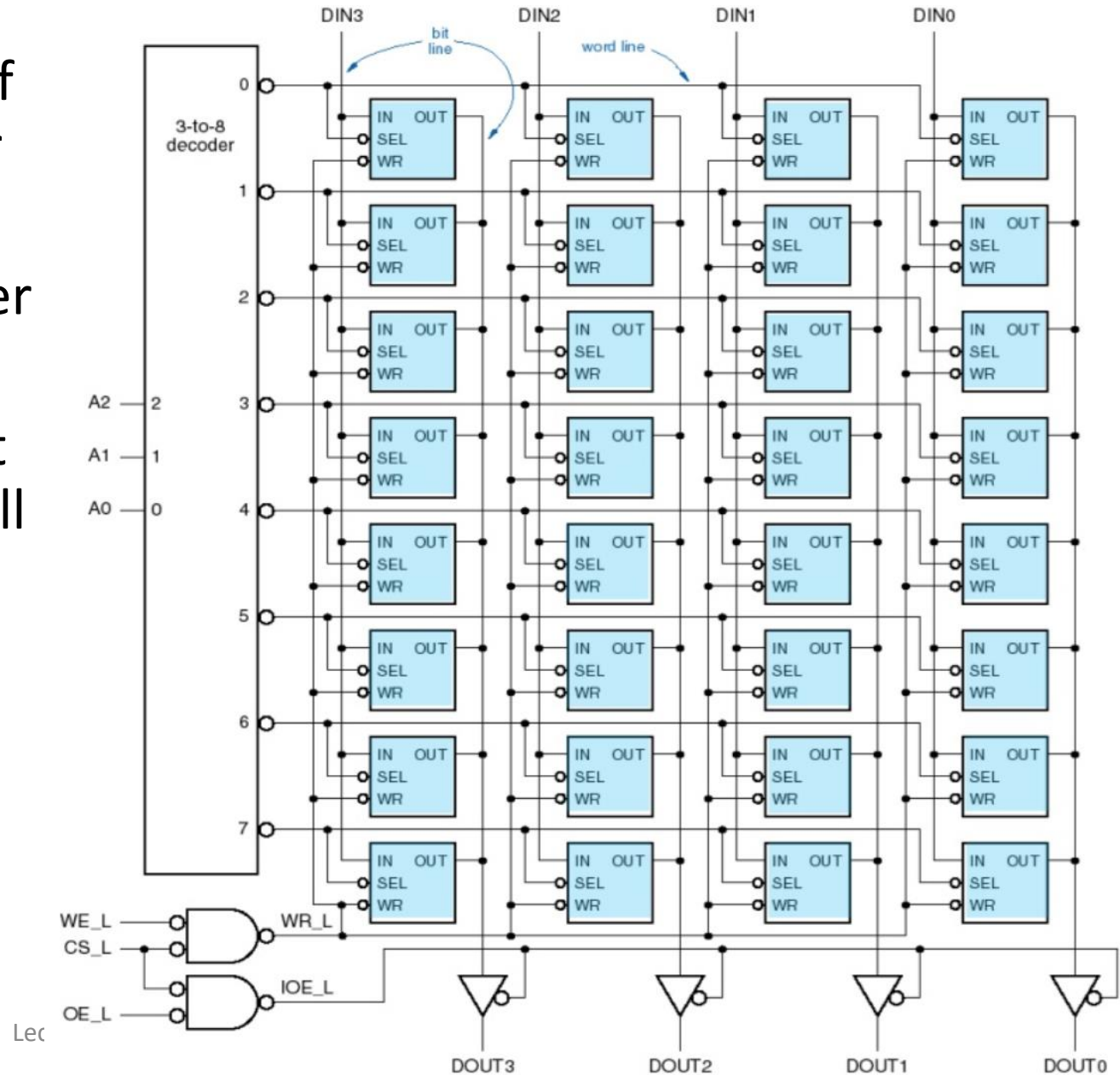
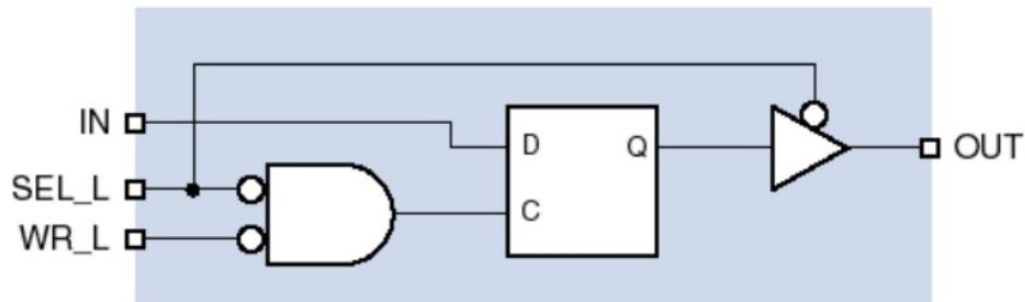


