



1.4.1 Binary, Hexadecimal & Denary

Q1

Explain how the unsigned binary number 01011000 can be divided by 8 by using bit shift.

[2 marks]

- Shift right
- ... by 3 places

Q2

Show the byte below after having an AND applied with the masking byte.

Byte	1	0	1	1	0	1	0	1
AND	1	1	1	1	1	1	1	1
Result	1	0	1	1	0	1	0	1

Show the byte below after having an OR applied with the masking byte.

Byte	1	0	1	1	0	1	0	1
OR	1	1	1	1	1	1	1	1
Result	1	1	1	1	1	1	1	1

[2 marks]

Q3

Describe a mask that could be applied to an 8-bit number to ensure that:

- The most significant bit is always set to 1
- All other bits remain unchanged.

[2 marks]

- Use the mask 1000000
- ... and perform an OR operation





Q4

Show the result of applying an XOR mask of 1010 0111 to the byte 0011 1101:

Byte	1010 0111
XOR mask	<u>0011 1101</u>

[2 marks]

- **1001 1010** (1 mark for each 4 bits)
-

Q5

00001010 is shifted two places to the left.

a) Show the result.

[1 mark]

- **00101000**

b) Identify what arithmetic operation this shift is equivalent to..

[1 mark]

- **Multiplying by 4**
-

