Solutions

Q1a

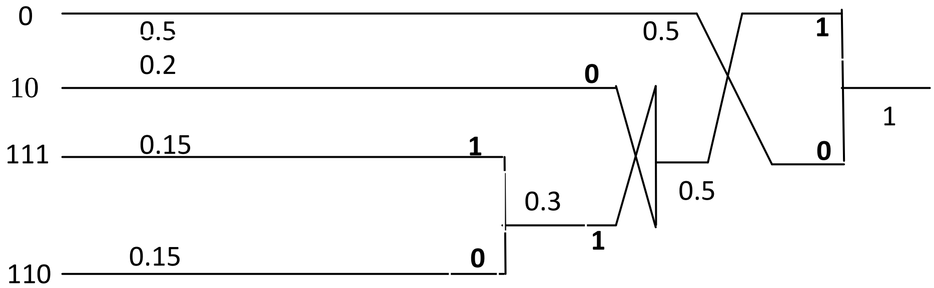
In binary signalling we transmit only one bit of information per symbol (i.e. either a '1' or a '0'). **(1 mark)**

In M-Ary signalling we send a symbol, which can take one of M different states (levels, frequencies, phases). **(1 mark)**

Q1b

In the AM DSB SC system, all of the signal's power is contained in the same phase, whereas the noise is evenly distributed between the in-phase and out-of–phase components. **(2 marks)**
The out-of-phase components get rejected during demodulation and so we lose half the noise. Therefore, the signal to noise ratio improves by a factor of two. **(2 marks)**

Q1c



-

(2 marks)

Computer code	Probability	Huffman code
00	0.5	0
01	0.2	10
10	0.15	111
11	0.15	110

(1 mark)

Number of bits used in the computer code is

$$2(0.5+0.2+0.15+0.15)=2 \text{ bits}$$ **(1 mark)**

Number of bits used in the Huffman code is

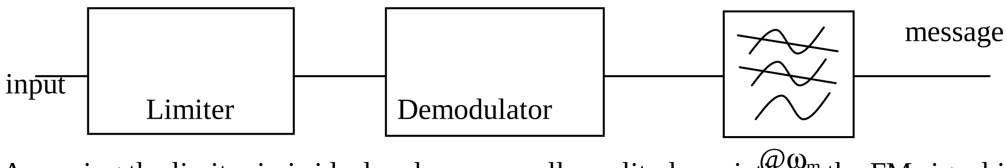
$$1\times 0.5+2\times 0.2+3\times 0.15+3\times 0.15=1.8 bits$$ **(1 mark)**

The percentage reductions in number of bits

$$100\times\frac{0.2}{2}=10\%$$ **(1 mark)**

Q1d

Consider following FM receiver



(1 mark)

Assuming the limiter in is ideal and removes all amplitude variations, the FM signal is represented by,

$$s_i(t)=\alpha\cos\left(\omega_c t+c\int_0^t f(\tau)d\tau\right).$$ **(1 mark)**

Here *f(t)* is the modulation, α the signal amplitude and *c* is a constant dependent upon the transmitting system. The instantaneous frequency is the derivative of the phase,

$$\frac{d}{dt}\left(\omega_c t + c\int_0^t f(\tau)d\tau\right) = \omega_c + cf(t).$$ **(1 mark)**

The output of the demodulator is the difference in this instantaneous frequency and the carrier frequency. Therefore the signal output is,

$$\begin{aligned} s_o(t) &\propto \omega_c + cf(t) - \omega_c \\ s_o(t) &= Kcf(t) \end{aligned}.$$ **(1 mark)**

Here K is a constant that is dependent upon the receiver. Now let us say that our signal is a simple single frequency tone, ω_m, for example a whistle, in this case $f(t) = A\cos(\omega_m t)$. Substituting this into the first equation of this section,

$$s_i(t) = \alpha\cos\left(\omega_c t + \frac{cA}{\omega_m}\sin(\omega_m t)\right)$$ **(1 mark)**

Looking at this equation we can see that the maximum frequency deviation is cA, and so we define $cA = \Delta\omega$ for convenience. We also find it useful to denote $\Delta\omega/\omega_m = \beta$, the modulation index. The above then becomes,

$$s_i(t) = \alpha\cos(\omega_c t + \beta\sin(\omega_m t)).$$ **(1 mark)**

Similarly, we can say, $s_o(t) = K\Delta\omega\cos(\omega_m t)$. **(1 mark)**

The time averaged output signal power can be found quite easily,

$$S_o = \frac{1}{T}\int_{-T/2}^{T/2} s_o{}^2(t)dt = \frac{K^2\Delta\omega^2}{2}.$$ **(1 mark)**

