Solutions to Assignment 1

Question 1

(a) The entropy of this message source in bits per message is

$$\sum_i P_i \log_2(P_i) = 1.78$$

which can be conveniently computed with a spreadsheet:

	A	B
1	45%	=A1*LOG(A1,2)
2	30%	=A2*LOG(A2,2)
3	15%	=A3*LOG(A3,2)
4	10%	=A4*LOG(A4,2)
5		=SUM(B1:B4)

(b) If one wind direction report is generated every two seconds, the information rate of the source in bits per second is 1.78 bits / 2 seconds = 0.89 bits/second.

(c) If the best possible compression method was used, the data rate would be the same as the information rate: 60 minutes/hour $\times$ 60 seconds/minute $\times 0.89 = 3208$ bits or about 401 bytes. This many bytes would be transmitted in one hour and would have to be stored.

(d) If each direction was encoded using 2 bits per message then $60 \times 60 \times 2/2 = 3600$ bits (450 bytes) would be required to store these messages.

(e) (i) The information rate depends only the message probabilities so it would not change. (ii) The data rate increases to 2 bits / 2 seconds = 1 bps when each message is encoded using two bits per message.

Question 2

The duration of each frame, including the header (32 bits) plus data payload (8×8=64 bits) transmitted at 500 kbps and the 20 μs gap between frames is:

$$\frac{32+64}{500 \times 10^3} + 20 \times 10^{-6} = 212\ \mu\text{s}$$

To compute the throughput of the highest-priority control unit we divide the useful payload bits delivered by this unit by the total time required to transmit the frame.

In this case one out of every 4 frames is used by the highest-priority unit so on average the time required to transmit one frame is 4 frames. The duration of 4 frames is $4 \times 212 = 848\ \mu$s.

The throughput is thus:

$$\frac{64 \text{ bits}}{848 \times 10^{-6} \text{ bits/s}} \approx 75.5 \text{ kbps}$$

Question 3

The Unicode character "`CANADIAN SYLLABICS SH`" has a code point of U+1525. In binary this is 0001 0101 0010 0101 which must be encoded as three bytes using the third row of Table 3-6 in the Unicode Standard. In this case zzzz is 0001, yyyy yy is 0101 00 and xx xxxx is 10 0101. The binary values of the three bytes in the UTF-8 encoding are thus: 1110 0001, 10 01 0100 and 10 10 0101 which in hex are 0xE1, 0x94 and 0xA5.

Question 4

(a) Each answer will be different. For example, the ID number 123456 (base 10) can be converted using a calculator to hexadecimal: $1E240_{16}$. To make a 32-bit number we add leading zeros to make up 8 hex digits: 0001 E240. In binary this is:

0000 0000 0000 0001 1110 0010 0100 0000

(b) The same number using base-16 (hexadecimal) notation is 0x1E240 or with leading zeros (which don't change the value): 0x0001 E240.

(c) When a number is stored in memory or transmitted using little-endian byte order the bytes are stored or transmitted in order from the LS to MS byte. In this example the value of the four bytes (in hexadecimal notation) are:

40, E2, 01, 00

(d) The corresponding bits written in conventional (msb-first) order (and the bytes still in little-endian byte order) are:

0100 0000, 1110 0010, 0000 0001, 0000 0000

Writing the bits of each byte in lsb-first order we have:

0000 0010, 0100 0111, 1000 0000, 0000 0000

Question 5

The character 'A' (0x41) with 7 bits/character and no parity would be sent as a start bit (positive, 0), 7 data bits in lsb-first order (1000 001), and a stop bit (negative, 1). The transmitted waveform would thus be:

$$S - + + + + + - T$$

where S represents the start bit (high), $+$ and $-$ represent the data bits (positive or negative respectively) and T represents the stop bit (low). The bit duration is $\frac{1}{4800} = 208\ \mu$s. If the modem's UART was set to 19200 bps it would wait for a rising edge and then sample the waveform four times faster, seeing the waveform:

$$++++ ---- ++++ ++++ ++++ ++++ ++++ ---- ----$$

which would be interpreted, if the UART was configured for a start bit, 8 data bits and a stop bit as:

$$S{+}{+}{+}\ ----\ {+}FFF\ FFFF\ FFFF\ FFFF\ FFFF\ MMMM\ MMMM$$

where F indicates a framing error (a high level where a stop bit was expected) and M indicates a mark level in-between characters. The received bits are 000 1111 0 in lsb-first order or 0111 1000 in msb-first order or 0x78 which is a lower-case '**x**' in ASCII or UTF-8 encodings.

The modem would detect a framing error after the character because the input would be high at the location where a stop bit (low) was expected (at the first bit position marked F above). The modem would wait for the next rising edge of the waveform (which would be the next character in this case).