ICT – Analogue and Digital Data – Binary Units and ASCII

1 of 20 – Welcome

Welcome to this session on analogue and digital data – binary units and ASCII.

By the end of this session, you will:

* Know what bytes and units are
* Understand how bytes and units are measured
* Know what is meant by ASCII
* Understand how this relates to binary structures

2 of 20 – Analogue and digital data: a quick reminder

Analogue data and digital data are terms that you might remember from an earlier session. These are two different types of data that can be stored on multiple devices.

Remember: **analogue data** is **naturally occurring data**, meaning it is represented by something physical – like temperature, or the human voice – and is not found in a digital setting. This type of data needs to be changed, or converted, before it can be understood by a computer.

**Digital data** is often thought of as a representation of something. It is **usually a** **number sequence** – remember, this is where binary comes in – and the numbers themselves stand for whatever data is being stored. This converted data can then be understood by a computer system.

3 of 20 – Binary: a quick reminder

You might remember from an earlier session that the binary system is how data is input and understood by a computer system.

Each part of a computer understands binary code, which is why analogue data needs to be converted – using binary code – for it to be understood by a computer as digital data.

Binary is made up of bits, which are individual binary digits (meaning 0 and 1, because binary works with these two digits alone).

These bits are usually grouped together, and when eight bits are grouped they can then be referred to as a byte.

4 of 20 – Bytes, bits and words

You might also remember that bits can be used to indicate the amount of data a processor can handle, without it slowing down the general functions of your computer system, and they can also be used in memory addresses, too.

Think back to terms such as 32-bit and 64-bit – these are huge improvements on the 8-bit and 16-bit memory options that used to be available for computers.

These 32-bit and 64-bit examples illustrate how much memory a certain computer system has, explained in binary terms.

Another term that we can think of alongside binary bits and bytes is word, or word length.

5 of 20 – Word length: what does this mean?

In computer systems, the term word (and word length) is closely related to 32-bit and 64-bit discussion, as word is typically used to describe the individual number of bits that have been grouped together in order to then be used by the computer’s processor.

This means a word is a set group of binary data (bits), which is then handled by various instruction-led components, and by the Central Processing Unit (CPU) generally.

When word length is mentioned, this is referring to the processing power of the computer system – ultimately meaning how many bits (or rather, how long a word) can be processed by the CPU at one time.

If the word is longer, then the data that is being moved per clock cycle is larger, meaning the computer has a particularly high processing power.

6 of 20 – Units and bytes: how do they work?

In computer systems, units is the term used to cover different types of bytes, such as: kilobytes, megabytes, gigabytes, terabytes, and petabytes. Each of these different units is made up from a certain amount of bytes (remember: bytes are when eight bits have been grouped together).

These different units can explain to a user what the memory and data capabilities are of a storage unit, which is why they are important when considering different types of data.

Because computers rely on binary numbers, it is worth noting that although these units are expressed as even numbered bytes – for example, a megabyte is thought of as a million bytes – there are likely to be some bytes left over from this figure – meaning a megabyte is actually made up of 1,048,576 bytes, but it is written differently for the sake of binary formatting.

7 of 20 – Units and bytes: an example

From the table below we can see these calculations in practice:

|  |  |  |  |
| --- | --- | --- | --- |
| Unit | Approximate number | How the number is calculated | Accurate number |
| Megabyte (million bytes) | 1,000,000 | 1024 times by 1024 | 1,048,576 |
| Gigabyte (thousand million bytes) | 1,000,000,000 | 1024 times by 1024 times by 1024 | 1,073,741,824 |

1024 is used in these calculations as it is representative of a perfectly round number in binary notation. Because of this, it is used a lot in computer processes and so it is an important figure to remember.

8 of 20 – Binary in other systems

While binary has many parts which can be tricky to fully understand, when using computer systems it is worth knowing how binary works, simply because it is used in so many places.

Computers programmers – and computer users generally – rely on binary within their computer systems on a daily basis, which is why other systems now rely on binary too.

ASCII, which we will discuss in some depth a little later in this session, is just one example of how binary notation is being used to design new coding systems and languages.

9 of 20 – Question 1

Match these terms; **bit**, **byte** and **word**, to the descriptions below:

1. Typically a collection of eight individual bits
2. An individual binary digit, this will be either 0 or 1
3. This is when any number of bits are grouped together; they can vary in size and overall length

The correct answers are:

Typically a collection of eight individual bits is a description of a **byte**.

An individual binary digit, this will be either 0 or 1 is a description of a **bit**.

This is when any number of bits are grouped together; they can vary in size and overall length is a description of a **word**.

10 of 20 – Question 2

Read the statements below and decide which ones are true and which ones are false.

Analogue data needs to be converted, using binary code, to become digital data.

True

False

The correct answer is: True

A binary digit is a binary number, and these can be any number between 0 and 10.

True

False

The correct answer is: False

If the length of a word is longer then this will slow down your processing speed.