Q2a

Lossless compression is used when every byte of the data is important, such as executable programmes and sources codes. **(1 mark)**

Run length coding is an example of lossless coding where the data is compressed in such a way that original data may be reconstructed exactly from the compressed data. **(1 mark)**

Lossy data compression is used when compressing a file and then decompressing it retrieves a file that may be different from the original but is close enough to be useful. It is used with data such us sounds and pictures, where a small loss of quality can be tolerated without losing the essential nature of the data. **(1 mark)**

The advantage of lossy over lossless compression is that in some cases a lossy compression can produce a much smaller compressed file than any known lossless method, while still meeting the requirements of the application. **(1 mark)**

Q2b

The probability of n bit errors in an N bit long symbol can be calculated using the binomial theorem,

p(n errors in an N bit word) = p(n,N) $= \binom{N}{n} P_E{}^n (1 - P_E)^{(N-n)}$

Number of possible ways n errors could occur in an N bit word

p(n bits are in error)

p(N-n bits are not in error)

where $\binom{N}{n} = \frac{N!}{(N-n)!n!}$. **(1 mark)**

when $P_E = 1\times10^{-9}$, a typical bit error rate on optical fibre systems.

$$p(1,5) = \binom{5}{1} \times 10^{-9} \times \left(1-10^{-9}\right)^4 \approx \binom{5}{1} \times 10^{-9} = 5 \times 10^{-9}$$

$$p(2,5) = \binom{5}{2} \times \left(10^{-9}\right)^2 \times \left(1-10^{-9}\right)^3 \approx \binom{5}{2} \times 10^{-18} = 1 \times 10^{-17}$$ **(2 marks)**

$$p(3,5) = \binom{5}{3} \times \left(10^{-9}\right)^3 \times \left(1-10^{-9}\right)^2 \approx \binom{5}{3} \times 10^{-27} = 1 \times 10^{-26}$$

Notice how $p(n,N)$ varies approximately as $P_E^{\,n}$, hence a one bit error is vastly more likely than a two bit error and so on. Note that this approximate relationship will become a worse approximation as P_E increases. **(1 mark)**

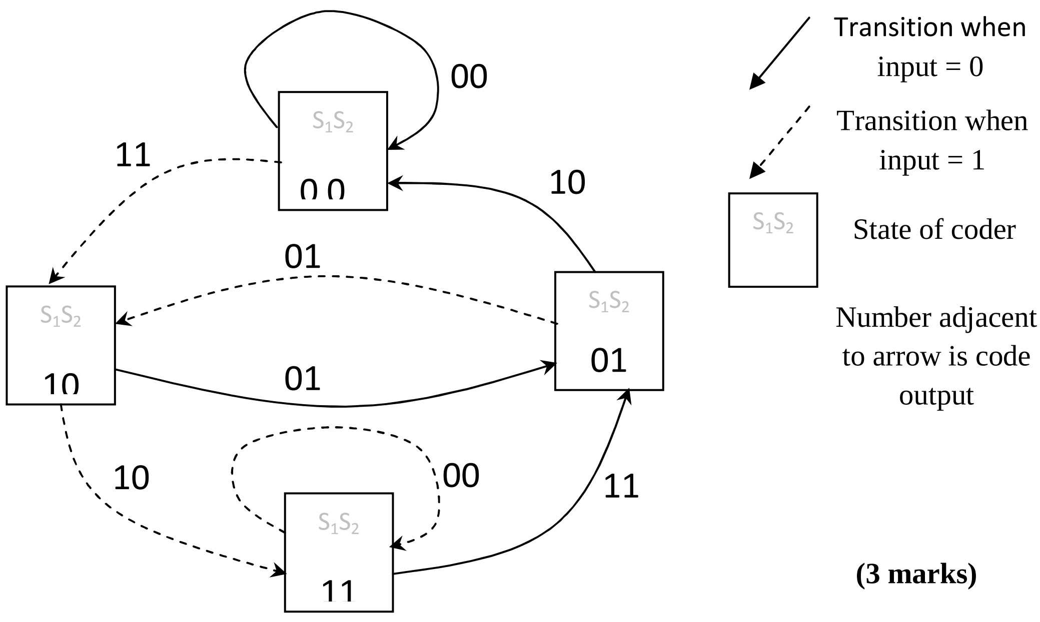
Q2c

i.

