COMPUTER MEMORY

Learning Objectives

✓Know the difference between primary and secondary and volatile and non-volatile storage

- RAM is primary storage. It is volatile so only retains its contents when there is a connected power supply
- A hard disk is secondary storage. It is non-volatile so retains its contents even if there is no connected power supply

✓Explain the relationship between the width of the address bus and the number of memory locations that can be addressed.

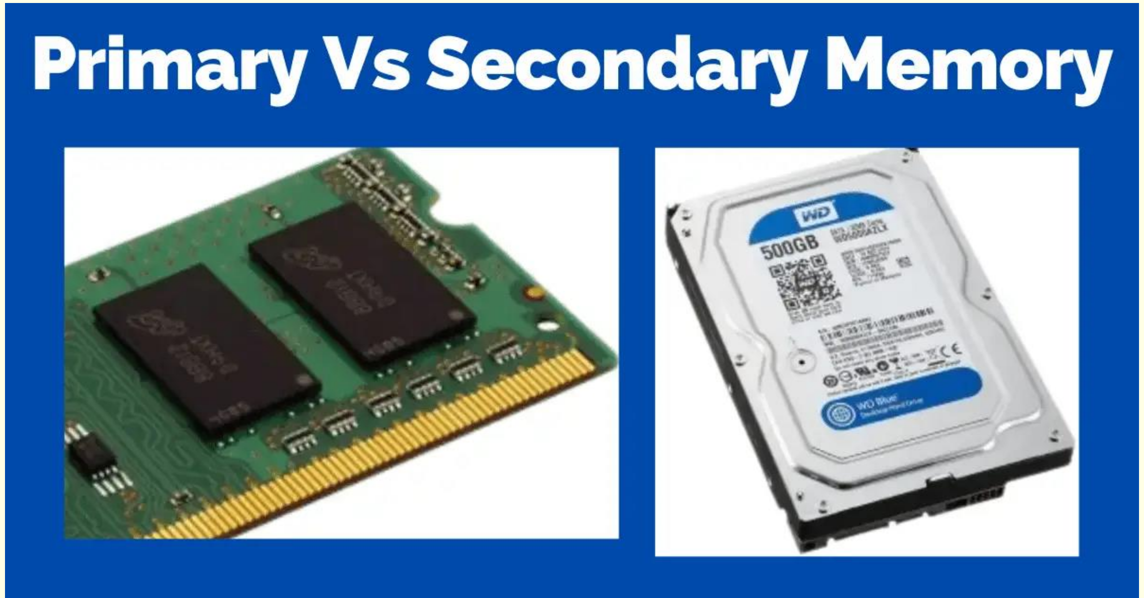
- Each unique address identifies one memory location.
- The more ‘bandwidth’ that is available in the address bus, the more uniquely identifiable memory locations there are

✓Calculate the number of addressable memory locations provided by an address bus of a specified width.

- Use 2^n to calculate the number of unique binary patterns that can be produced by *n* bits.

Memory (RAM)

- Random Access Memory (RAM) is **volatile memory** – this means that its contents are erased when its power supply is turned off.
- RAM is often referred to as '**primary storage**', as it sits close to the CPU and instructions are sent to RAM before being processed by the CPU
- However, it isn't possible to keep a computer on permanently. So it is necessary to have somewhere to store the programs and data (that make a computer general purpose) when the power is switched off.
- The **non-volatile storage** required for this job is most often a hard disk.
- Hard disks are often referred to as 'secondary storage', to make them distinct from 'primary storage'.



Main memory Instructions and data

- Instructions and data are stored in RAM until they are needed by the CPU.
- Each instruction and item of data is stored in a location in memory.
- Each element of the memory has a unique address.
- This means that each instruction has an address.

Address	Data
0001	LOAD 0101
0010	ADD 0110
0011	STORE 0111
0100	STOP
0101	23
0110	12
0111	

Opcode and operand

The instruction is split into an **Opcode** and an **Operand**

Opcode is which operation to carry out. A processor has an instruction set.

The **operand** specifies the data that needs to be acted on.
For example: ADD 5

Address	Data
0001	LOAD 0101
0010	ADD 0110
0011	STORE 0111
0100	STOP
0101	23
0110	12
0111	

