CHAPTER 4

DIGITAL Q^{2}PSK — Modulation, Demodulation and Synchronisation

In this chapter the DSP design and implementation of the Q^{2}PSK modulator, demodulator and synchronisation processes are considered. Furthermore, detail concerning the block transmission strategy adopted is presented. The different levels of synchronisation, namely carrier, symbol and frame synchronisation are considered. In addition, the sensitivity of the demodulator to phase and frequency synchronisation errors is discussed. The final part of this chapter is concerned with the analysis and design of carrier tracking algorithms. Specifically, for carrier frequency (Doppler) and phase tracking, the recursive scalar Kalman estimator is proposed and analysed. Two methods are proposed. The first method employs the Kalman estimator directly to estimate the carrier frequency information. The second method, the so-called Dual Phase and Frequency Kalman Estimator (DPFKE), employs two scalar Kalman estimators to simultaneously track both carrier frequency and phase information.

4.1 MODEM REALISATION

4.1.1 Digital Q^{2}PSK Modulator

The digital modulator implemented is shown in Figure 4.1 and can be viewed as two parallel quadrature modulations, followed by another quadrature modulation. The first set of modulations is related to data shaping pulses, $q_1(t)$ and $q_2(t)$ as defined in Chapter 2, equation (2.5). The second modulation translates the outputs of the first modulation onto quadrature carriers at a carrier frequency of $f_c = \omega_c/2\pi$ Hz.

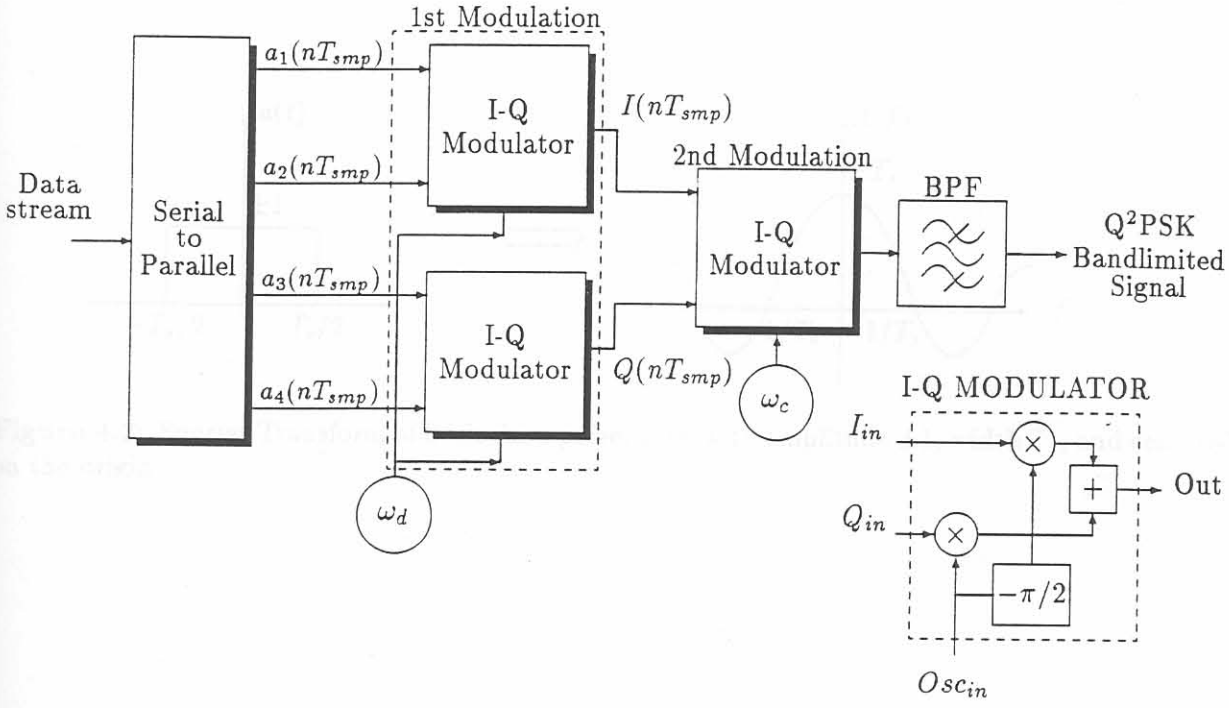

Figure 4.1: Q²PSK Transmitter block diagram.

The Serial-to-Parallel (S/P) converter accepts serial data and following parallel conversion, the information bits are multiplied by the data shaping pulses, which are expressed as:

$$\begin{aligned} I(nT_{smp}) &= [a_1(nT_{smp}) - ja_2(nT_{smp})]\, e^{j2\pi f_d nT_{smp}} \\ Q(nT_{smp}) &= [a_3(nT_{smp}) - ja_4(nT_{smp})]\, e^{j2\pi f_d nT_{smp}} \end{aligned} \tag{4.1}$$

where $\{a_i(nT_{smp})\}$ represents the discrete time signal of $\{a_i(t)\}$ (from Chapter 2), and T_{smp} is the transmitter output sample period. In general, $T_s = I \times T_{smp}$, where T_s denotes the symbol duration, and I is the integer interpolation factor.

The basic data pulse, with amplitude ± 1 (each with probability one half), width T_s, and centered at the origin, is expressed as

$$a(t) = rect\left(\frac{t}{T_s}\right) = \begin{cases} \pm 1 & |\,t\,| < T_s/2 \\ 0 & |\,t\,| > T_s/2 \end{cases} \tag{4.2}$$

Letting $A(f)$ denote the Fourier transform of $a(t)$, as illustrated in Figure 4.2, the first data shaping modulation process (denoted by 1st modulator in Figure 4.1), can be visualised in the frequency domain as illustrated in Figure 4.3 for the transmitted data sequence $\{a_1, a_2, a_3, a_4\} = \{+1, +1, +1, +1\}$. Note, that the sidelobes are omitted to facilitate easier and clearer graphical representation of $I(nT_{smp})$ given in (4.1). An equivalent process is carried out for $Q(nT_{smp})$.

After the first modulations, the components of the in-phase and quadrature-phase digital signals $I(nT_{smp})$ and $Q(nT_{smp})$ respectively (given in (4.1)), are digitally translated by the second (2nd) modulator onto a carrier frequency, f_c. This second modulation process is visualised in the frequency domain as illustrated in Figure 4.4, resulting in the final digital modulator output $s(nT_{smp})$, similar to (2.8) given in Chapter 2 with t replaced by nT_{smp}. Finally, the signal generated is band-pass filtered before transmission.

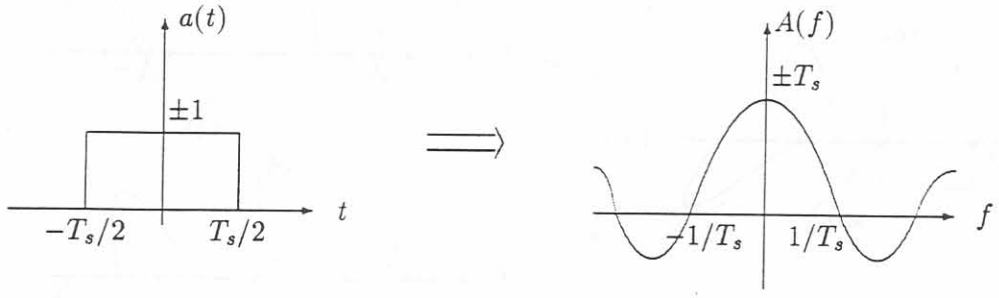

Figure 4.2: Fourier Transform of basic data pulse, $a(t)$; with amplitude ± 1, width T_s, and centered on the origin.