# Lecture 24 – Memory architecture 2

## Chapter 7

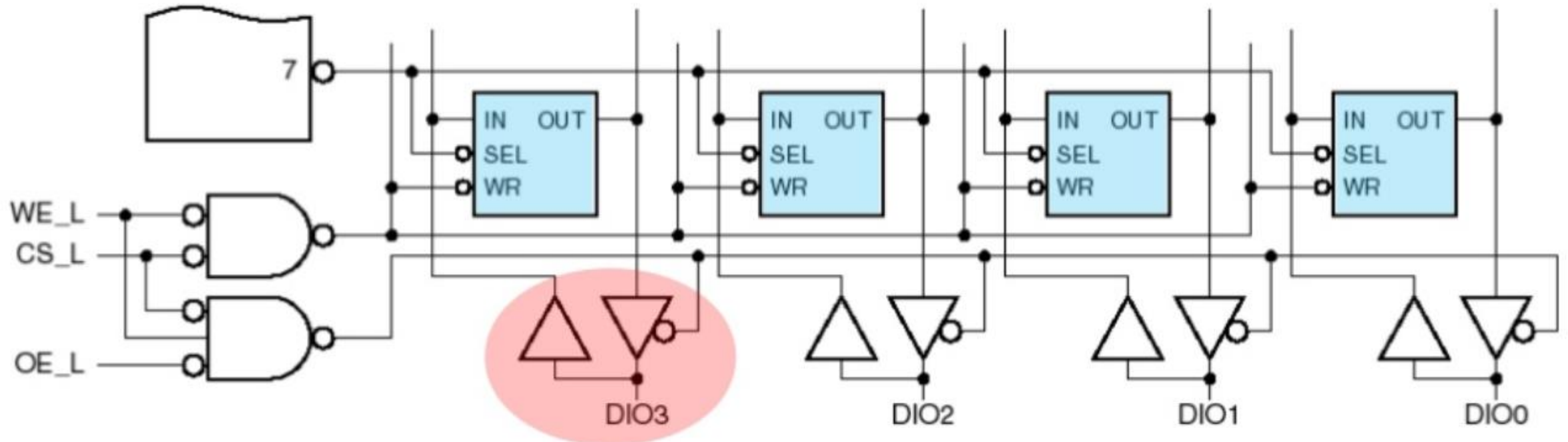
# Memory design

- The design of a memory will consist of a decoder circuit to select a particular “word line” based on the address
- The write enable, chip select and other inputs are common to all the latches
- The “bit line” determines what the bit at a particular position in the word will be