Input S_0	Initial state S_1S_2	Next state	Output bits u_1	u_2
0	00	00	0	0
1	00	10	1	1
0	10	01	0	1
0	01	00	1	0
1	00	10	1	1
0	10	01	0	1
1	01	10	0	1
0	10	01	0	1
1	01	10	0	1

(3 marks)

ii.



(3 marks)

Q2d

We can make a decision as to what bit is received based upon the pdfs we have just defined. One way would be to say that the intended message when we receive an amplitude a is the message with the highest pdf for that amplitude. This is what is known as a hard decision, and is usually adequate channels with small amounts of AWGN. **(1 mark)**

There is however a chance that we could guess wrong. Say a logic 0 was sent, and the noise was such that we received $a > a_m$ then we would incorrectly decode that message as a 1. The probability of this happening is,

$$p(false\_1) = p_0(a > a_m) = \int_{a_m}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-\{a - a_0\}^2}{2\sigma^2}\right) da$$ **(1 mark)**

Similarly the chance that we detect a false 0 is,

$$p(false\_0) = p_1(a < a_m) = \int_{-\infty}^{a_m} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-\{a - a_1\}^2}{2\sigma^2}\right) da \, .$$ **(1 mark)**

Generally we can write the average error probability for a system with N messages as,

$$P_E = \sum_{n=0}^{N} p(n) r(n) .$$

where $p(n)$ is the probability of the message n being sent, and $r(n)$ is the probability of the message n being incorrectly detected. **(1 mark)**

In the case of simple binary system, $r(0) = r(1) = 0.5$, i.e. there are an even number of '1's and '0's. Hence, for the simple binary system above,

$$P_E = \frac{1}{2}\left(p\{false\_0\} + p\{false\_1\}\right).$$

Most of the time, σ which describes p_0 and p_1 are the same and so p(false 0) = p(false 1) = P_E.
(1 mark)

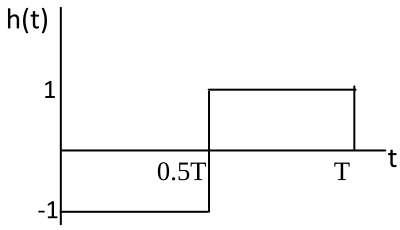
When the degree of AWGN noise is high (i.e. when σ is large), it is usually the case that soft decisions are made, i.e. the decisions made have a confidence value attached to them. For example if we received a signal with a just smaller than a_m, a soft decision would return a value of logic 0, but with a small degree of confidence. **(1 mark)**

Q3a

The impulse response of the match filter is given by

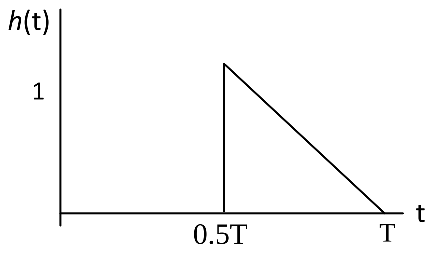
$$h(t) = ks(t_0 - t)$$

i.e., the response of the matched filter for the signal in Figure 2 is



(2 marks)

and the response of the matched filter for the signal in Figure 3 is



(2 marks)

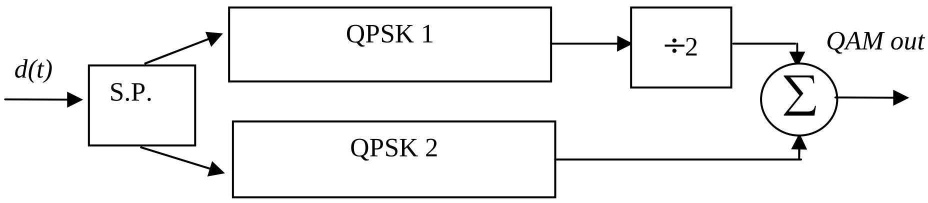
Q3b

Matched filters in digital applications have a subtly different job to perform than analogue filters. In analogue systems, e.g. AM radio, information is contained within the wave shape, and so the filter must preserve this as much as possible. **(2 marks)**