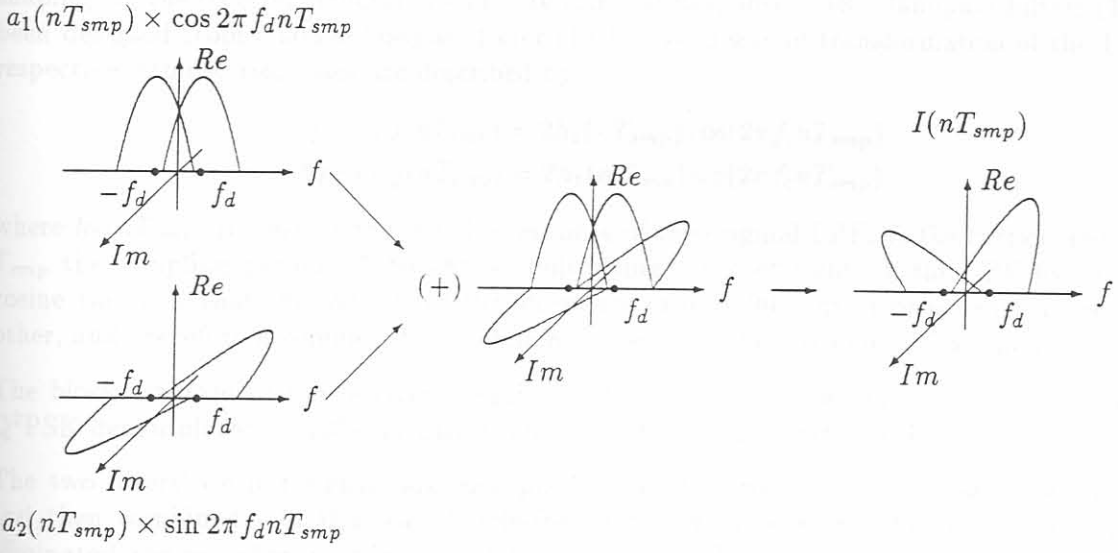

Figure 4.3: Visualisation of first quadrature modulation in the frequency domain, producing in-phase digital signal, $I(nT_{smp})$.

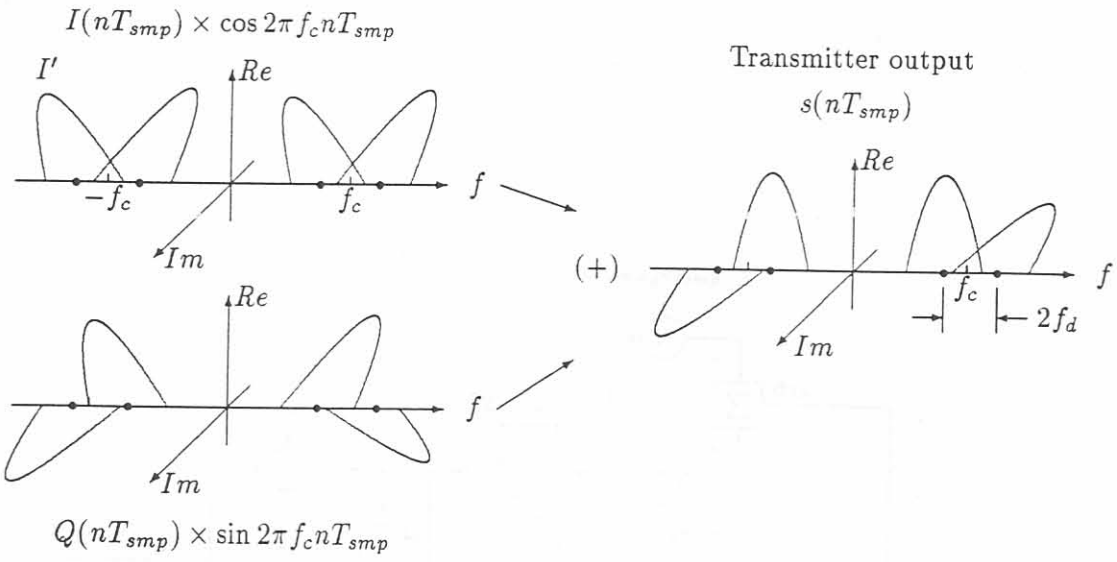

Figure 4.4: Visualisation of final modulation process in the frequency domain, producing Q²PSK transmitter output, $s(nT_{smp})$.

4.1.2 Digital Q²PSK Demodulator

The basic structural blocks of the Q²PSK modem receiver are the input bandpass filters, the carrier, f_c demodulator, the second bank of filters and the demodulation associated with the data pulse shaping. In the receiver structure two Finite Impulse Response (FIR) Bandpass Filters (BPF) have been designed from a single Lowpass Filter (LPF), by means of transformation of the LPFs. The respective impulse responses are described by:

$$\begin{aligned} h_{BPF1,I}(nT_{smp}) &= 2h_1(nT_{smp})\cos(2\pi f_c nT_{smp}) \\ h_{BPF1,Q}(nT_{smp}) &= 2h_1(nT_{smp})\sin(2\pi f_c nT_{smp}) \end{aligned} \tag{4.3}$$

where $h_1(nT_{smp})$ represents the impulse response of the original LPF, f_c the carrier frequency and T_{smp} the sampling period. A benefit of multiplying the coefficients of the LPF by the sine and cosine values is that the outputs of the filters are exactly 90° out of phase with respect to each other, and therefore a complex demodulation process may be implemented at the receiver.

The block diagram of the receiver is shown in Figure 4.5. (Note that the block diagram of the Q²PSK demodulator in [23] is incorrect, and should be as illustrated in Figure 4.5.)

The two filters' output signals are multiplied by both a cosine and sine signal of frequency f_c and then combined. In this way double-frequency terms generated by the multiplications are eliminated and no subsequent low-pass filtering is needed to recover the baseband signals [64]. The real and imaginary components of the Q²PSK signal can be obtained by means of a single complex demodulation. The real part of the Q²PSK is given by the following expression:

$$\begin{aligned} I'(nT_{smp}) &= [a_1(nT_{smp}) - ja_2(nT_{smp})]\, e^{j2\pi f_d nT_{smp}} \\ &= a_1(nT_{smp})\cos(2\pi f_d nT_{smp}) + a_2(nT_{smp})\sin(2\pi f_d nT_{smp}) \end{aligned} \tag{4.4}$$

Similarly, the imaginary part is given by:

$$Q'(nT_{smp}) = [a_3(nT_{smp}) - ja_4(nT_{smp})]\, e^{j2\pi f_d nT_{smp}} \tag{4.5}$$