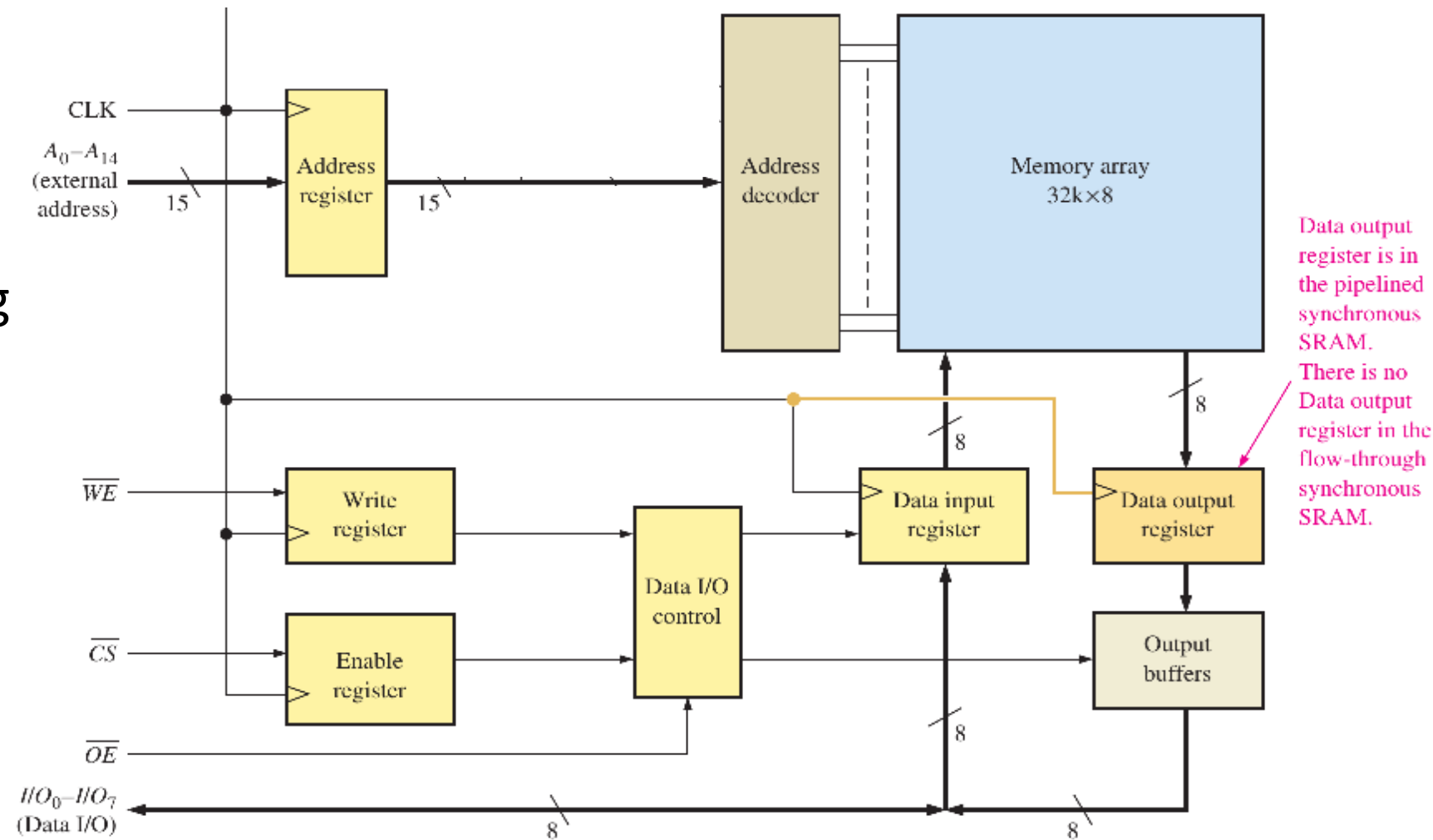
# Memory design

- We can modify the output such that we have the same lines for both input and output case
- This is very common both inside the processor (internal buses) and outside the processor (such as digital IO or GPIO)

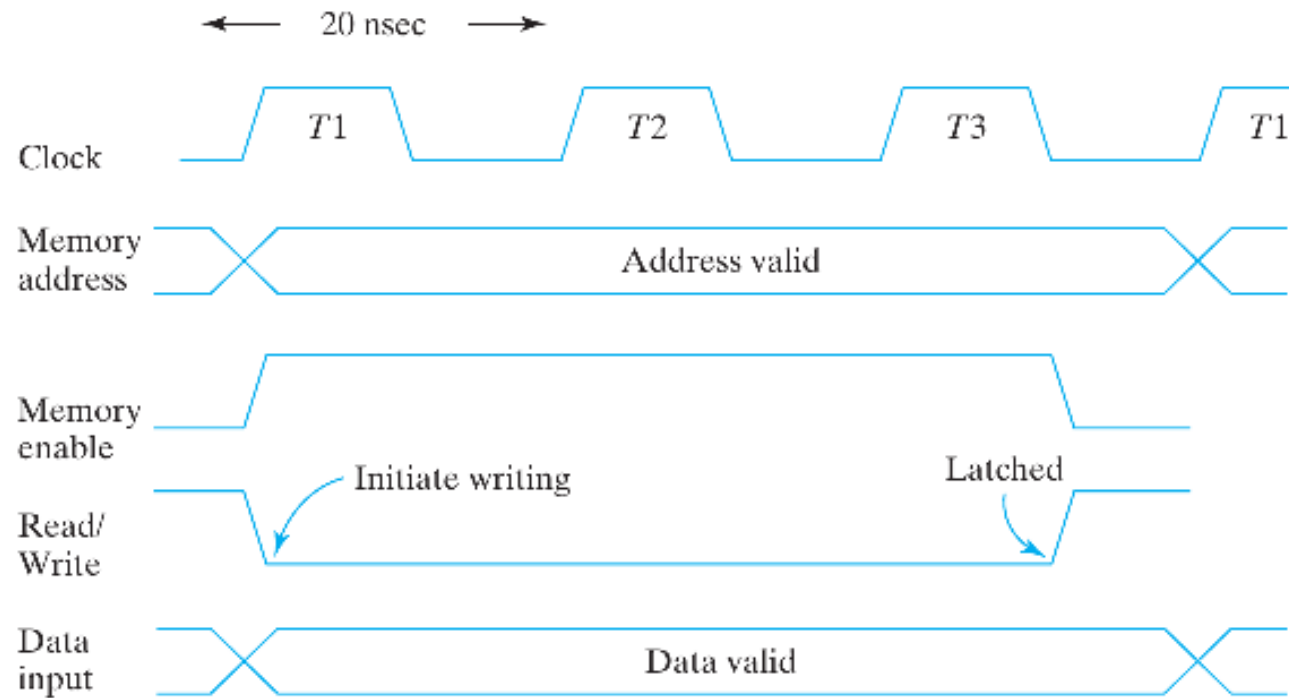


# Synchronous RAMs

- We can modify the input output behaviour of the SRAM array to make it synchronous
- We can easily do it using registers at appropriate points
- We can make the output *synchronous* or *flow through*
- They are all controlled using a common clock

