1	**1**	**Mark is for AO1 (knowledge)** (Using an algorithm) to convert a message into a form that is not understandable (without the key to decrypt it); (Using an algorithm) to convert a message into a form that is only understandable by the intended parties // can only be read with the correct key; (Using an algorithm) to convert a message into cipher text; **N.E.** Scrambling unless further explanation is provided **N.E.** Coding **A.** "Unreadable" for "understandable" **A.** "Data" for "a message" **R.** Responses that do not make clear that encryption is a process **Max 1**	**1**
1	**2**	**Marks are for AO1 (understanding)** Vernam cipher (if implemented correctly) is unbreakable / harder to crack / Caesar cipher can be easily cracked; Frequency / statistical analysis of ciphertext reveals nothing about plaintext; More possible keys; Vernam cipher does not always translate a ciphertext character to the same plaintext character (removing repeated patterns); **A.** Points made in reverse, ie as disadvantages of the Caesar cipher **Max 2**	**2**

1	**3**	**Marks are for AO2 (apply)** **1 mark** for identifying 1001000 1001111 1000111 as the binary representation of 'HOG' **1 mark** for final result being 21 bits long; **R.** if result is the same as HOG (1001000 1001111 1000111) or SON (1010011 1001111 1001110) **1 mark** for correct application of XOR; 0011011 0000000 0001001 **A.** follow through mistakes	**3**

2	1	**Mark is for AO2 (analyse)** Subtract 48 / 00110000 from the character code / bit pattern; AND the character code / bit pattern with the bit pattern 00001111; XOR the character code / bit pattern with the bit pattern 00110000; **Max 1**	1
2	2	**Marks are for AO1 (understanding)** Introduced to support a larger range of characters; Due to increased international communication // use of files in multiple countries; **A.** sensible alternatives to international communication: eg facilitate interchange of documents between countries. eg culturally unacceptable to only allow non-English speaking countries to communicate in English **NE.** use in other countries or examples of this. Each character code is always interpreted as the same character; **Max 2**	2

3	1	**Marks are for AO1 (understanding)** Parity bits can only detect errors not correct them // Majority voting can correct (most) errors that occur during transmission; Majority voting can detect multiple (bit) errors; Majority voting is more efficient at detecting errors; Majority voting can (sometimes) detect an even number of errors; **Max 1**	1

Qu		Marks	
4	1	**Mark is for AO2 (analyse)** C;	1
5	1	**2 marks are for AO1 (knowledge)** Consists of a digit calculated (using an algorithm); from the other digits/letters (in the input sequence); **A.** Answer by example.	2

6	1	**2 marks are for AO1 (understanding)** If the number of 1s received/in the byte is even, the data is (assumed to have been) received correctly // has not been corrupted; **A**. the data is correct If the number of 1s received/in the byte is odd, the data has been corrupted / is incorrect; **A.** odd/even part of second point by implication eg if student has written “is even” for the first point and then “otherwise” for the second.	2

Qu	Pt	Marking Guidance	Marks
7	1	**Mark is for AO1 (knowledge)** A (unique) number used to represent a character; **R.** code	1

Qu	Pt	Marking Guidance	Marks
7	2	**Marks are for AO1 (understanding)** Introduced to support a larger range of characters; Due to increased international communication // use of files in multiple countries // requirement to use additional symbols (allow examples, eg mathematical / scientific / engineering / emoji symbols) // facilitates interchange of documents between countries // culturally unacceptable to only allow non-English speaking countries to communicate in English// (concurrent) support for multiple languages; **A.** representation of characters in languages other than English (using codes that are globally unique). **MAX 2**	2

Qu	Pt	Marking Guidance	Marks
7	3	**Marks are for AO1 (understanding)** The number of 1s is summed / counted; if the total is even, the parity bit is set to 0, otherwise it is set to 1 // if the total is odd, the parity bit is set to 1, otherwise it is set to 0 // the parity bit is set to ensure the total number of 1s is even; The bits are XOR'd with each other; and the result is the parity bit; **MAX 2**	2

Qu	Pt	Marking Guidance	Marks
7	4	**Mark is for AO2 (application)** 0;	1

Qu	Pt	Marking Guidance	Marks
7	5	**Marks are for AO2 (application)** Showing that 'EGG' is represented by 1000101 1000111 1000111; Providing a 21-bit answer that is not 'DAB' or 'EGG'; Correct answer (reached by applying XOR): 0000001 0000110 0000101; **A.** Correct result of XORing 1000100 1000001 1000010 with an incorrect representation of 'EGG'.	3

