Limitations of binary representation

2.2.4 · Data · revision notes

Why a fixed number of bits limits what can be stored: the range of numbers, the precision of measurements, and the number of colours, amplitude values and characters. Expect 'explain why…' questions set in a context, often with a short calculation.

Key facts

- n bits give 2^n different values, no more
- Two's complement range: -2^{n-1} to $2^{n-1} - 1$
- Precision: a fixed number of bits gives fixed steps, so values in between are rounded
- More bits = more range or precision, but bigger files
- Unsigned range: 0 to $2^n - 1$
- Overflow: a result too big for the bits available
- Colour depth limits colours; bit depth limits amplitude values; 7 bits limit ASCII to 128 characters

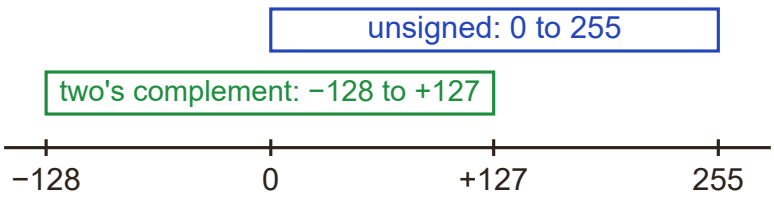
Notes

The main idea

- Every value is stored in a fixed number of bits. n bits can only make 2^n different patterns, so no more than 2^n different values can ever be represented.
- Anything that needs more values than this either cannot be stored, or has to be rounded to the nearest value that can.

Range of numbers

- 8-bit unsigned integers store 0 to 255. 8-bit two's complement integers store −128 to +127.

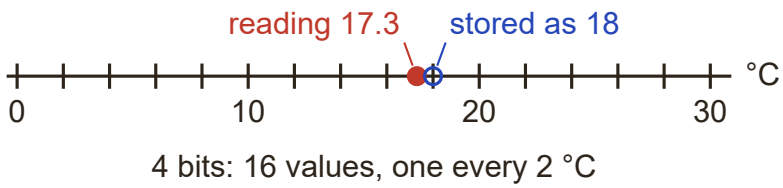


Both use 256 patterns; they just cover a different range

- A result outside the range causes an **overflow**: the extra bit is lost and a wrong value is stored. For example, an 8-bit counter cannot count past 255: adding 1 to 255 gives 0.
- The fix is to use more bits. Each extra bit doubles the number of values.

Precision

- When a measurement such as a temperature is stored in a fixed number of bits, only certain values are possible, in equal steps. A reading between two steps is rounded, so some accuracy is lost.



4 bits: 16 values, one every 2 °C

Only the tick values can be stored, so a reading between two ticks is rounded. More bits give more, closer ticks

- Shifting bits off the right-hand end loses precision in the same way: the fraction is thrown away.
- More bits give smaller steps, so values are stored more precisely.

Images, sound and text

- Colour depth: 1 bit gives only 2 colours and 8 bits only 256. A photo stored with too few bits loses shades and may show bands of colour.
- Bit depth: each sound sample must be rounded to one of 2^n amplitude values, so the stored wave is only an approximation.
- Characters: 7-bit ASCII has only 128 codes, so it cannot store accented letters, other alphabets or emoji.
- The trade-off: more bits improve range, precision or quality, but increase file size, the storage needed and the time taken to transmit the data. Grade 9+ only

How to answer each type of question

Explain why a value cannot be stored 2 marks Grade 5

1. State the largest (or smallest) value the number of bits allows.
2. Compare it with the value in the question.
3. Say what happens (an overflow, a wrong value) or what is needed (more bits).

Example. A school stores the number of students in each year group as a 7-bit unsigned binary integer. A new year group has 150 students.
Explain why this number cannot be stored.

Model answer. The largest value 7 bits can store is 127 ($2^7 - 1$) (1). 150 is greater than 127, so it would cause an overflow / at least 8 bits are needed (1).