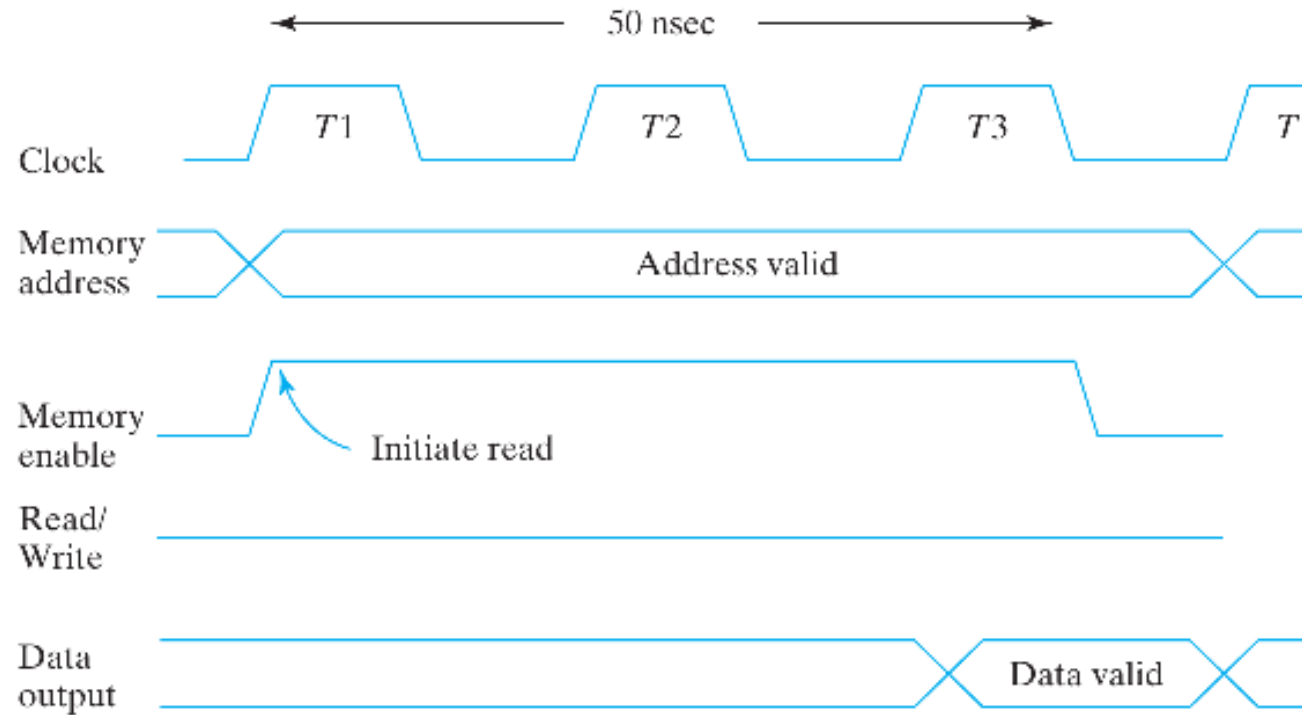


# Memory read write timing



## Write cycle

# Memory read write timing



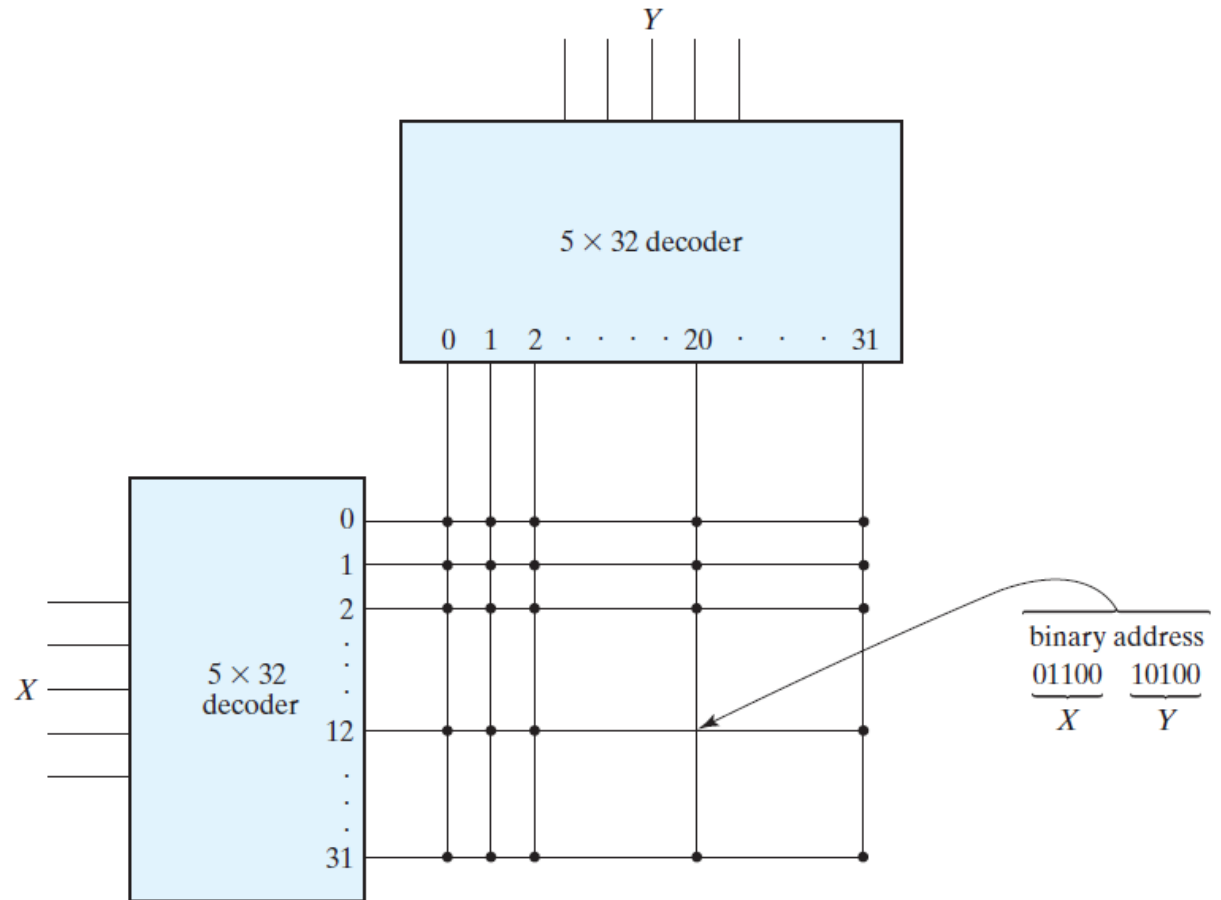
Read cycle

# Address decoding: coincident decoding

- A decoder with  $k$  inputs and  $2^k$  outputs requires  $2^k$  AND gates with  $k$  inputs per gate
- The total number of gates and the number of inputs per gate can be reduced by employing two decoders in a two-dimensional selection scheme
- The basic idea in two-dimensional decoding is to arrange the memory cells in an array that is close as possible to square
- In this configuration, two  $k/2$ -input decoders are used instead of one  $k$ -input decoder