8	**1**	**All marks AO2 (apply)** Stop Bit, Parity Bit, Start Bit **0 1 0 1 1 0 1 0 0 1** **1 mark:** Start bit and stop bit each have the value 1 and 0 and must be different to each other (it does not matter which is 1 or 0). **1 mark:** Correct bit pattern for character `'4'`: 0110100 **1 mark:** Parity bit is 1 **A.** parity bit of 0 if this would be correct based upon an incorrect ASCII code used and reject parity bit of 1 if would not be correct for ASCII code used.	**3**

8	**2**	**All marks AO1 (understanding)** **Improvements (Max 3):** #Errors can (sometimes) be corrected as well as detected; **A.** the location of an error can be identified #Multi-bit errors can be detected; **A.** errors that change an even number of bits can be detected *If neither of the points marked # is awarded then award one mark if the general point that transmissions should be more reliable is made.* A greater range of characters can now be transmitted; **A.** any response that implies this eg support for multiple languages // languages with large sets of characters, inclusion of specialised symbols in character set Elimination of problems caused by different versions of ASCII character sets / extended ASCII / use of code pages // eliminates problem of some ASCII codes representing different characters in different countries // Unicode values can be interpreted more consistently than ASCII codes; **Disadvantages (Max 3):** *Each character will require more bits // 8 bits // 16 bits // 32 bits // between 8 and 32 bits; *Each bit will be sent multiple times // three or more times // there will be redundancy in the data transmissions; **A.** code, character instead of bit *If neither of the points marked * is awarded then award one mark if the general point that more bits are required is made.* The (effective) rate at which information / (useful) data can be transmitted will be reduced; **A.** transmissions will take longer **R.** references to storage space	**4**

Qu	Pt	Marking guidance	Total marks
9	1	**All marks AO1 (understanding)** **Explain what the checksum is used for:** To check if the contents of the packet/data have been corrupted/changed (during transmission) // to check if the received data is/is not the same as the transmitted data; **A.** "tampered" for "corrupted/changed" **A.** To check if an error has occurred during transmission **A.** To check if the data has been sent/transmitted correctly **NE.** To check if received/transmitted data is correct **NE.** To correct (some) errors in the received data **NE.** Error checking **Outline how the checksum's value will be determined:** Calculated from the payload/data/contents (of the packet); **A.** Hash/apply a function to the payload/data/contents of the packet **A.** Explanation of a reasonable calculation that could be done **NE.** Explanation of a calculation that could not reasonably be performed to produce a useful checksum **NE.** "apply an algorithm to data" unless clear that this is mathematical, or produces a single value as an output **I.** Responses that go on to talk about a comparison being made using the checksum to check if the data is received correctly.	**2**

Qu	Pt	Marking guidance	Total marks
10	1	**Mark is AO2 (apply)** Award **1 mark** for correct value in R0: R1: 0 1 0 0 0 1 1 0 15: 0 0 0 0 1 1 1 1 R0: 0 0 0 0 0 1 1 0 **R.** Any cells of R0 left empty	1

Qu	Pt	Marking guidance	Total marks
10	2	**Mark is AO2 (apply)** Award **1 mark** for correct value in R0: R1: 0 0 0 0 0 1 1 0 48: 0 0 1 1 0 0 0 0 R0: 0 0 1 1 0 1 1 0 **R.** Any cells of R0 left empty	1