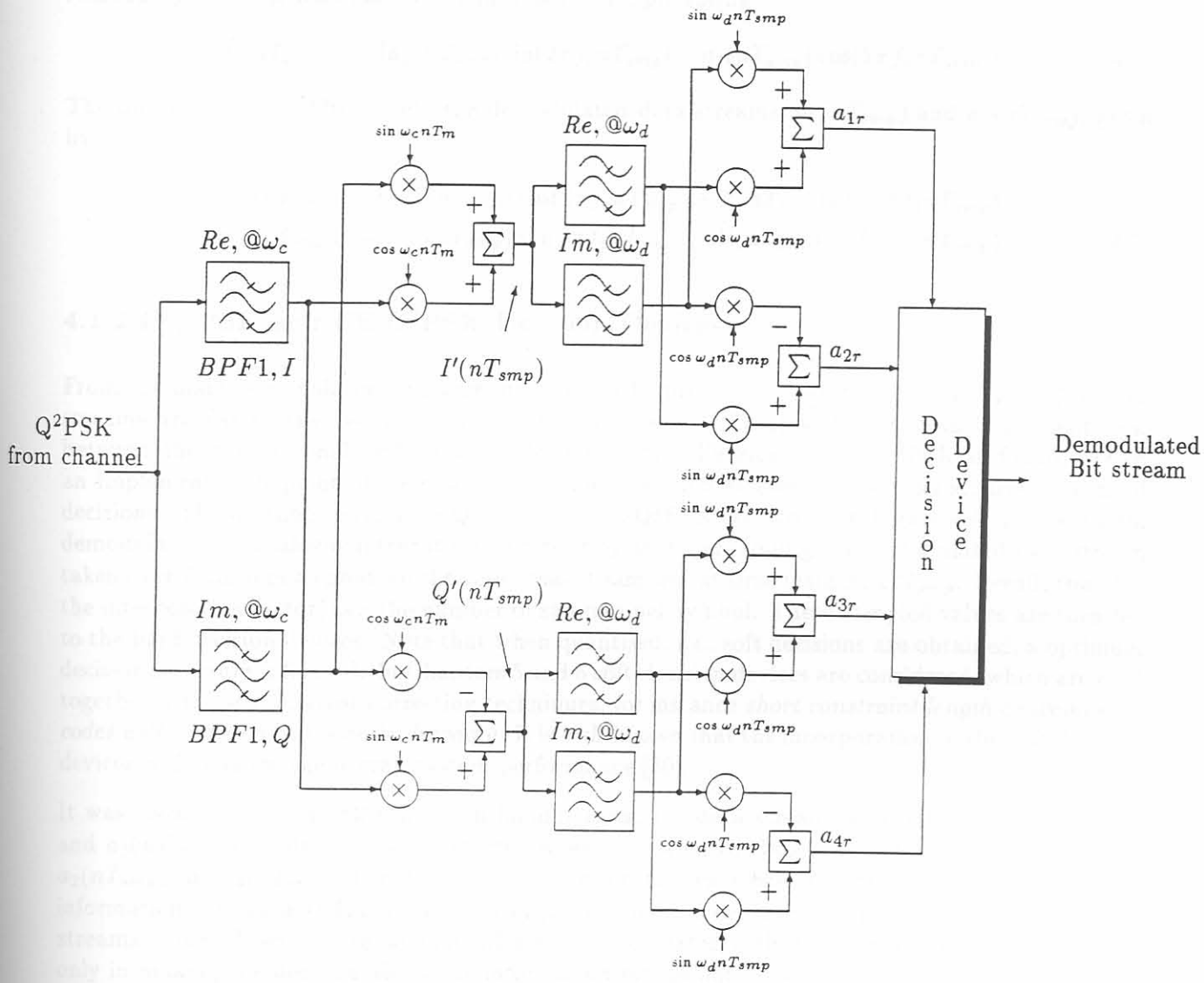

Figure 4.5: Q²PSK DSP based receiver block diagram

The method to recover the baseband signals $\{a_i(nT_{smp})\}$, from $I'(nT_{smp})$ and $Q'(nT_{smp})$ is to implement a new complex demodulator scheme for each quadrature signal component obtained from the carrier demodulation system. Only the demodulation of the $I'(nT_{smp})$ given by (4.4) signal will be shown. The demodulation of the information available in $Q'(nT_{smp})$ is obtained by identical means.

The demodulated signal $I'(nT_{smp})$ is applied to a pairs of Hilbert Transformers (HT), which are realised by two FIR bandpass filters in quadrature, producing

$$\hat{I}'(nT_{smp}) = j\left[a_1(nT_{smp})\sin(2\pi f_o nT_{smp}) - a_2(nT_{smp})\cos(2\pi f_o nT_{smp})\right] \tag{4.6}$$

The outputs of these filters yield the demodulated data streams, $a_1(nT_{smp})$ and $a_2(nT_{smp})$, given by

$$\begin{aligned} a_{1r}(nT_{smp}) &= I'(nT_{smp})\cos(2\pi f_d nT_{smp}) + \hat{I}'(nT_{smp})\sin(2\pi f_d nT_{smp}) \\ a_{2r}(nT_{smp}) &= I'(nT_{smp})\sin(2\pi f_d nT_{smp}) - \hat{I}'(nT_{smp})\cos(2\pi f_d nT_{smp}) \end{aligned} \tag{4.7}$$

4.1.2.1 Q^{2}PSK and CE-Q^{2}PSK Decision Devices

From the main demodulator structure, depicted in Figure 4.5, the estimates $\{a_{ir}(nT_{smp})\}_1^4$ of data streams are fed to the decision device. It is here where the only difference in implementation between the conventional Q^{2}PSK and coded Constant Envelope (CE) Q^{2}PSK is found. From an implementation point of view, a non-optimum decision strategy is considered, based on hard decisions. The decision devices for Q^{2}PSK and CE-Q^{2}PSK are shown in Figure 4.6. In the digital demodulator the analogue integrator is replaced by an adder, adding the demodulated data stream taken over I samples to produce the value that is sampled at time instant, kIT_{smp}. Recall, that I is the interpolation factor, i.e., the number of samples per symbol. These sampled values are then fed to the hard decision devices. Note that when quantised, i.e., soft decisions are obtained, a optimum decision structure is formed. In Chapters 5 and 6 soft-decision devices are considered, which are used together with forward error-correcting techniques, for instance *short constraint length convolutional codes with soft-decision Viterbi decoding*. It is well-known that the incorporation of the soft-decision devices will enhance the overall system performance [30].

It was shown in [12, 11, 23] that upon bandlimiting, the data streams associated with $a_1(nT_{smp})$ and $a_3(nT_{smp})$ are relatively less distorted when compared to those associated with data streams $a_2(nT_{smp})$ and $a_4(nT_{smp})$. For this reason, in the constant envelope decision device, decisions about information bits in $a_1(nT_{smp})$ and $a_3(nT_{smp})$ are made independently from the respective data streams alone. Then, the redundant information associated with data stream $a_4(nT_{smp})$, is used only in making the decision about the information bits in data stream $a_2(nT_{smp})$. In other words, in order to make a decision about $a_2(nT_{smp})$, a simplifying assumption is made that $a_1(nT_{smp})$ and $a_3(nT_{smp})$ are decoded correctly. In his PHD thesis [12], Saha has shown that this decision is the random variable V, given by

$$V = a_{2r}(nT_{smp}) - \frac{a_1(nT_{smp})}{a_3(nT_{smp})}a_{4r}(nT_{smp}) \tag{4.8}$$

A decoder, which is optimum in the sense of minimising the probability of error, will make a ± 1 decision, depending on the sign of V ($V \geq 0$ or $V < 0$). However, information of the correct decision statistic, V is subject to the correctness of the decisions about $a_1(nT_{smp})$ and $a_3(nT_{smp})$.

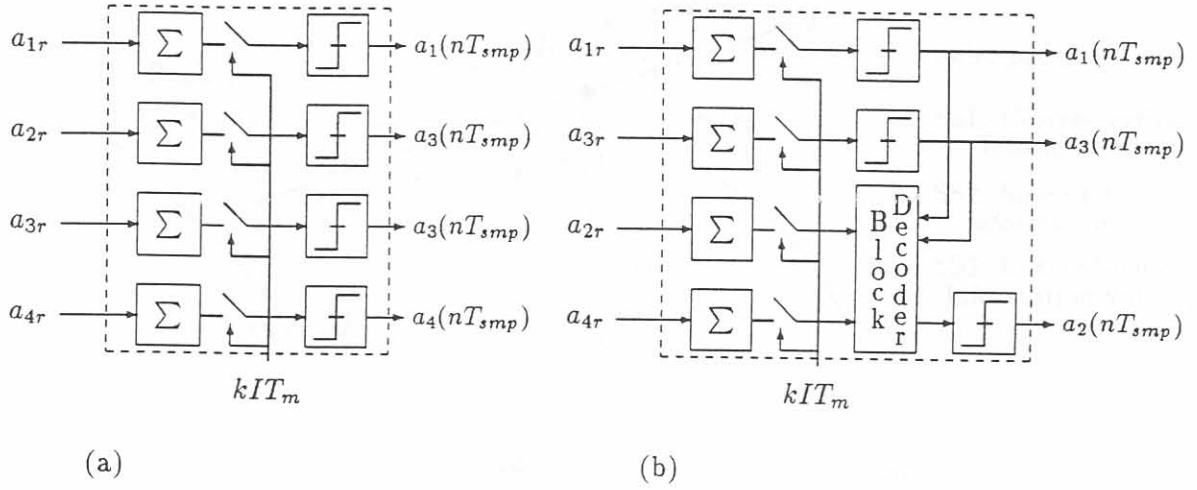

Figure 4.6: Non-optimum demodulator decision devices for (a) Q²PSK, and (b) CE-Q²PSK.

4.1.2.2 Decoupled Q²PSK signal space

The constellation of a Q²PSK signal consists of the vertices of a hypercube of dimension $N = 4$ centered around the origin. The difficulty of representing a four-dimensional space leads one to decouple it into three two-dimensional (2D) subspaces, the first associated with the cosine-carrier, the second with the sine-carrier, and the third with the combination of the sine- and cosine-carrier information associated with data shaping pulse, $q_1(t)$, as illustrated in Figure 4.7.

Recall, that two of the hypercube dimensions come from the orthogonality of the data shaping pulses $q_1(t)$ and $q_2(t)$, and the other from the orthogonal carriers. One may therefore consider the vertices of the hypercube as the Cartesian product of two sets of biorthogonal, or 4-PSK signal points $\{a_i\}$ and $\{b_i\}$ [23].