Explain how the number of bits limits accuracy

2 marks | Grade 7

1. Say how many different values the bits allow.
2. Say that the values come in fixed steps, so readings between steps must be rounded.
3. Link to the context: what detail is lost.

Example. A light sensor measures from 0 to 1000 lux. Each reading is stored as a 4-bit unsigned integer: 0 means 0 lux, 15 means 1000 lux, and the values in between are evenly spaced.
Explain why the stored readings are not very accurate.

Model answer. 4 bits give only 16 different values to cover the whole range, so each step is about 67 lux (1000 ÷ 15) (1). Any reading between two steps has to be rounded to the nearest stored value, so small changes in the light level cannot be recorded (1).

Calculate the minimum number of bits needed

2 marks | Grade 8

1. Find the largest value (or the number of different values) that must be stored.
2. Find the power of 2 that is just too small, and the next one, which is big enough.
3. State the number of bits.

Example. A machine counts the products made during a shift. Up to 5000 products are made in one shift.
Calculate the minimum number of bits needed to store the count. Show your working.

Model answer.

12 bits can store up to $2^{12} - 1 = 4095$, which is not enough (1)

13 bits can store up to $2^{13} - 1 = 8191$, so 13 bits (1)

Discuss a trade-off

3 to 4 marks | Grade 9

1. Say what the extra bits improve (range, precision, number of colours or amplitude values).
2. Explain the benefit in the context.
3. Give the cost: bigger files, more storage, longer to transmit.
4. Explain the effect of that cost in the context.

Example. A smart doorbell stores 10-second audio clips. The designers are deciding whether to store each sample using 8 bits or 16 bits.
Discuss the advantages and disadvantages of using 16 bits.

Model answer. 16 bits give 65,536 amplitude values instead of 256 (1), so each sample is rounded less and voices are recorded more accurately (1). However, each clip needs twice as much storage (1), so fewer clips can be kept and each one takes longer to send to the owner's phone (1).

Don’t lose marks

- Writing 'the number is too big' without giving the limit for that number of bits (e.g. 127 for 7 bits).
- Saying an extra bit adds one value. It doubles the number of values.
- Forgetting that 0 uses up a pattern: 8 bits store 0 to 255, not 0 to 256.
- Saying more bits are 'better' without mentioning the cost: bigger files and more storage.
- Not linking the answer to the context in the question (the sensor, the counter, the image).

More tips

Memory tricks

- Ask 'how many patterns?' first: 2^n. Every limitation comes back to that number.
- Out of range: overflow. Between steps: rounding. Not in the set: cannot be represented.
- One more bit doubles the number of values; it does not just add one.
- Useful limits: 4 bits 15, 7 bits 127, 8 bits 255, 16 bits 65,535 (unsigned).

Exam technique

- Use numbers in your explanation: the limit (e.g. 255), the value that does not fit, and how many bits would be needed.
- In 'discuss' or 'evaluate' questions, give both sides: what extra bits improve and what they cost.
- Spot this topic in disguise: questions on overflow, colour depth, bit depth or ASCII can all be testing limitations.

What each grade needs

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

Grade 4 — **State the range of values for n bits**
Unsigned: 0 to $2^n - 1$; two's complement: -2^{n-1} to $2^{n-1} - 1$.

Grade 5 — **Explain why a value cannot be stored**
A value outside the range needs more bits than are available, so it causes an overflow.

Grade 6 — **Explain the limits of colour and bit depth**
Few bits give few colours or amplitude values, so detail is lost or values are rounded.

Grade 6 — **Explain the limits of a character set**
7-bit ASCII has only 128 codes, so it cannot represent every character in every language.

Grade 7 — **Explain precision limits when storing measurements**
A fixed number of bits gives fixed steps, so values between the steps have to be rounded.

Grade 8 — **Calculate the minimum bits for a requirement**
Find the smallest power of 2 that covers every value needed.

Grade 9 — **Evaluate trade-offs between bits, quality and size**
More bits give more range or precision but need more storage and take longer to transmit.

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.