- One decoder performs the row selection and the other the column selection in a two-dimensional matrix configuration

# Coincident decoding

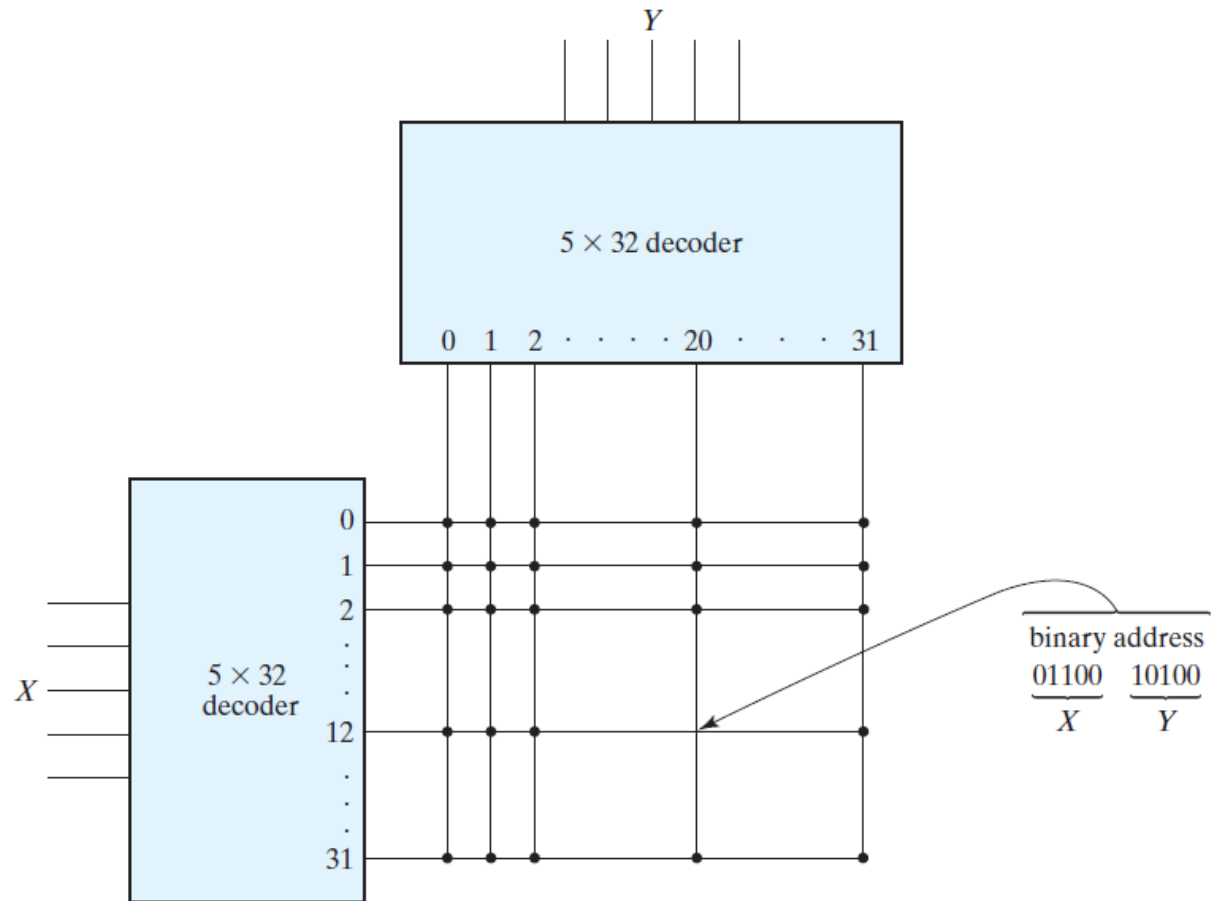
- For example, instead of using a single  $10 \times 1,024$  decoder, we use two  $5 \times 32$  decoders
- With the single decoder, we would need 1,024 AND gates with 10 inputs in each
- In the two-decoder case, we need 64 AND gates with 5 inputs in each
- The five most significant bits of the address go to input  $X$  and the five least significant bits go to input  $Y$
- Each word within the memory array is selected by the coincidence of one  $X$  line and one  $Y$  line
- Thus, each word in memory is selected by the coincidence between 1 of 32 rows and 1 of 32 columns, for a total of 1,024 words