The geometry of the hyper cube is, however, lost in extremely bandlimited situations, because the pulses, $q_1(t)$ and $q_2(t)$ chosen do not remain perfectly orthogonal under these circumstances. Alternative pulse shapes which are not affected by Intersymbol Interferences (ISI) and remain perfectly orthogonal in bandlimited situations are shown by Saha [11, 12]. However, these pulses suffer from other problems such as physical realisability. If, on the other hand, the data shaping pulses used are not minimum-bandwidth (i.e., Nyquist) signals, the modulation scheme suffers from ISI, resulting in degradation of the system performance.

4.1.3 Modem waveforms and spectra

The generation of the Q²PSK signal is illustrated in Figures 4.8, for the case where $h = 1.0$. It is assumed that the binary random data sequence $\{a_n\}, n = 1, 2, \ldots$, represented in Figure 4.8(a) as a rectangular unit pulse stream $a_n(t)$, is divided into four subsequences $\{a_i\}_1^4$. The subsequences illustrated in Figures 4.8(b) and 4.8(d) are fed to the first (upper) quadrature modulator of Figure 4.1, while subsequences illustrated in Figures 4.8(c) and 4.8(e) are fed to the bottom quadrature modulator. The final output, illustrated in Figure 4.8(h), are formed by the summation of the outputs obtained from the two quadrature modulators.

Similarly, the generation of the CE-Q²PSK is illustrated in Figure 4.9, with the only difference

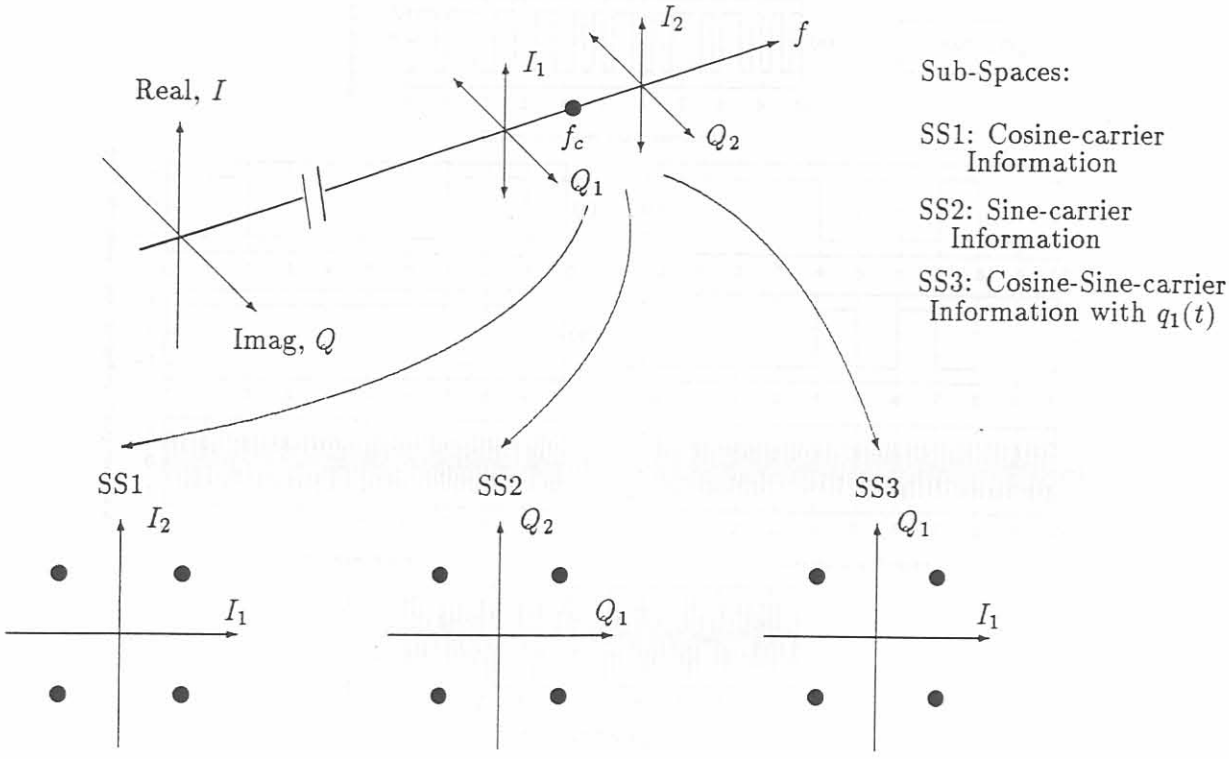

Figure 4.7: Decoupled Q²PSK signal space.

being that the original information data sequence is coded by the rate-3/4 block coder given by (2.15) to facilitate a constant envelope signal.

Figure 4.10 shows the simulated power spectral densities of unfiltered Q²PSK (CE-Q²PSK) and CP-Q²PSK, with the Intermediate Frequency (IF) chosen as 25.0 *kHz*. Note the faster spectral fall-off of the constant phase CP-Q²PSK system, with the latter being achieved at the expense of an increase of 25% in bandwidth requirement.

In the analysis so far, the assumption was made that the reference signal used for demodulation is perfectly frequency and phase synchronised to the transmitted signal. In a practical receiver, imperfect carrier frequency and phase results in performance degradation [65, 66]. Since the carrier recovery system forms its demodulation reference from a *noise-* and *interference-perturbed* version of the transmitted signal, the phase and frequency errors are random processes. This is even more prominent in SFH systems, where the large FH bandwidths may cause the FH synthesisers to loose frequency and phase coherence over successive hops. The degradation is a function of the rate of variation of the phase and frequency error over the symbol interval, and for a SFH application, over the *hop interval*, T_{HOP}.

4.2 BLOCK TRANSMISSION STRATEGY

As stated in paragraph 3.1.1.3, in Slow Frequency Hopped (SFH) or Time-Division Multiple Access (TDMA) applications a bursty signalling strategy is adopted, requiring that data be transmitted in short bursts or blocks, corresponding for instance with the time dwelt at each hopping frequency or time slot. Furthermore, proper synchronisation is needed to allow for any amount of timing

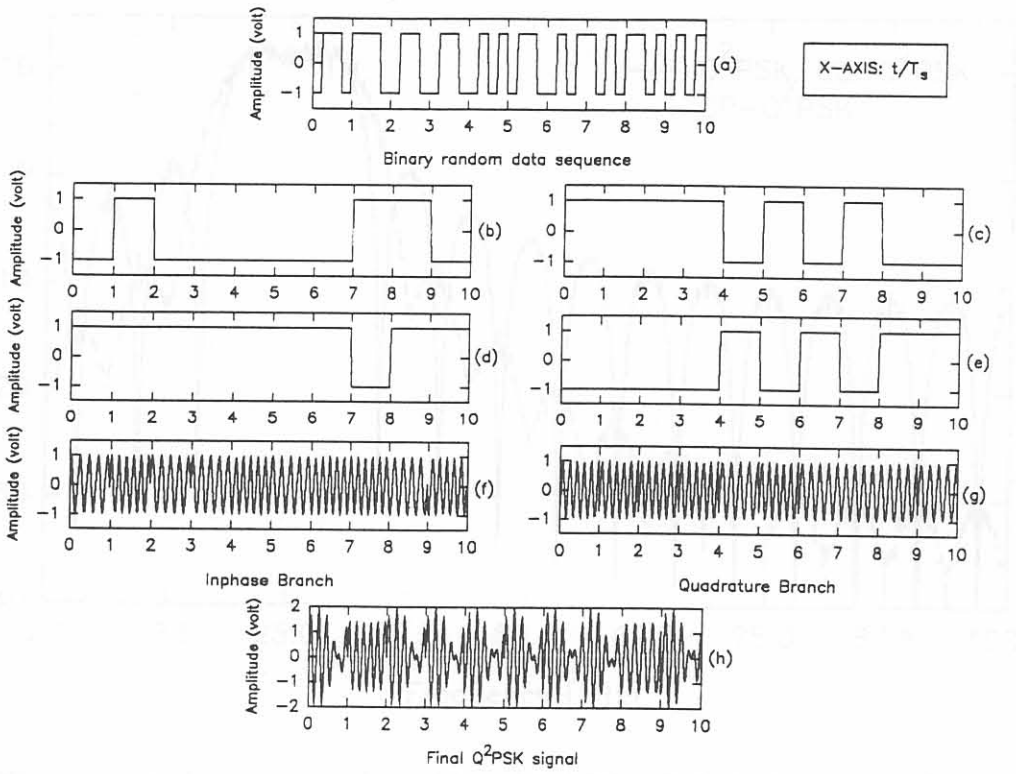

Figure 4.8: An illustrative timing diagram showing the waveforms involved in generating the Q^2PSK signal.