Qu	Pt	Marking guidance	Total marks
10	3	**4 marks AO2 (analysis) and 6 marks AO3 (programming)** **6 marks AO3 (programming syntax must be correct):** *MP1:* Value in memory location 100 is loaded into a register; *MP2:* After some manipulation has been carried out (whether correct or not) values are stored into memory locations 101 and 102 (do not award if it is the same value stored twice); *MP3:* Binary pattern of one digit correctly isolated from the input value (for leftmost digit must also be shifted so bits in correct place); *MP4:* Binary pattern of one digit correctly translated into ASCII for one of numeric digits **or** letter digits (ignore if the pattern is later changed again to be incorrect); *MP5:* Binary pattern of one digit correctly translated into ASCII for both numeric digits **and** letter digits (ignore if the pattern is later changed again to be incorrect); *MP6:* Conversion process fully working for the both digits (ASCII codes must be correct when program terminates); **Note:** If MP3 not awarded MP4, MP5, MP6 cannot be awarded **A.** Any understandable method for identifying labels **DPT.** Use of invalid register names eg R27, Rn **DPT.** Use of binary for immediate operand values **DPT.** Omission of # to indicate immediate operand values **DPT.** R before memory address eg R100 **DPT.** Use of MOV instead of LDR or STR, or vice-versa **DPT**. Repeated use of incorrect delimiters eg ; < > . " ' (occasional errors can be ignored) **4 marks AO2 (concept must be understood, syntax need not be correct):** *MP7:* Attempt to use masking **and/or** shifting to identify one digit; *MP8:* Attempt to use masking **and/or** shifting a second time to identify the second digit; *MP9:* Attempt to use comparison and branching to make program treat numeric digits and letter digits differently for at least one of the two digits (whether threshold values correct or not); *MP10:* Use of addition or masking to attempt to convert a digit to an ASCII code (whether correct ASCII codes produced or not); **Note:** If MP3 not awarded MP10 cannot be awarded **Max 9 if solution not fully working**	10

Example Solution 1

```
        LDR R0, 100             MP1
        AND R2, R0, #15         MP7, MP3
        CMP R2, #10
        BLT isnumber            MP9
        ADD R2, R2, #55         MP10, MP4
        B doleftdigit
isnumber:
        ADD R2, R2, #48         MP5
doleftdigit:
        AND R1, R0, #240        MP8
        LSR R1, R1, #4
        CMP R1, #10
        BLT isnumber2
        ADD R1, R1, #55
        B storetomemory
isnumber2:
        ADD R1, R1, #48         MP6
storetomemory:
        STR R1, 101
        STR R2, 102             MP2
```

Example Solution 2

```
        LDR R0, 100             MP1
        AND R0, R0, #15         MP7, MP3
        CMP R0, #9
        BGT isletter            MP9
        ORR R0, R0, #48         MP10, MP4
        B doleftdigit
isletter:
        SUB R0, R0, #9
        ORR R0, R0, #64         MP5
doleftdigit:
        STR R0, 102
        LDR R0, 100
        LSR R0, R0, #4          MP8
        AND R0, R0, #15
        CMP R0, #9
        BGT isletter2
        ORR R0, R0, #48
        B finish
isletter2:
        SUB R0, R0, #9
        ORR R0, R0, #64         MP6
finish:
        STR R0, 101             MP2
```

Example Solution 3

```
        LDR R1, 100             MP1
        LSR R2, R1, #4          MP7, MP3
        LSL R1, R1, #4
        LSR R2, R1, #4          MP8
        CMP R2, #10
        BLT number              MP9
        ADD R2, R2, #7          MP10
number:
        ADD R2, R2, #48         MP4, MP5
        STR R2, 101
        CMP R2, #10
        BLT number2
        ADD R2, R2, #7
number2:
        ADD R2, R2, #48         MP6
        STR R2, 102             MP2
```

Qu	Pt	Marking guidance	Total marks
11	1	**Mark is AO2 (analysis)** 2; **A.** Number not stated but identified that 4 and 5 are the valid patterns **I.** Incorrect patterns stated if correct answer 2 given	1

Qu	Pt	Marking guidance	Total marks
11	2	**Mark is AO1 (understanding)** Receiver and transmitter (continuously) synchronised by a common clock // timing information transmitted within/alongside the data // receiver and transmitter clocks are (continuously) synchronised; **A.** Both devices synchronised by same clock **NE.** Receiver and transmitter are synchronised **NE.** Transmission synchronised to a clock signal	1

Qu	Pt	Marking guidance	Total marks
11	3	**Mark is AO1 (understanding)** Errors that change an even number of bits (**A.** two bits) cannot be detected; **R.** multi-bit errors cannot be identified (Errors can be detected but) errors cannot be corrected; **A.** Position of errors cannot be identified **Max 1**	1

Qu	Pt	Marking guidance	Total marks
11	4	**Mark is AO1 (knowledge)** **A**; (Line A) **R.** If more than one lozenge shaded	1