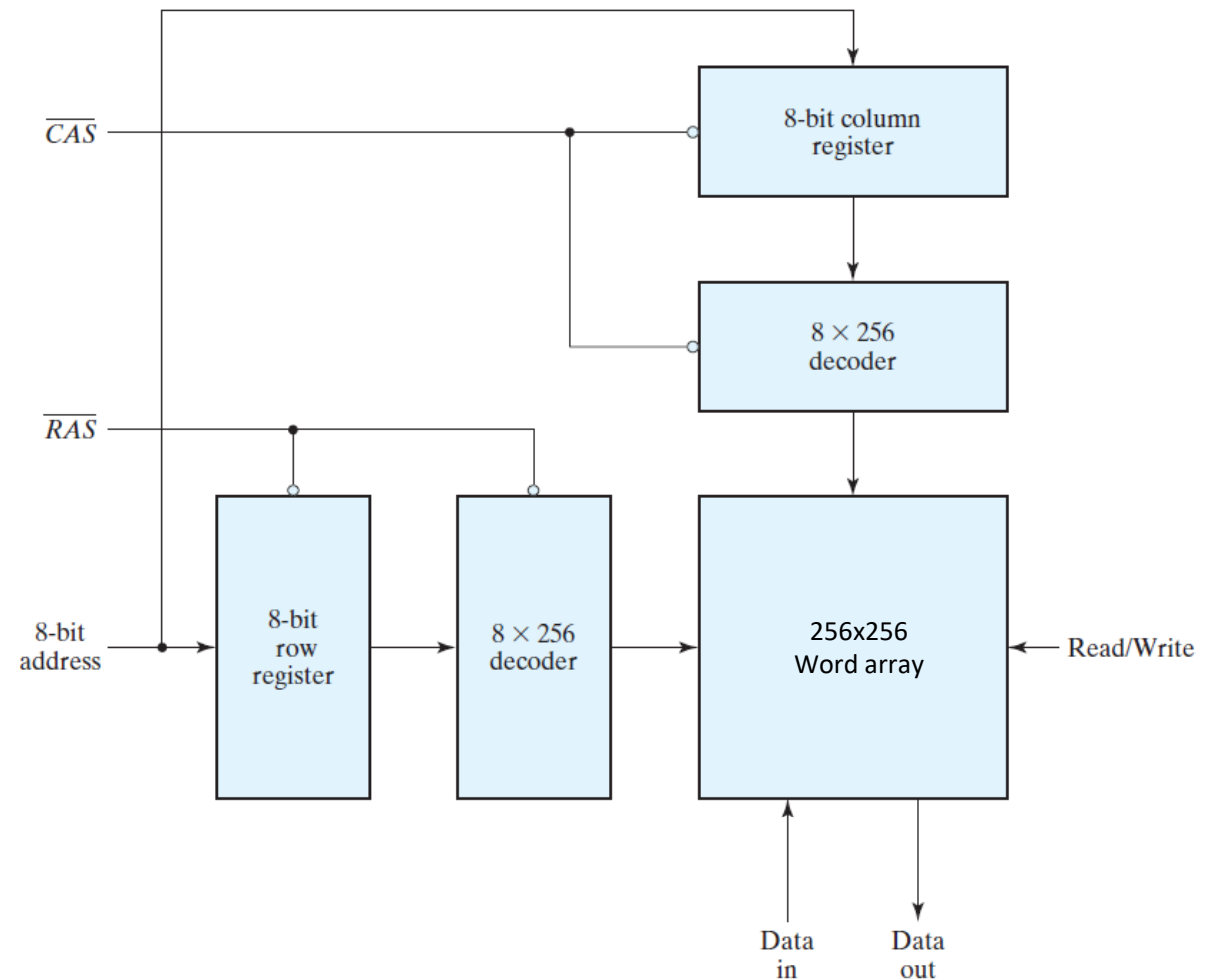
# Coincident decoding

- As an example, consider the word whose address is 404. The 10-bit binary equivalent of 404 is 0110010100. This makes  $X = 01100$  (binary 12) and  $Y = 10100$  (binary 20)
- The  $n$ -bit word that is selected lies in the  $X$  decoder output number 12 and the  $Y$  decoder output number 20
- All the bits of the word are selected for reading or writing



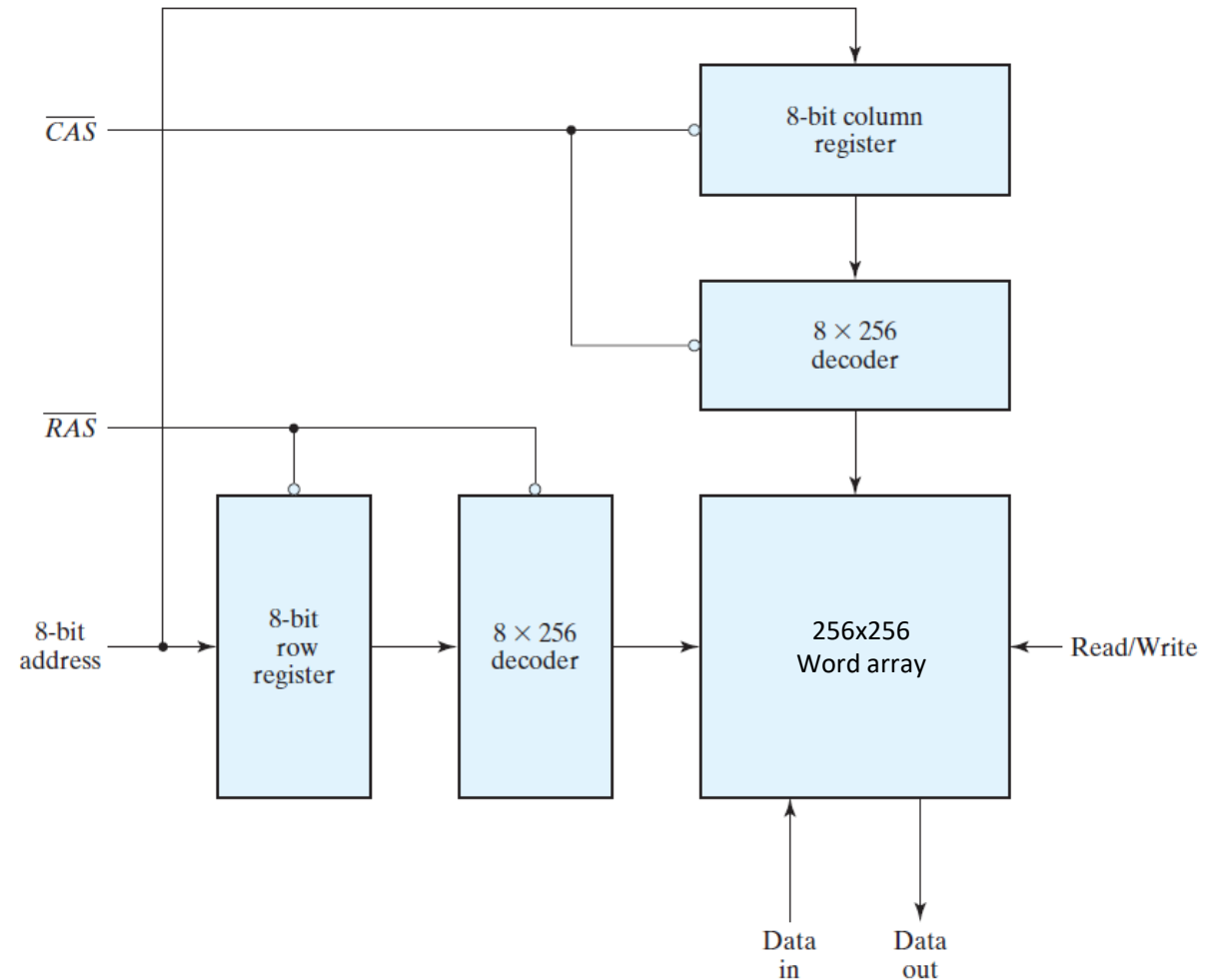
# Address multiplexing

- To reduce the number of pins in the IC package, designers utilize address multiplexing whereby one set of address input pins accommodates the address components
- In a two-dimensional array, the address is applied in two parts at different times, with the row address first and the column address second
- Since the same set of pins is used for both parts of the address, the size of the package is decreased significantly