True

False

The correct answer is: False

Computers rely on binary numbers, so unit amounts are expressed as even round numbers even though sometimes they are not.

True

False

The correct answer is: True

11 of 20 – What does ASCII stand for?

ASCII is the acronym, or abbreviation, used for the **A**merican **S**tandard **C**ode for **I**nformation **I**nterchange. This code was developed by the American National Standards Institute who began work on this project back in 1960.

The ASCII is a code that is used to represent characters in the English language – this includes alphabetical characters that are both lower and upper case, but it also includes numerical values and one or two punctuation symbols, too.

ASCII represents 128 characters (letters, numbers, or symbols) in total and each of these characters are given their own number, according to the binary system used by ASCII.

Visit the following website to find a completed table of ASCII letters:

[ASCII Binary Character Table](http://sticksandstones.kstrom.com/appen.html)

12 of 20 – ASCII: assigning values

In the previous section’s table we can see that lower-case characters are assigned to numbers 97 – 122, while upper-case letters are assigned to numbers 65 – 90.

Although it is not shown in this particular table, it is also worth noting that numbers (0 – 9 in our normal numbering system) are assigned to values 48 – 57 in ASCII.

The values that are left after these specific characters have been assigned are given to the punctuation characters that were mentioned earlier.

Because the ASCII system relies on binary formatting, we can think of these assigned values as binary units. If you take another quick look at the table in the website below, you will notice that each ASCII value also has a binary equivalent listed alongside it.

[ASCII Binary Character Table](http://sticksandstones.kstrom.com/appen.html)

13 of 20 – ASCII: binary units

It is important to note that ASCII uses binary units in a special and unique way.

The binary units that are used in ASCII are all 7-bit binary numbers. This means that all of the binary numbers are made up of 7 individual number parts (**remember:** these numbers will only ever be 0s and 1s).

However, there is now an extended version of ASCII which allows for 8-bit binary numbers instead of 7-bit. The popularity and reliability of ASCII means that it has needed to expand over time, which is why it now covers more characters than it did originally.

14 of 20 – ASCII: why do we use it?

ASCII is the most common format that is used for text files, both on computers and on the internet. It is well-known for being able to shrink text in a way that allows your computer to hold more without slowing down the system.

Because ASCII is primarily used for text files, this code type also makes it possible to transfer text data between different computer systems as well.

However, it is worth noting that this data exchange can only happen between computers that are using ASCII – if the other system does not use this code then the text will not transfer properly.

While ASCII is a popular code in modern computer systems, there are one or two limitations to using it.

15 of 20 – ASCII: disadvantages

While ASCII is useful for storing and transferring text files, it cannot be used to store or transfer any sort of numerical files, as the system is not designed for this information.

Also, a key criticism of ASCII is that it simply does not cover enough characters. The 128 values (meaning 0 – 127) covers all of the English alphabet and numbers, and some punctuation. Unfortunately this means that special or unique characters, or even characters that are taken from other languages, cannot be properly coded by ASCII.

Even though there is now an Extended ASCII, which provides 8-bit binary units and so covers 256 characters, this is still criticised as not covering enough individual characters, making it a difficult code to use for some users.

16 of 20 – Question 3

What does ASCII stand for?

1. Association for Standard Coding in Information Industries
2. American Standard Code for Information Interchange
3. American Standard Code for Ideas and Information
4. American Standardised Codes for Information Exchanges

The correct answer is B, American Standard Code for Information Interchange.

17 of 20 – Question 4

Using the following choice of words; **languages**, **National Standards**, **numerical**, **special**, **128**, **1960**, **punctuation** and **binary units**, fill in the blanks for the paragraph below:

ASCII was developed by the American **blank** Institute who started work on the project in **blank**. ASCII is a code system that uses **blank** to represent a number of different characters. These characters can be alphabetical or **blank**, and there are a few values reserved for **blank** too. However, because ASCII only allows for **blank** characters, some have criticised the system for being too limited by not allowing for **blank** characters, or characters from other **blank**.

The correct paragraph should read:

ASCII was developed by the American **National Standards** Institute who started work on the project in **1960**. ASCII is a code system that uses **binary units** to represent a number of different characters. These characters can be alphabetical or **numerical**, and there are a few values reserved for **punctuation** too. However, because ASCII only allows for **128** characters, some have criticised the system for being too limited by not allowing for **special** characters, or characters from other **languages**.

18 of 20 – Question 5

What does the Extended ASCII code provide that the original does not?

1. 7-bit binary units to cover 265 characters
2. 7-bit binary units to cover 128 characters
3. 8-bit binary units to cover 256 characters
4. 8-bit binary units to cover 365 characters

The correct answer is C, 8-bit binary units to cover 256 characters.

19 of 20 – End

Well done! You have now completed this session on analogue and digital data – binary units and ASCII.

In this session we have covered:

* What bytes and units are
* How bytes and units are measured
* What is meant by ASCII
* How this relates to binary structures

If you have any questions about any of these topics, make a note and speak to your tutor for more help.