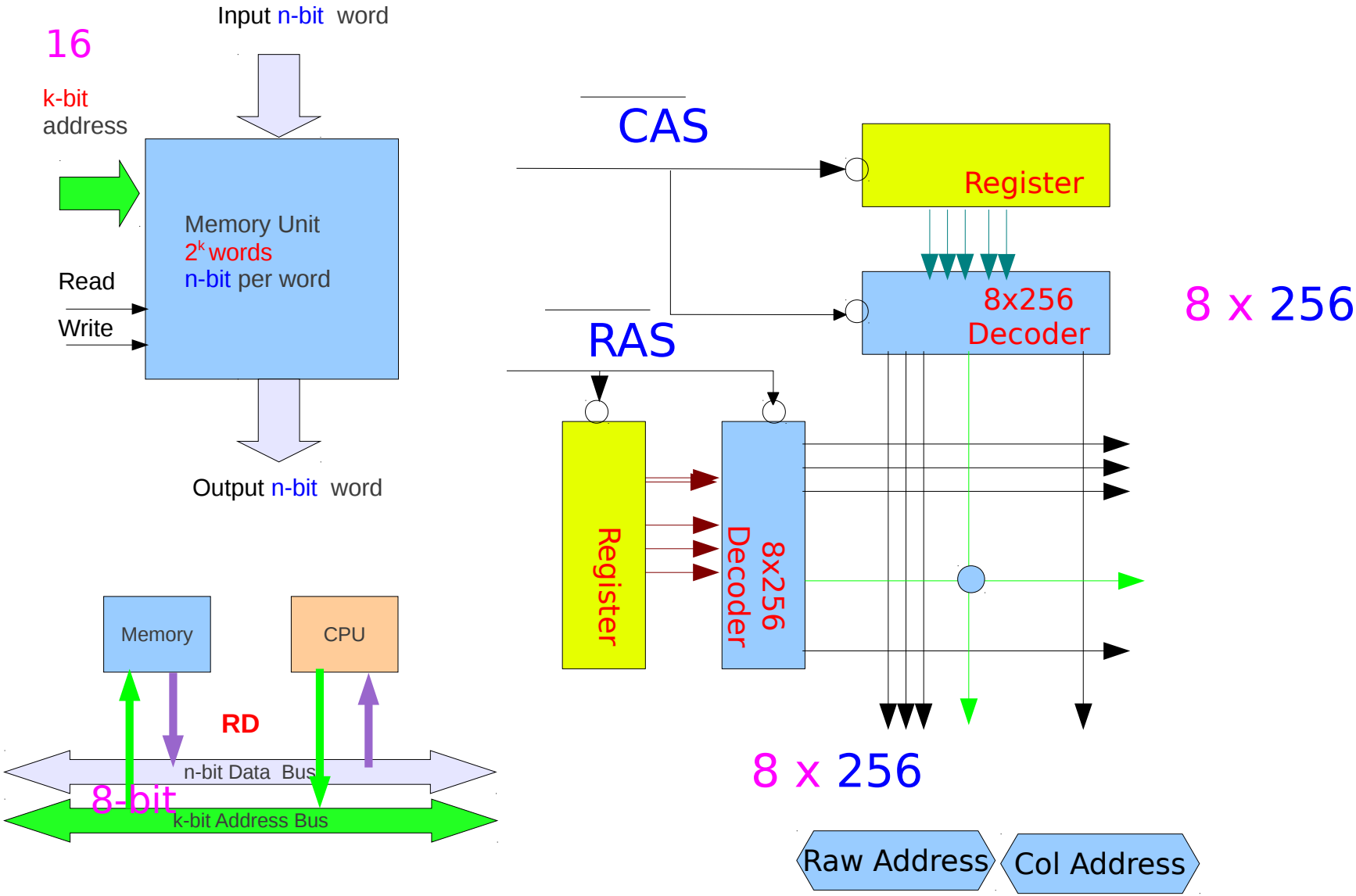
4 x 16 decoder : 16 AND gates

10 x 1024 decoder : 1024 AND gates

Coincidence Decoder



Time Sharing Address Bus



References

[1] Essential C, Nick Parlante
[2] Efficient C Programming, Mark A. Weiss
[3] C A Reference Manual, Samuel P. Harbison & Guy L. Steele Jr.
[4] C Language Express, I. K. Chun