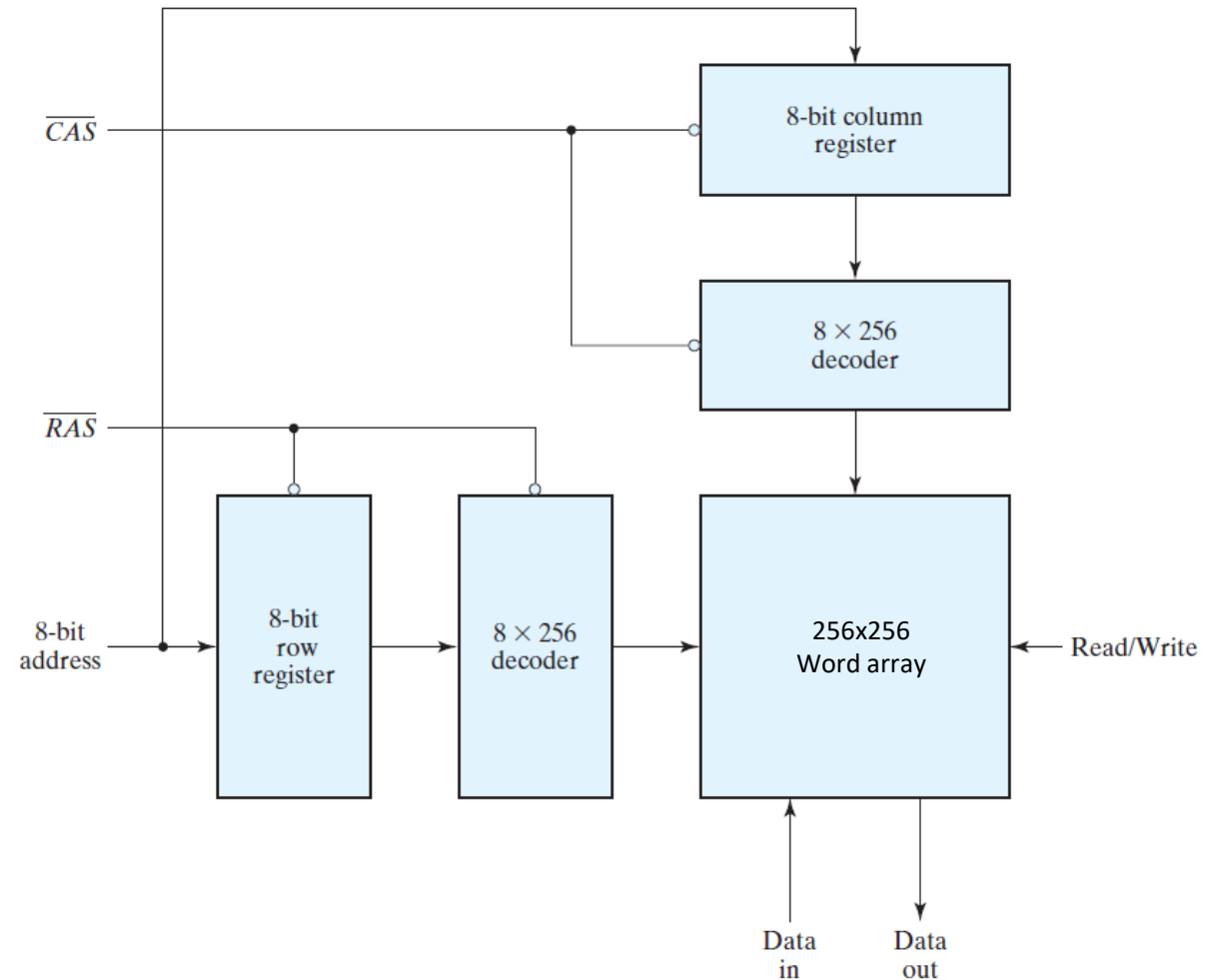
# Address multiplexing

- We will use a 64K-word memory to illustrate the address-multiplexing idea
- The memory consists of a two-dimensional array of cells arranged into 256 rows by 256 columns, for a total of  $2^8 * 2^8 = 2^{16} = 64K$  words
- There is a single data input line, a single data output line, and a read/write control, as well as an eight-bit address input and two address *strokes*, the latter included for enabling the row and column address into their respective registers
- The row address stroke (RAS) enables the eight-bit row register, and the column address stroke (CAS) enables the eight-bit column register



# Address multiplexing

- The 8-bit row address is applied to the address inputs and RAS is activated
- This loads the row address into the row address register
- RAS also enables the row decoder so that it can decode the row address and select one row of the array



# Address multiplexing

- The 8-bit column address is again applied to the address inputs, and CAS activated
- This transfers the column address into the column register and enables the column decoder
- Now the two parts of the address are in their respective registers, the decoders have decoded them to select the one cell corresponding to the row and column address, and a read or write operation can be performed on that cell
- CAS must go back to the logic 1 level before initiating another memory operation