A digital system, by contrast, conveys information by the presence or absence of a signal, and so we do not need to preserve the waveshape. Instead the aim is to maximize the SNR at the filter's output at time $t=t_o$. **(2 marks)**

Q3c

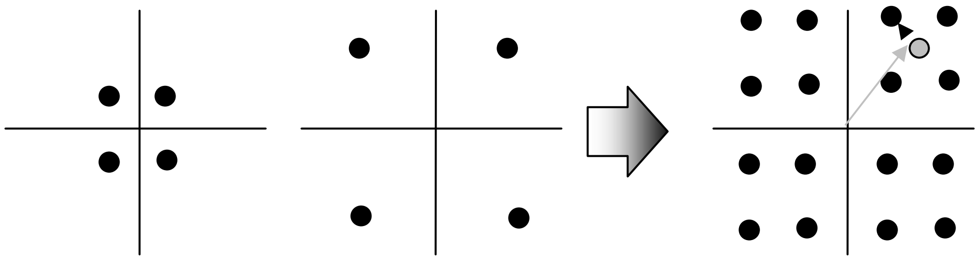
The following circuit can be used for the generation of 16-Ary QAM.



(2 marks)

The signal processing block (S.P.) splits the incoming four bit symbol into two separate 2 bit symbols. **(1 mark)**

The outputs from the QPSK modulators are as follows.



o/p from QPSK 1   o/p from QPSK 2   Vector addition to yield QAM

(1 mark)

We combine the 2 QPSK outputs to obtain 4×4 = 16 possible outputs. **(1 mark)**

Q3d

Allows us to transmit twice the information as BPSK in the same bandwidth as it uses both in-phase and quadrature carriers. The message waveform is as follows,

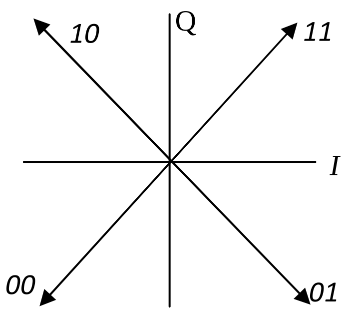
$$s(t)=\frac{A}{\sqrt{2}}d_1(t)\overset{\substack{\text{In phase}\\ \downarrow}}{\cos}(\omega_c t)+\frac{A}{\sqrt{2}}d_2(t)\overset{\substack{\text{Quadrature}\\ \downarrow}}{\sin}(\omega_c t).$$

Where $\begin{matrix} d_1(t)=\pm 1 \\ d_2(t)=\pm 1 \end{matrix}$, i.e. contains 1 bit of information. **(1 mark)**

If you look at the message waveform, we can see that the phase of the QPSK signal is given by

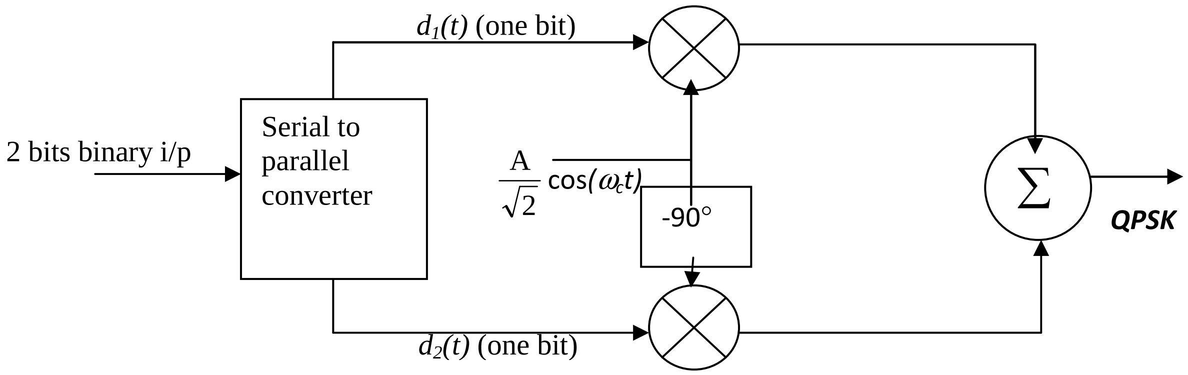
$$\theta=\tan^{-1}\left(\frac{d_2(t)}{d_1(t)}\right).$$ **(1 mark)**

Therefore we have four possible outcomes, $\pm 45^\circ, \pm 135^\circ$. Hence the phasor diagram is as follows,



