Binary representation and unsigned binary–denary conversion

2.1.1, 2.1.2, 2.1.3 · Data · revision notes

Why computers use binary, how many different values a given number of bits can represent, and how to convert between denary and 8-bit unsigned binary (0 to 255). These are quick 1 to 2 mark questions on Paper 1, and the skills are used in almost every other Data question.

Key facts

- Bit = one binary digit (0 or 1)
- Largest n-bit unsigned value = $2^n - 1$
- 8-bit unsigned range: 0 to 255 (0000 0000 to 1111 1111)
- MSB = leftmost bit (largest place value); LSB = rightmost bit (place value 1)
- n bits give 2^n different patterns
- 8-bit place values: 128 64 32 16 8 4 2 1
- Powers of 2: 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024
- Computers use binary because their switches (transistors) have two states

Notes

Why computers use binary

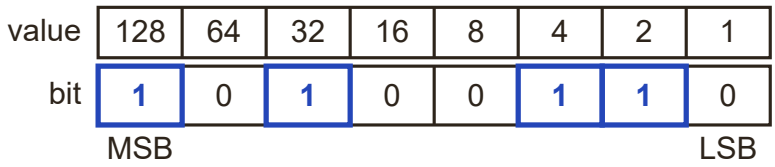
- A **bit** is a single binary digit: 0 or 1.
- Computers are built from billions of tiny switches (transistors). Each switch has only two states, on and off, so it can represent 1 or 0.
- Two states are easy to tell apart reliably. Ten voltage levels for denary would make circuits more complicated and errors more likely.
- All data (numbers, text, images, sound) and program instructions are stored as binary. The same pattern can mean different things: the program using it decides how it is interpreted.

How many different patterns?

- Each extra bit doubles the number of patterns. n bits give **2^n** different patterns (states): 1 bit gives 2, 2 bits give 4, 3 bits give 8, 4 bits give 16 and 8 bits give 256.
- An unsigned integer uses one pattern for 0, so n bits store **0 to $2^n - 1$**. 8 bits store 0 to 255.
- To find the fewest bits for a number of values, find the smallest power of 2 that is big enough. For 20 values: $2^4 = 16$ is too few, but $2^5 = 32$ is enough, so 5 bits. Grade 7+ only

Binary to denary

- Write the place values over the 8 bits: 128, 64, 32, 16, 8, 4, 2, 1. Each column is worth double the column to its right.
- Add the place values of the columns that hold a 1. Example: 1010 0110 = 128 + 32 + 4 + 2 = 166.



128 + 32 + 4 + 2 = 166

Add the place values of the columns that hold a 1

- The leftmost bit is the **most significant bit** (MSB); the rightmost bit is the **least significant bit** (LSB).

Denary to binary

- Start at 128. If the number is at least the place value, write 1 and subtract it; if not, write 0. Move to the next column and repeat.
- Example: 93. 128 does not fit (0); 93 − 64 = 29 (1); 32 does not fit (0); 29 − 16 = 13 (1); 13 − 8 = 5 (1); 5 − 4 = 1 (1); 2 does not fit (0); 1 − 1 = 0 (1). So 93 = 0101 1101.
- Always give all 8 bits, including any leading 0s, when the question asks for 8-bit binary.

How to answer each type of question

Convert between binary and denary

1 to 2 marks | **Grade 4**

1. Write the place values 128 to 1 above the 8 bits.
2. Binary to denary: add the place values of the columns that hold a 1.
3. Denary to binary: work from 128 downwards, subtracting each place value that fits.
4. Give all 8 bits, then check by converting back.

Example. (a) Convert the 8-bit binary number 1100 1001 to denary. Show your working.
(b) Convert the denary number 150 to 8-bit binary. Show your working.

Model answer.

(a) 128 + 64 + 8 + 1 (1) = 201 (1)

(b) 150 = 128 + 16 + 4 + 2 (1), so 1001 0110 (1)

State how many values, or the largest value, n bits can store

1 mark each | Grade 5

1. Number of different patterns or values: 2^n.
2. Largest unsigned value: $2^n - 1$.
3. Read carefully: 'how many values' and 'the largest value' have different answers.

