Two's complement

2.1.2, 2.1.3 · Data · revision notes

How computers store negative whole numbers using two's complement, and how to convert between denary and 8-bit two's complement (−128 to +127). Expect 1 to 2 mark conversions and short explanations of why two's complement is used.

Key facts

- Two's complement stores negative and positive integers
- MSB 1 = negative; MSB 0 = zero or positive
- Negative denary to binary: write the positive number, invert every bit, add 1
- Grade 8+ only n-bit range: -2^{n-1} to $2^{n-1} - 1$
- 8-bit place values: −128 64 32 16 8 4 2 1
- 8-bit range: −128 to +127
- −1 = 1111 1111, −128 = 1000 0000, +127 = 0111 1111

Notes

What two's complement is

- Unsigned binary can only store zero and positive whole numbers. **Two's complement** is a way of storing **signed** integers: negative and positive whole numbers.
- In 8-bit two's complement, the most significant bit has the place value **−128**. The other place values are the usual 64, 32, 16, 8, 4, 2, 1.
- So the MSB works as a **sign bit**: 0 means zero or positive, 1 means negative.
- Positive numbers look exactly the same as in unsigned binary, with an MSB of 0.
- The 8-bit range is −128 (1000 0000) to +127 (0111 1111). −1 is 1111 1111.
- For n bits the range is -2^{n-1} to $2^{n-1} - 1$, e.g. 4 bits store −8 to +7 and 9 bits store −256 to +255. Grade 8+ only

Two's complement to denary

- If the MSB is 0, convert as normal.
- If the MSB is 1, start at −128 and add the place values of the other 1s. Example: 1011 0010 = −128 + 32 + 16 + 2 = −78.

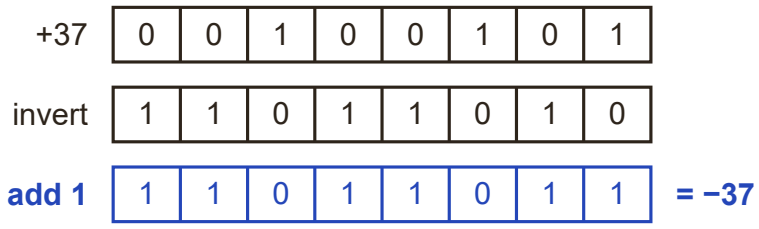
value	−128	64	32	16	8	4	2	1
bit	1	0	1	1	0	0	1	0
	MSB							LSB

−128 + 32 + 16 + 2 = −78

In 8-bit two's complement the MSB is worth −128 (red), so a 1 there makes the number negative

Denary to two's complement

- For a positive number, convert as normal. The MSB will be 0.
- For a negative number: (1) write the positive number in 8-bit binary, (2) invert every bit (0 becomes 1 and 1 becomes 0), (3) add 1.
- Example: −37. +37 = 0010 0101. Invert: 1101 1010. Add 1: 1101 1011.



Write +37, flip every bit, then add 1

- Check by converting back: −128 + 64 + 16 + 8 + 2 + 1 = −37.
- Another method: −37 = −128 + 91, so write 1 in the MSB and 91 in the other seven bits (101 1011), giving 1101 1011.

One pattern, two meanings Grade 7+ only

- 1111 0000 is 240 as an unsigned integer but −16 as a two's complement integer. The program decides which it means.
- When the MSB is 1, the unsigned value is always exactly 256 more, because that bit is worth +128 instead of −128. Grade 8+ only

How to answer each type of question

Convert a negative denary number to two's complement 2 marks Grade 6

1. Write the positive number in 8-bit binary.
2. Invert every bit.
3. Add 1, carrying as in normal binary addition.
4. Check: −128 plus the other place values should give the number you started with.

Example. Convert the denary number −90 to 8-bit two's complement binary. Show your working.

Model answer.

+90 = 0101 1010 (1)

Invert: 1010 0101; add 1: 1010 0110 (1)

Convert two's complement binary to denary

1 to 2 marks | Grade 6

1. Check the MSB: if it is 1, the number is negative.
2. Use −128 as the MSB's place value.
3. Add the place values of all the other 1s to −128.

Example. The 8-bit two's complement integer 1110 1001 is stored in a register.
Convert it to denary. Show your working.

Model answer. −128 + 64 + 32 + 8 + 1 (1) = −23 (1)

Explain why two's complement is used

2 marks | Grade 5

1. Say what in the context can be negative.
2. Say that unsigned binary can only store zero and positive numbers.
3. Say that two's complement can store both negative and positive numbers.

Example. A weather station stores the outdoor temperature, in °C, as an 8-bit integer.
Explain why the station uses two's complement rather than unsigned binary.

Model answer. Outdoor temperatures can be below 0 °C (negative) (1). Unsigned binary can only represent zero and positive numbers, but two's complement can represent both negative and positive numbers (1).

Calculate the bits needed for a signed range

2 marks | Grade 8

1. Use the range -2^{n-1} to $2^{n-1} - 1$.
2. Try values of n until both ends of the required range fit.
3. Show the n that is too small and the n that is enough.

Example. A robot stores its position along a track as a whole number of centimetres from −300 to +300.
Calculate the minimum number of bits needed to store the position as a two's complement integer.
Show your working.

Model answer.

9 bits store −256 to +255, which is not enough (1)
10 bits store −512 to +511, so 10 bits (1)

Don’t lose marks

- Only changing the MSB to 1 and leaving the rest as the positive number. 1100 0110 is −58, not −70.
- Forgetting to add 1 after inverting the bits.
- Carrying wrongly when adding 1 to a pattern that ends in 1s: 1101 0111 + 1 = 1101 1000.

- Treating the MSB as +128 when the question says the number is two's complement.
- Giving the range as −127 to +127 or −128 to +128. It is −128 to +127.

More tips

Memory tricks

- 'Flip and add 1': write the positive number, flip every bit, add 1. The same steps turn a negative number back into its positive.
- Quick method: copy the bits from the right up to and including the first 1, then flip every bit to its left. +37 = 0010 0101 becomes 1101 1011.
- Sense check: a negative answer must start with 1; a positive answer must start with 0.
- −1 is all 1s (1111 1111). The most negative number is a 1 followed by all 0s (1000 0000 = −128).

Exam technique

- Check whether the question says 'unsigned' or 'two's complement': the same bits give different values.
- Show each stage (positive number, inverted, add 1). If your final answer is wrong you can still earn a method mark.
- Always give 8 bits, and check your answer by converting it back to denary.
- In 'explain why' questions, use the context: say what can be negative (temperatures, depths, changes in a score).

What each grade needs

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

Grade 4 — **Recognise a negative two's complement number**
In two's complement, a 1 in the most significant bit means the number is negative.

Grade 5 — **State the 8-bit two's complement range**
8-bit two's complement stores whole numbers from −128 to +127.

Grade 5 — **Explain why two's complement is needed**
Unsigned binary can only store zero and positive numbers; two's complement can also store negative numbers.

Grade 6 — **Convert two's complement binary to denary**
Treat the MSB as −128 and add the place values of the other 1s.

Grade 6 — **Convert negative denary to two's complement**
Write the positive number in 8 bits, invert every bit, then add 1.

Grade 8 — **Find the range for any number of bits**
n bits store -2^{n-1} to $2^{n-1} - 1$, e.g. 4 bits store −8 to +7.

Grade 8 — **Compare a pattern's unsigned and signed values**
When the MSB is 1, the unsigned value is 256 more than the two's complement value, because that bit is worth +128 instead of −128.

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.