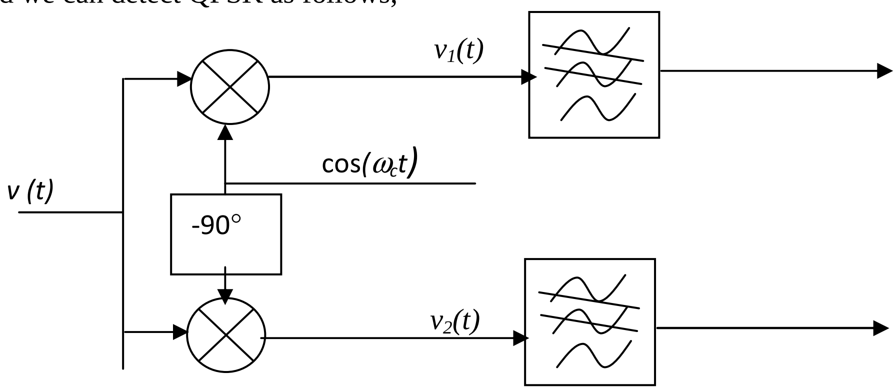
(1 mark)

One can generate QPSK by the following circuit.



(1.5 marks)

And we can detect QPSK as follows,



(1.5 marks)

Let us consider the top channel,

$$v(t)\cos(\omega_c t)$$

$$v_1(t) = \frac{A}{\sqrt{2}} d_1(t)\cos^2(\omega_c t) + \frac{A}{\sqrt{2}} d_2(t)\cos(\omega_c t)\sin(\omega_c t) .$$

$$\frac{A}{\sqrt{2}} d_1(t)\frac{1}{2}[1+\cos(2\omega_c t)] + \frac{A}{\sqrt{2}} d_2(t)\frac{1}{2}[\sin(0)+\sin(2\omega_c t)]$$

Output from the low-pass filter is thus $Ad_1(t)/2\sqrt{2}$.

Similarly, the output from the bottom channel is $Ad_2(t)/2\sqrt{2}$. **(1 mark)**

Q4a

Pseudonoise (PN) code is a signal which appears to be random noise but are in fact completely deterministic. **(1 mark)**

A good PN sequence is the one that appears to be random, in other words it should have similar properties to true random sequences:

a) Balance: We want roughly the same number of 0s as we have 1s **(1 mark)**

b) Run sequence: A run sequence is a sequence of consecutive bits with the same value. To be noise-like we want

1/2 of run sequences to be of length 1

1/4 of run sequences to be of length 2

1/8 of run sequences to be of length 3… **(1 mark)**

c) Autocorrelation

We would like the PN sequence to have very low correlation with shifted copies of itself. **(1 mark)**

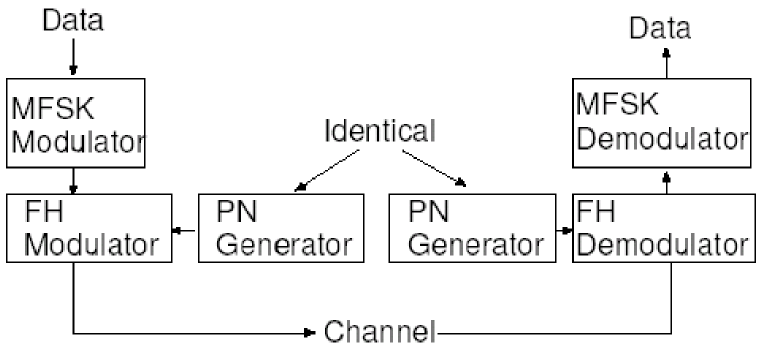
Q4b

One problem of using maximal length codes is that sometimes they have good correlation properties between difference PN codes, so called cross-correlation. This can be a problem, since we don't want two signals being unlocked by a PN code, we only want one (our signal). To combat this problem, 'Gold codes' which have specially designed cross correlation properties are used. **(2 marks)**

Maximal length sequences can also suffer from another problem. Ideally we want the autocorrelation properties of a PN code to be very good, i.e. for the autocorrelation function to be 2-valued (large or close to zero). Maximal length sequences do not always have this property. **(2 marks)**