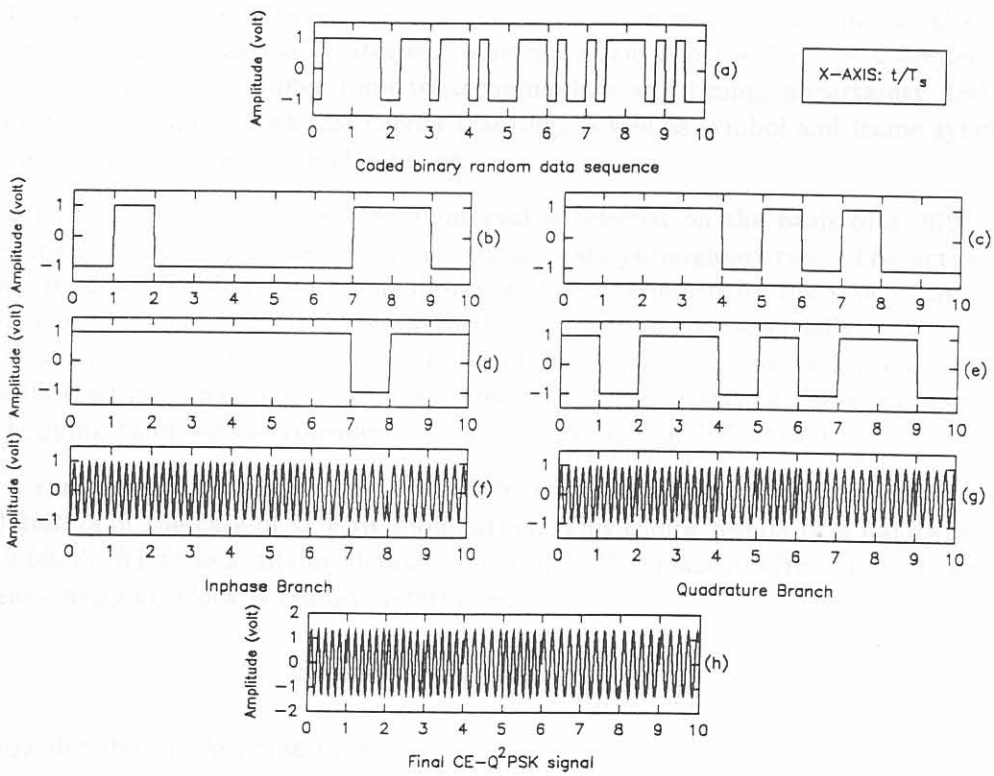

Figure 4.9: An illustrative timing diagram showing the waveforms involved in generating the CE-Q^2PSK signal.

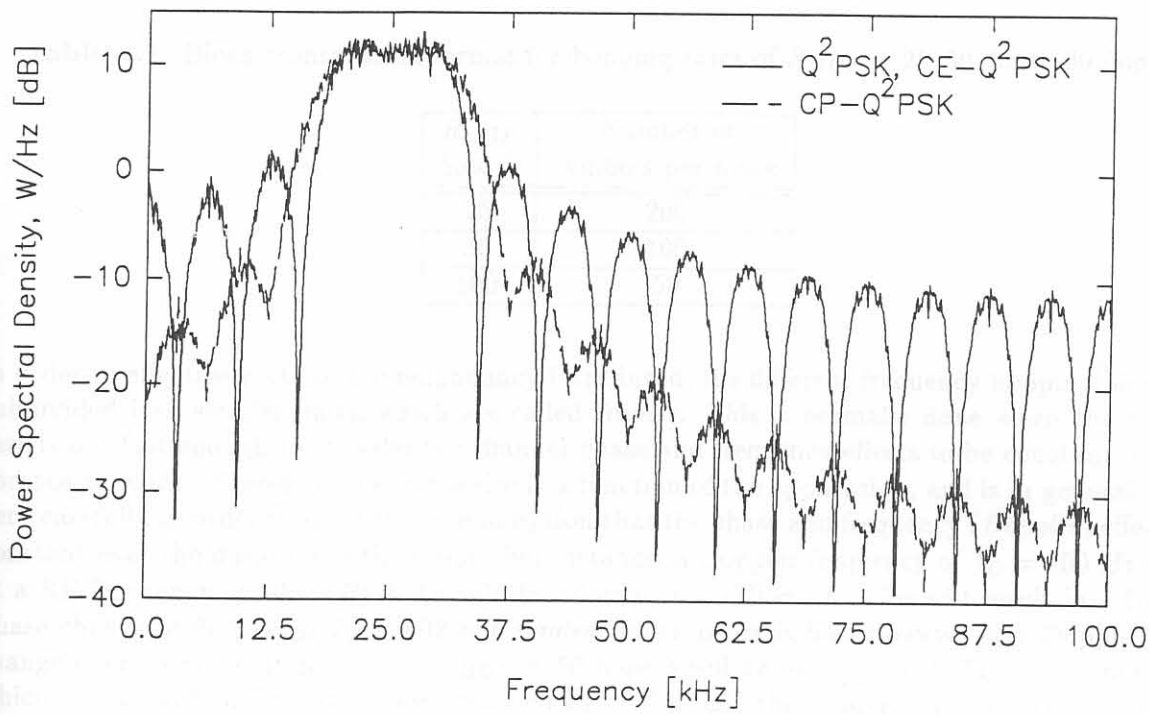

Figure 4.10: Power spectral densities of Q²PSK, CE-Q²PSK and CP-Q²PSK.

uncertainty that may arise between the transmitter and the receiver in a practical situation. To incorporate this synchronisation strategies, some redundancy in the form of a *header* is required. The header should provide buffer time to accommodate any timing uncertainty and to include information required to establish fast carrier tracking, as well as symbol and frame synchronisation between the modem transmitter and receiver.

For the current system design the header interval is selected on the basis of a 20% redundancy factor, resulting in a 20% decrease in the maximum data throughput rate. The actual amount of redundancy required is a function of the design and is determined by the type of application and operational environment. For TDMA systems this factor can be substantially lower due to the fact that all users share a common system synchronisation slot. In order to facilitate a FH system, the block signalling strategy has to be specifically designed to incorporate different hopping rates. The following hopping rates will be considered: $R_{HOP} = 25, 50,$ and 100 *hops/s*.

Consider a symbol rate of $R_s = 5.0$ *ksymbols/s*, which corresponds to a maximum bit rate of $R_b = 20.0$ *kbits/s* in the case of Q²PSK modulation. This choice results in a bandwidth efficiency of $\eta_f = 2.0$ *bits/s/Hz* if the available channel bandwidth, W equals 10 *kHz*. The number of symbols per frequency hopping block is readily determined by

$$\text{Symbols per Block} = \frac{1}{R_{HOP} \cdot T_s} \tag{4.9}$$

where R_{HOP} denotes the hopping rate.

For the chosen symbol rate, R_s, the symbol duration, T_s is 200×10^{-6} *s*. The number of symbols per block can now be calculated for the different hopping rates, R_{HOP}. These are summarised in the table below.

Table 4.1: Block transmission format for hopping rates of $R_{HOP} = 25, 50$, and 100 *hops/s*.

