Binary & Hexadecimal Revision Notes

Revision notes by **FutureLogic Education** covering binary, hexadecimal, binary addition, overflow, and two's complement for IGCSE Computer Science.

What is Binary?

Binary is a base-2 number system using only **0** and **1**. Computers use binary because electronic circuits have two states — **ON (1)** and **OFF (0)**.

KEY FACT: Binary uses only 0 and 1. Circuits use ON and OFF states.

Place Values

128	64	32	16	8	4	2	1
0	1	0	1	0	1	0	1

EXAM TIP: Write the place values above every binary number BEFORE converting. Never do it in your head.

Binary to Denary

Add together the place values wherever there is a **1**.

128	64	32	16	8	4	2	1
0	1	0	1	0	1	0	1
—	*64*	—	*16*	—	*4*	—	*1*

01010101 → 64 + 16 + 4 + 1 = **85**

Teacher note: The green row shows which values are active. Ask students to add them aloud — it reinforces the process and catches arithmetic errors before they write the answer.

Denary to Binary

Work left to right across the place values. If the place value fits into your remaining number, write **1** and subtract it. If it does not fit, write **0** and move on.

Place Value	Fits into 45?	Bit	Remainder
128	No	0	45
64	No	0	45
32	Yes	1	13
16	No	0	13
8	Yes	1	5
4	Yes	1	1
2	No	0	1
1	Yes	1	0

45 in denary = **00101101** in binary

EXAM TIP: Always pad your answer to 8 bits. Cambridge expects leading zeros — 00101101 not 101101.

What is Hexadecimal?

Hexadecimal is a **base-16** number system using digits 0–9 and letters A–F. It is used because it is much shorter and easier for humans to read than long binary strings.

Denary	Hex	Binary	Denary	Hex	Binary
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	3	0011	11	B	1011
4	4	0100	12	C	1100
5	5	0101	13	D	1101
6	6	0110	14	E	1110
7	7	0111	15	F	1111

KEY FACT: Each hex digit = 4 binary bits (one nibble). A = 1010, B = 1011 … F = 1111.

Teacher note: Students only need to memorise A–F. The rest follows from counting. A quick test: cover the binary column and ask students to convert hex digits from memory.

Why Hexadecimal is Used

Reason	Explanation
Shorter than binary	8 binary bits = 2 hex digits. Far less to write and read.
Easier to read	FF is much clearer than 11111111 — reduces human error.
Easy to convert	Split binary into nibbles of 4 — each converts directly to one hex digit.
Used in computing	Colour codes (#FF5733), memory addresses, and MAC addresses all use hex.

EXAM TIP: Explain why hex is used: "Hexadecimal is shorter than binary because each digit represents 4 bits, therefore it is easier for humans to read and less prone to error."

Binary Addition

Sum	Result	Carry
0 + 0	0	0
0 + 1	1	0
1 + 1	0	1 ← carry
1 + 1 + 1	1	1 ← carry

EXAM TIP: Always write carry digits above the next column. Never track carries mentally — you will lose marks.

Teacher note: Column-by-column worked examples on the board are the most effective way to teach this. Insist students show every carry — examiners award method marks for this.

Overflow

Overflow occurs when the result of a binary addition is **too large to fit** in the available number of bits. The extra bit is lost, producing an incorrect result.

Example: 11111111 + 00000001 in 8 bits
True result = 100000000 (9 bits) — but only 8 bits available.
Stored result = 00000000 ← **This is overflow.**

EXAM TIP: If there is a carry out of the most significant (leftmost) bit, overflow has occurred.

Two's Complement

Two's complement is used to represent negative numbers in binary. There are two steps — always in this order:

Step	Action	Example (00101101 = 45)
1	Invert all bits (0→1, 1→0)	11010010
2	Add 1 to the result	11010011 (= –45)

EXAM TIP: Invert ALL bits first, then add 1. Never reverse the order.

Teacher note: The most common error is adding 1 before inverting. Drill Step 1 then Step 2 as a mantra. A worked column-by-column example prevents this mistake.

Common Exam Mistakes

✗ Forgetting place values.
Write 128 64 32 16 8 4 2 1 above every binary number before starting.

✗ Missing carries in addition.
Write carry digits above the column — never track them mentally.

✗ Mixing denary and binary.
Pause before calculating. Confirm which number system the question requires.

✗ Forgetting leading zeros.
Always pad answers to 8 bits. 101101 should be written 00101101.

✗ Incorrect hex conversions.
Split binary into nibbles of 4 from the right. Convert each nibble separately.

✗ Reversing Two's Complement steps.
Invert first, add 1 second. Never the other way around.

Examiner Tips

✔ Always show working.
Method marks are awarded even if the final answer is wrong.

✔ Write place values above numbers.
It takes seconds and prevents the most common conversion errors.

✔ Check carries carefully.
A missed carry will give a wrong answer for every bit to the left of it.

✔ Practise conversions regularly.
Speed and accuracy come from repetition — ten minutes a day is enough.

Quick Quiz

1.	Convert 10101010 into denary.
2.	Convert 45 into binary. Show your working.
3.	What is hexadecimal A in binary?
4.	What causes overflow? How can you detect it?
5.	Convert 00110011 to its two's complement.

Answers on the next page →

Quick Quiz — Answers

1.	10101010 = 128 + 32 + 8 + 2 = **170**
2.	45 = 32 + 8 + 4 + 1 = **00101101**
3.	Hexadecimal A = **1010** in binary.
4.	Overflow occurs when the result is too large for the available bits. It can be detected when there is a carry out of the most significant (leftmost) bit.
5.	Invert: 11001100 → Add 1: **11001101**

Final Summary

Topic	Key Point
Binary	Base-2. Uses only 0 and 1. Place values: 128 64 32 16 8 4 2 1.
Binary → Denary	Add place values where 1 appears.
Denary → Binary	Subtract place values left to right. Pad to 8 bits.
Hexadecimal	Base-16. Digits 0–9 and A–F. Each hex digit = 4 binary bits.
Binary Addition	1+1=10 (carry). Always write carries above the column.
Overflow	Result too large for available bits. Carry out of MSB = overflow.
Two's Complement	Step 1: Invert all bits. Step 2: Add 1. Used for negative numbers.