Storing instructions/ data/memory addresses

Instruction Set

INP
OUT
LOAD
ADD
COMPARE
JUMP IF
JUMP

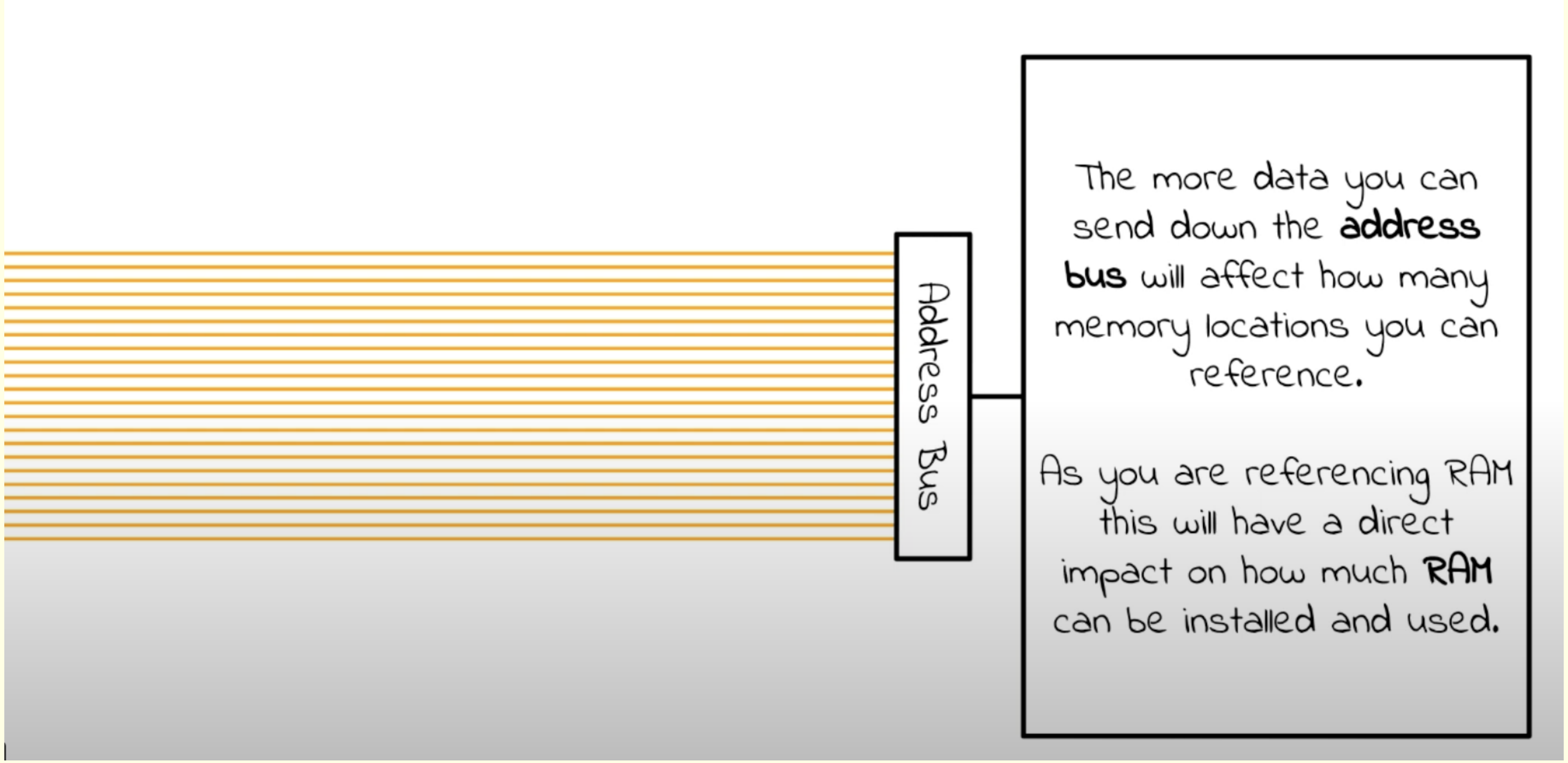
Memory Addresses

Recap Buses

Data bus	Carries instructions from RAM that is being transferred to the CPU. Carries instructions from CPU to RAM. Bidirectional
Control bus	Carries signals that control the CPU components. For example a signal to start reading the next instruction from RAM Bidirectional
Address bus	Carries the address of a memory location – for example the address of an instruction being fetched from memory. unidirectional

Address Bus Width

Address Bus Width refers to the number of unique memory addresses that can be accessed.



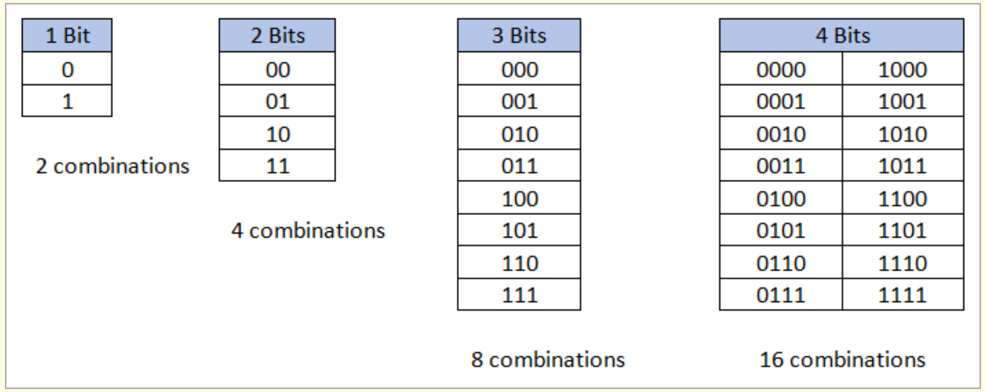
The more data you can send down the **address bus** will affect how many memory locations you can reference.

As you are referencing RAM this will have a direct impact on how much **RAM** can be installed and used.

Calculating the maximum number of memory addresses

Formula finding the **maximum number of memory addresses**:

$$2^n$$



1 Bit
0
1

2 combinations

2 Bits
00
01
10
11

4 combinations

3 Bits
000
001
010
011
100
101
110
111

8 combinations

4 Bits	
0000	1000
0001	1001
0010	1010
0011	1011
0100	1100
0101	1101
0110	1110
0111	1111

16 combinations

Worked Example:

Consider a CPU with a 4-bit bandwidth (address bus). How many unique memory addresses can it access?

4-bit addresses range from 0000 to 1111 inclusive.

1111 in denary is 15 (plus 1 because we must include 0)

which gives 16 addresses

16 addresses is $2^{4.}$