R_{HOP} *hops/s*	Number of symbols per block
25	200
50	100
100	50

In order to effectively utilise the redundancy introduced, the different frequency hopping blocks are subdivided into smaller units, which are called *frames*. This is normally done when the hopping rate is not fast enough to consider the channel phase and frequency effects to be constant over one composite block. Therefore, the frame size is a function of the application, and is in general chosen very carefully in order to support the assumption that the phase and frequency (*Doppler*) effects are constant over the duration of the frame. For instance, a Doppler frequency of $f_D = 100$ *Hz* results at a RF frequency of 900 *MHz* if the vehicle velocity, $\nu = 120$ *km/h*. This will result in a Doppler phase change of $\theta_{D,T_s} = f_D T_s = 0.02$ *rad/symbol*, which is negligible. However, the Doppler phase change over an block at $R_{HOP} = 1/T_{HOP} = 50$ *hops/s* will be $\theta_{D,T_{HOP}} = f_D T_{HOP} = 2$ *rad/block*, which is substantial. This is however reduced by 80% if only the header interval is considered, i.e, 0.4 *rad/header interval*. T_{HOP} denotes the block duration. In order to provide information about the channel to establish fast and effective synchronisation, the redundant information associated with each block, is included in each frame. The basic frame structure is shown in Figure 4.11, where T_H and T_D denote the header and data intervals respectively.

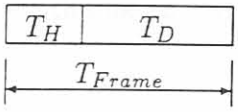

Figure 4.11: Basic frame structure.

In order to maintain the throughput rate, and to support practical header lengths of 8 and 16 symbols, it is assumed for the current design that a maximum allowable frame duration of $T_{Frame} =$ 10 *ms* and 20 *ms*, for the length 8 and 16 header sequences respectively. The block signalling strategy is depicted in Figure 4.12, shown for the different hopping rates, $R_{HOP} = 25, 50$, and 100 *hops/s*, utilising an 8 symbol header. Note, that an amount of "dead time" is included in the form of guard symbols, inserted at the beginning and end of the SFH signalling block. These are inserted to ensure that data will not be lost at the begin/end of the frame when a longer/shorter transmission path suddenly appears. In the situation where more than 8 symbols of header information is lost at the beginning of a block, due to multipath phenomena which causes loss of synchronisation, the loss of a total hopping frame will be inevitable.

The designs for the SFH application were extended to support two different symbols rates, $R_s =$ 5.0, 6.0 *ksymbols/s*, for the hopping rates of 25, 50, and 100 *hops/s*. These designs are summarised in Table 4.2.

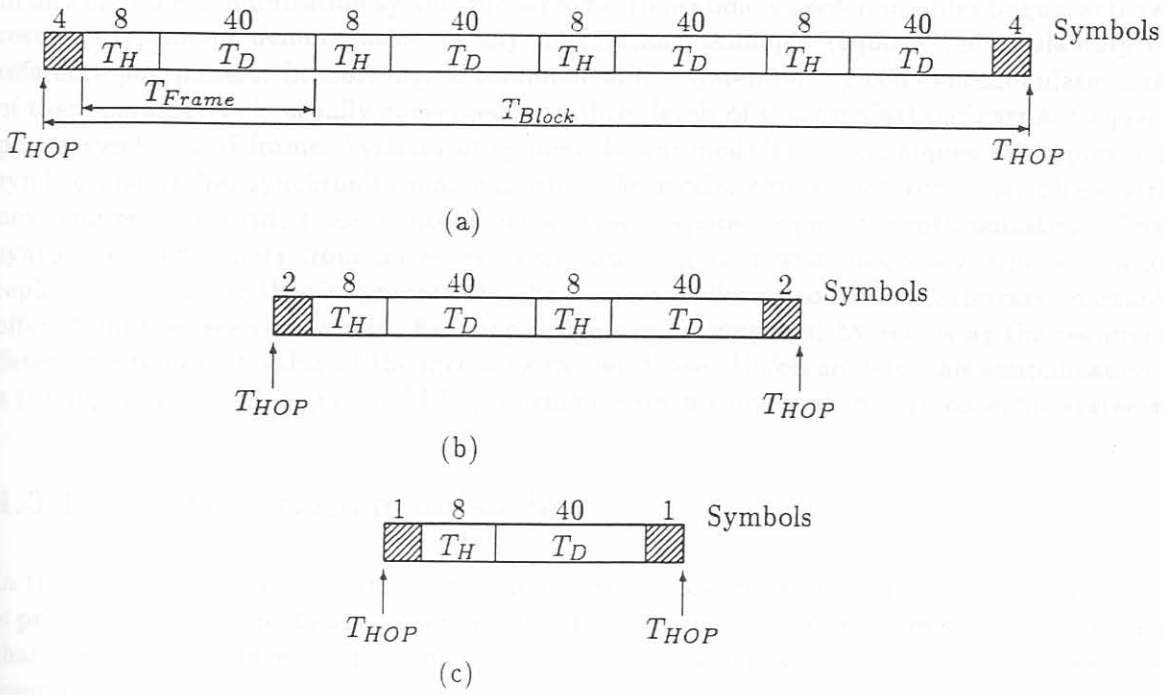

Figure 4.12: Block signalling format for symbol rate of $R_s = 5.0$ *ksymbols/s* for the different hopping rates, (a) 25 (b) 50 (c) 100 *hops/s*, utilising an 8 symbol header.

Table 4.2: SFH design summary for hopping rates of $R_{HOP} = 25, 50$, and 100 *hops/s*.

Hopping Rate	Symbol Rate, R_s	Effective Bit Rate	Header Length, L	Symbols per Block	Symbols per Frame	Frame(s) per Block	Guard Symbols
25 *hops/s*	5.0 *kSymb/s*	16.0 *kbits/s*	8	200	40 + 8	4	8
			16	200	80 + 16	2	8
	6.0 *kSymb/s*	19.2 *kbits/s*	8	240	48 + 8	4	16
			16	240	96 + 16	2	16
50 *hops/s*	5.0 *kSymb/s*	16.0 *kbits/s*	8	100	40 + 8	2	4
			16	100	80 + 16	1	4
	6.0 *kSymb/s*	19.2 *kbits/s*	8	120	48 + 8	2	8
			16	120	96 + 16	1	8
100 *hops/s*	5.0 *kSymb/s*	16.0 *kbits/s*	8	50	40 + 8	1	2
			16	50	—	—	—
	6.0 *kSymb/s*	19.2 *kbits/s*	8	60	48 + 8	1	4
			16	60	—	—	—

4.3 Q²PSK SYNCHRONISATION

In any digital communication system proper synchronisation is needed in order to guarantee efficient results. Optimum demodulation of any modulation technique requires the availability of some reference parameters. In most digital communications systems using coherent modulation, recovery of these parameters is usually concerned with three levels of synchronisation: carrier frequency and phase, symbol, and frame. Systems using noncoherent modulation techniques will typically require symbol and frame synchronisation, but since the modulation is not coherent, phase locking is not required. Instead, these noncoherent systems require frequency synchronisation. Frequency synchronisation differs from phase synchronisation in that with frequency synchronisation, the replica of the carrier that is generated by the receiver is allowed to have an arbitrary constant phase offset from the received carrier. Receiver design can be simplified by removing the requirement to determine the exact value of the incoming carrier phase. Unfortunately, this simplification causes a penalty in terms of degraded BER performance when compared to fully coherent systems.

4.3.1 Frame Synchronisation

In this section a novel multidimensional double complex correlation frame synchronisation strategy is proposed for Q²PSK. In asynchronous (burst) communication applications, e.g. FH, it is required that the data be transmitted in bursts or frames corresponding to the time spent at each hopping frequency.

The frame synchronisation procedure is closely related to the radio FH process. It is assumed that appropriate control signals are provided by the radio receiver to signify the start and end of the hopping sequence at each hopping frequency. Although these signals could be used to control the modem data block transmission process at every hopping frequency, it was decided to derive an independent Frame Synchronisation control signal from a special header preceding the data block in the block transmission scheme proposed in section 4.2.

In traditional quadrature modems, for instance QPSK, a single complex correlation is performed, requiring a relatively long correlation sequence to keep the probability of false detection low. In 4D Q²PSK, by exploiting the multidimensional signal space effectively, shorter correlation sequences can be used by performing two complex correlations to achieve the same probability of false detection. The delay introduced by this synchronisation strategy will be less than that resulting from a single long correlation sequence. Furthermore, since the complexity of the synchronisation procedure is proportional to the length of the codeword sequences, the complexity of the double complex correlation process will be comparable to a single long correlation process.