Example. A sensor stores each reading as a 6-bit unsigned binary integer.
(a) State the number of different values that can be stored.
(b) State the largest value that can be stored.

Model answer.

(a) 64 (2^6) (1)
(b) 63 (1)

Calculate the minimum number of bits needed

2 marks | Grade 7

1. Work out how many different values or codes are needed.
2. List powers of 2 until one is at least that many.
3. Show the power of 2 that is too small and the one that is big enough, then state the number of bits.

Example. A cinema gives every seat its own binary code. There are 300 seats.
Calculate the minimum number of bits needed so that every seat has a unique code. Show your working.

Model answer.

$2^8 = 256$, which is not enough (1)
$2^9 = 512$ is enough, so 9 bits (1)

Explain why computers use binary

2 marks | Grade 4

1. Say that computers are made of switches (transistors).
2. Say that each switch has only two states, on and off, which represent 1 and 0.
3. For the second mark, add that two states are much easier to tell apart reliably than ten.

Example. Explain why computers represent data in binary.

Model answer. Computers are made from switches (transistors) that have only two states, on and off, which can represent 1 and 0 (1). Two voltage levels are much easier to tell apart reliably than the ten that denary would need (1).

Don’t lose marks

- Mixing up the number of values (2^n) with the largest value ($2^n - 1$): 8 bits give 256 values, but the largest is 255.
- Leaving out leading 0s: 13 in 8-bit binary is 0000 1101, not 1101.
- Writing the place values in the wrong order, with 1 on the left instead of the right.
- Saying computers use binary 'because it is easier' without saying why: their switches have only two states.
- For minimum bits, not checking that the power of 2 is big enough, e.g. saying 8 bits for 300 values.

More tips

Memory tricks

- Place values double from the right: 1, 2, 4, 8, 16, 32, 64, 128. Write them out before every conversion.
- Odd or even check: an odd denary number always ends in 1 in binary; an even number always ends in 0.
- Any 8-bit number with a 1 in the MSB is at least 128, and 1111 1111 is 255. Use these to sense-check answers.
- Adding one bit doubles the number of patterns: 8 bits give 256, 9 bits give 512, 10 bits give 1024.
- Split 8 bits into two groups of four (nibbles) so you don't lose your place.

Exam technique

- In 'Convert' questions, write the place-value sum as well as the answer: it can earn a method mark.
- When a question says 'Show your working' for minimum bits, write the power of 2 that is too small as well as the one that is big enough.
- Check every conversion by converting your answer back the other way. It takes seconds.
- Space binary answers in groups of four (1010 0110). They are easier for you to check and for the examiner to read.

What each grade needs

What you need to be able to do, from the first marks up to the top grade.

Grade 3 — **State why computers use binary**
Computers are built from switches (transistors) that have two states, on and off, which represent 1 and 0.

Grade 3 — **Name the kinds of data stored in binary**
Numbers, text, images, sound and program instructions are all stored as binary patterns.

Grade 4 — **Convert 8-bit binary to denary**
Add the place values (128, 64, 32, 16, 8, 4, 2, 1) of every column that holds a 1.

Grade 4 — **Convert denary 0 to 255 into binary**
Work from 128 downwards, writing 1 for each place value you can subtract and 0 for each you cannot.

Grade 5 — **Find the number of patterns for n bits**
n bits give 2^n different patterns (states), so 8 bits give 256.

Grade 6 — **Find the largest unsigned value for n bits**

The largest value is $2^n - 1$, because one of the patterns is used for 0.

Grade 7

Work out the minimum bits for a requirement

Find the smallest power of 2 that is at least the number of different values needed.

Grade 8

Explain how one pattern has different meanings

The bits have no meaning on their own: the program using them decides whether they are a number, a character, a colour or an instruction.

Free notes, Full packs, practice papers and mock exams at chhetriacademy.com. Written for this site; not taken from past papers. Stuck? 1-to-1 tutoring with a free 30-minute trial lesson.