Q4c

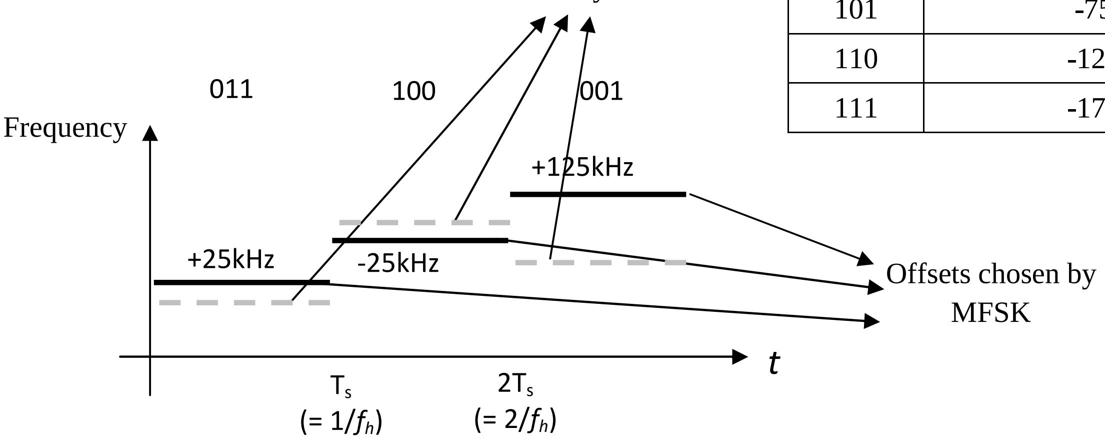
Frequency hopping spreads the spectrum transmitting the data signal on one of a series of carriers which occupy a very large bandwidth. The specific carrier is chosen by a PN code which changes with a rate, f_h, also known as the hop rate. For S.S. we require that the bandwidth containing the carrier frequencies is much larger than the data bandwidth. FHSS is usually achieved using the following type of circuitry,



(2 marks)

Again, the boxed sections perform the spreading and de-spreading of the message waveform. We have discussed MFSK modulation briefly in the M-Ary section of the course, it involves assigning a frequency to a particular symbol (sequence of bits), as shown in the example over. The changing PN sequence then selects the carrier frequency, which is shown schematically below. **(2 marks)**

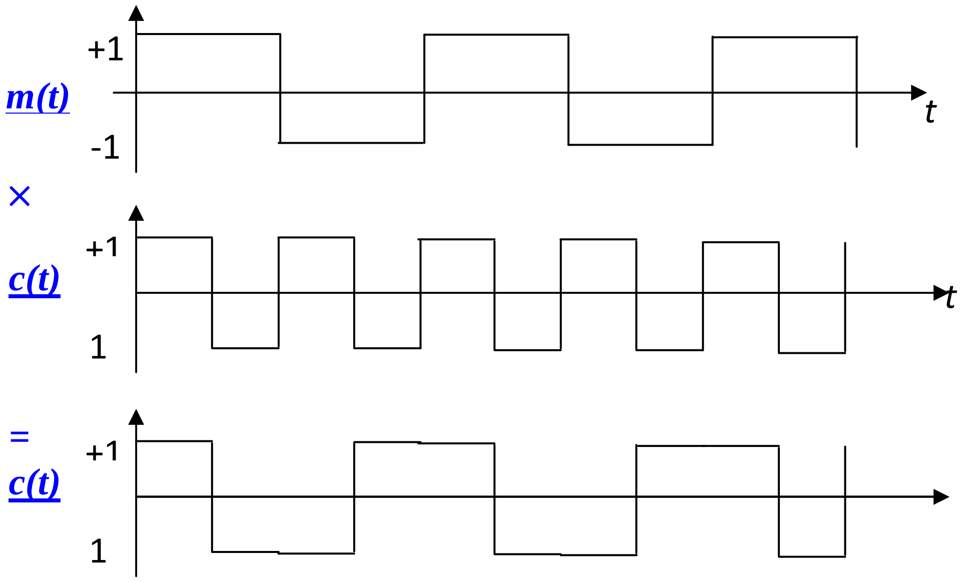
Data	Carrier offset (kHz)
000	175
001	125
010	75
011	25
100	-25
101	-75
110	-125
111	-175



Hopping (i.e. the span over which the PN code can move the MFSK signal) occurs typically over several GHz. **(2 marks)**

Q4d

Assume a 4 bit PN sequence of (1010) (Any PN sequence can be assumed since none is given in the question) then the resulting waveforms are **(1 mark)**



(1 mark)

(2 marks)

(2 marks)