In an attempt to utilise the available dimensions within the Q²PSK signal space effectively, a novel frame synchronisation procedure is proposed using two synchronisation sequences or codewords, which are included as the header of each message frame [60]. The synchronisation sequences are selected from a family of complex correlation sequences of equal length L, exhibiting perfect periodic and near-perfect aperiodic correlation characteristics [67, 68]. From equation (2.8) binary data sequences a_1 and a_4 are used to form the real and imaginary parts of the first correlation sequence $X_{14} = a_1 + ja_4$, while data sequences a_2 and a_3 are used to form the real and imaginary parts of the second correlation sequence $X_{23} = a_2 + ja_3$. Each complex sequence occupies one distinct pair of two available pairs of dimensions within the Q²PSK signal space. By using all available dimensions of the Q²PSK signal space a form of synchronisation diversity is introduced

whereby a significant frame synchronisation improvement can be gained relative to existing (2D) QPSK techniques.

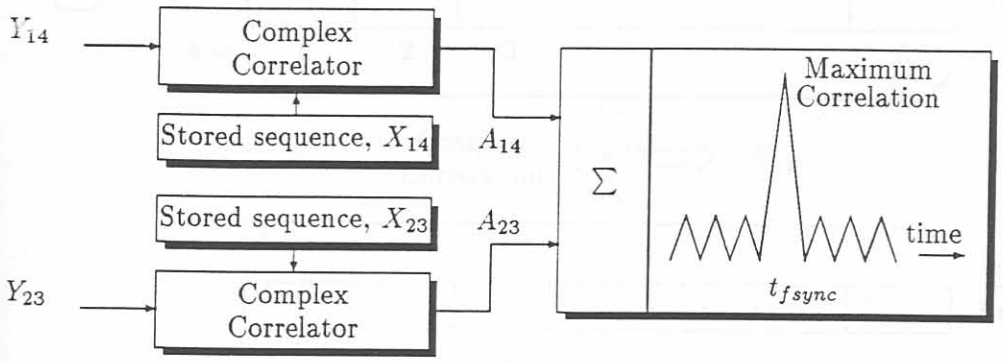

Figure 4.13: Proposed Q²PSK frame synchronisation procedure.

The proposed Q²PSK frame synchronisation procedure is illustrated in Figure 4.13. Two complex correlations are performed between the received (possibly distorted) S times oversampled synchronisation sequences, Y_{pq} and one sample per symbol local replicas of the reference sequences, X_{pq}, respectively, where $p\epsilon\{1,2\}$ and $q\epsilon\{3,4\}$. The complex correlation processes yield S complex correlation outputs $\Re_{pq}(n,i), i = 1,2,\ldots,S$, where i denotes the sampling index and $n = 1,\ldots,N$ is the range in symbol periods over which the optimum correlation peak is sought, starting from symbol interval $n = 1$ to typically $N = L$.

The complex correlation procedure illustrated in Figure 4.14 produces the following output sequences:

$$\Re_{pq}(n,i) = E[X^*_{pq}Y_{pq}] = \Re^R_{pq}(n,i) + j\,\Re^I_{pq}(n,i) \tag{4.10}$$

with $p \epsilon \{1,2\}$, $q \epsilon \{3,4\}$ and $j = \sqrt{(-1)}$, where

$$\begin{aligned}
\Re^R_{pq}(n,i) &= \sum_{k=1}^{L} \left[X^R_{pq}(k)Y^R_{pq}(k+n,i) + X^I_{pq}(k)Y^I_{pq}(k+n,i)\right] \\
\Re^I_{pq}(n,i) &= \sum_{k=1}^{L} \left[X^R_{pq}(k)Y^I_{pq}(k+n,i) - X^I_{pq}(k)Y^R_{pq}(k+n,i)\right] \\
&\quad p \epsilon \{1,2\},\ q \epsilon \{3,4\}; \\
&\quad i = 1,\ldots,S, \text{ and } n = 1,\ldots,N
\end{aligned} \tag{4.11}$$

where R and I denote the real and imaginary parts of the complex sequences, respectively.

Note that the reference sequences' samples $X_{pq}(k)$ are spaced by the symbol period T_s, whereas the received sequences $Y_{pq}(k)$ are S times oversampled, as illustrated in Figure 4.14.

The magnitudes of the correlator outputs, denoted by $A_{pq}(n,i)$ are given by:

$$\begin{aligned}
A_{pq}(n,i) &= \sqrt{[\Re^R_{pq}(n,i)]^2 + [\Re^I_{pq}(n,i)]^2} \\
\text{where} &\quad i = 1,\ldots,S \text{ and } n = 1,\ldots,N
\end{aligned} \tag{4.12}$$

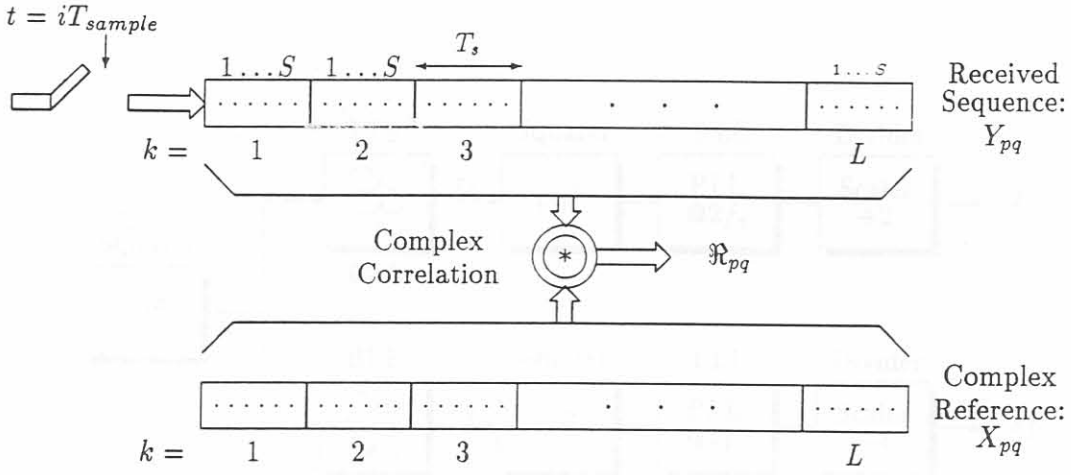

Figure 4.14: Complex correlator block diagram.

The resulting magnitudes are summed to produce the correlation magnitude, $A(n,i) = A_{14}(n,i) + A_{23}(n,i)$. By comparing these magnitudes to a fixed threshold, the optimum symbol and frame synchronisation instant t_{fsync} is obtained as that sampling instant $i_m = i$ within the symbol period corresponding with $n = n_{opt} \epsilon\{1,2,\ldots,N\}$ producing the maximum correlation peak. In this way frame and symbol synchronisation are obtained to within $1/S$ of a symbol period T_s. The optimum symbol samples for the rest of the FH frame are obtained by incrementing the sampling index i by S, starting from $i_m = t_{fsync}$, expressed as $t_o = t_{fsync} + (i_m - 1)T_s$.

In addition, the carrier phase rotation, θ_o, can be obtained from

$$\theta_o = \theta_o(i_m) = \arctan\left[\frac{\Re^R_{pq}(n,i)}{\Re^I_{pq}(n,i)}\right] \tag{4.13}$$

It is assumed that the initial phase change is set to zero, so that any other value of θ_o corresponds to the carrier phase offset induced by the channel.

4.3.2 Carrier and Symbol Synchronisation

Since Q²PSK, and its constant envelope counterpart utilise all the available signal space dimensions, non-coherent operation is not feasible. For coherent or matched filter detection one needs to have carrier phase and clock timing information. As in an MSK receiver, the Q²PSK timing information is also derived from the received signal itself. The self-synchronising property of Q²PSK is most easily explained by reference to Sunde's FSK. The timing information signals are derived from the Q²PSK modulated signal by a nonlinear operation, such as a frequency doubler (squaring), and appropriate filtering.

In 1989 Saha proposed a carrier synchronisation procedure for Q²PSK signals, based on traditional carrier recovery methods utilising nonlinear squaring devices, filters and Phase Locked Loops (PLL). A variation of Saha's synchronisation scheme was analysed by De Gaundenzi [55], employing a fourth power generator device instead of a square device. A detailed diagram of the Saha synchronisation circuit for Q²PSK and CE-Q²PSK are shown in Figures 4.15 and 4.16.

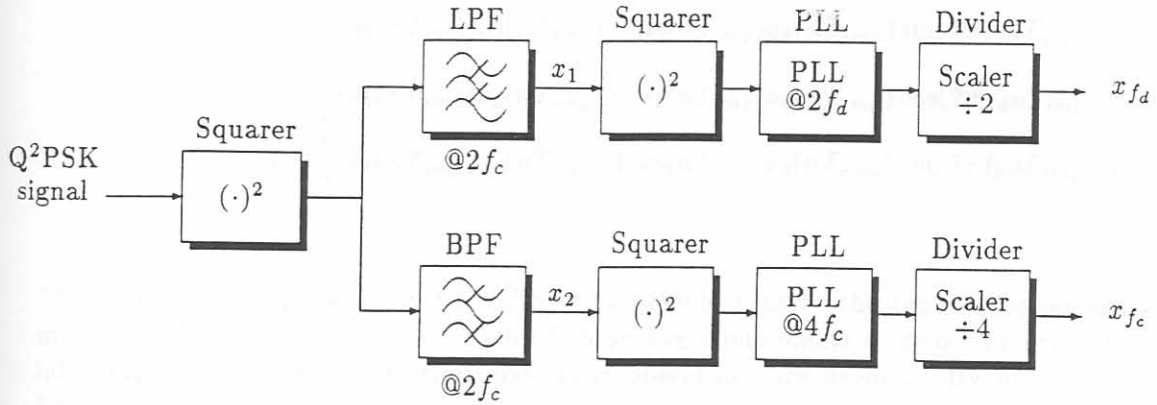

Figure 4.15: Saha synchronisation circuit for Q²PSK.

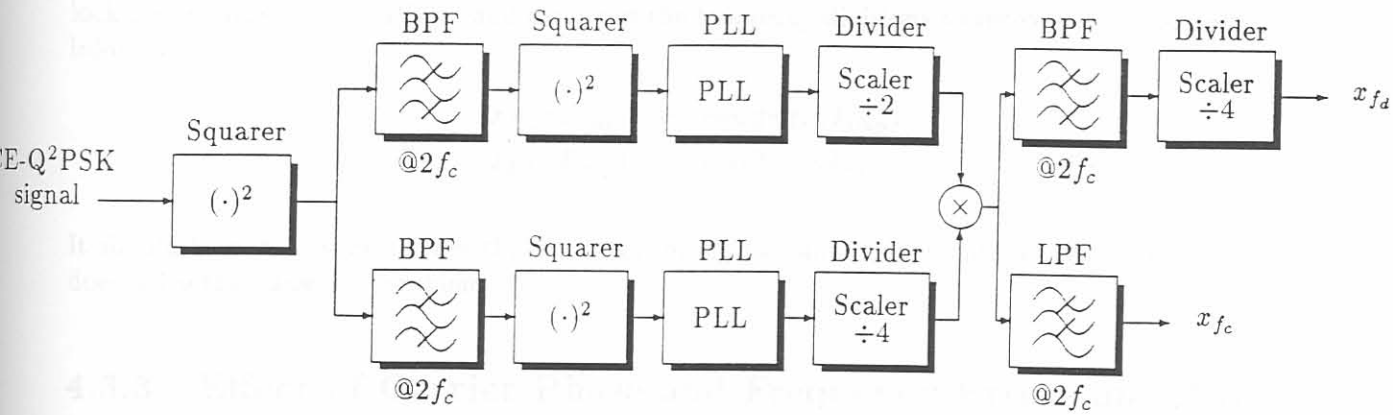

Figure 4.16: Saha synchronisation circuit for CE-Q²PSK.

If the sampled Q²PSK signal is passed through a squaring device, one obtains the following signal

$$\begin{aligned} s^2_{Q^2PSK}(nT_{smp}) &= 1 + \frac{1}{2}(a_1(nT_{smp})a_2(nT_{smp}) + a_3(nT_{smp})a_4(nT_{smp}) \sin(2\pi f_d nT_{smp}) \\ &+ \frac{1}{2}(a_1(nT_{smp})a_2(nT_{smp}) - a_3(nT_{smp})a_4(nT_{smp}) \sin(2\pi f_d nT_{smp}) \cos(4\pi f_c nT_{smp}) \\ &+ \frac{1}{2}(a_1(nT_{smp})a_3(nT_{smp}) + a_2(nT_{smp})a_4(nT_{smp}) \sin(4\pi f_c nT_{smp}) \\ &+ \frac{1}{2}(a_1(nT_{smp})a_3(nT_{smp}) - a_2(nT_{smp})a_4(nT_{smp}) \cos(2\pi f_d nT_{smp}) \sin(4\pi f_c nT_{smp}) \\ &+ \frac{1}{2}(a_1(nT_{smp})a_4(nT_{smp}) + a_2(nT_{smp})a_3(nT_{smp}) \sin(2\pi f_d nT_{smp}) \sin(4\pi f_c nT_{smp}) \end{aligned} \tag{4.14}$$

There are five components on the RHS of (4.14) which carry the required synchronisation information. The expected value of each of these five components is zero. In order to recover the information, filtering and further non-linear operations are needed. By lowpass and bandpass filtering of the squared signal, one may construct the two signals $x_1(t)$ and $x_2(t)$, given by:

$$x_1(nT_{smp}) = \frac{1}{2}(a_1(nT_{smp})a_2(nT_{smp}) + a_3(nT_{smp})a_4(nT_{smp}) \sin(4\pi f_d nT_{smp}) \tag{4.15}$$

$$x_2(nT_{smp}) = \frac{1}{2}(a_1(nT_{smp})a_3(nT_{smp}) + a_2(nT_{smp})a_4(nT_{smp}) \sin(4\pi f_c nT_{smp}) \tag{4.16}$$

After squaring and taking the expectation (average), one obtains

$$E\{x_1^2(nT_{smp})\} = \frac{1}{4}(1 - \cos 4\pi f_d nT_{smp}) \tag{4.17}$$

$$E\{x_2^2(nT_{smp})\} = \frac{1}{4}(1 - \cos 8\pi f_c nT_{smp}) \tag{4.18}$$

Thus, on average, $x_1^2(t)$ and $x_2^2(t)$ contain spectral lines at $2f_d$ and $4f_c$. One can use these lines to lock phase-locked loops (PLLs) and carry out the frequency divisions to recover the synchronisation information as

$$x_{f_d}(nT_{smp}) = \cos(2\pi f_d nT_{smp}) \tag{4.19}$$

$$x_{f_c}(nT_{smp}) = \cos(2\pi f_c nT_{smp}) \tag{4.20}$$

It should be noted, that neither the circuit proposed by Saha, nor the proposed by De Gaundenzi, does solve the hang-up[1] problem.

4.3.3 Effect of Carrier Phase and Frequency Errors on BER

Lets consider the effect of an imperfect carrier recovery, due to Doppler and channel effects, on the BEP performance of the Q²PSK demodulator. The effect of bandwidth-limiting will, however, not be taken into account. In other words it is assumed that the IF receiver bandpass filter has a wide enough bandwidth so that the received signal is not appreciably distorted.

[1] Hang-up is defined as the prolonged dwelling at large phase errors.

In order to evaluate the effect of static phase error, θ_e and static frequency error, f_e, the channel input signal, $s_{Q^2PSK}(t)$ to the demodulator is mixed with the "coherent" references at the IF carrier frequency, f_c, resulting in the synchronisation information given in (4.19) and (4.20) to be modified to

$$\begin{aligned} x_{f_d}(nT_{smp}) &= \cos\left(2\pi(f_c + f_e)nT_{smp} + \theta_e\right) \\ x_{f_c}(nT_{smp}) &= \cos\left(2\pi(f_c + f_e)nT_{smp} + \theta_e\right) \end{aligned} \tag{4.21}$$

The effect of the static carrier phase error, θ_e, is considered, by eliminating the frequency error component, f_e from (4.21). Assuming the non-optimum integrate-and-dump decision device for Q²PSK depicted in Figure 4.6(a), the expression for average BEP conditioned on a fixed value of phase error θ_e, is found to be [55]:

$$\begin{aligned} P_e(\theta_e) &= \frac{1}{2}\left\{ Q\left[(\cos\theta_e + \sin\theta_e)\sqrt{2\frac{E_b}{N_o}}\right] \right. \\ &\quad \left. + Q\left[(\cos\theta_e - \sin\theta_e)\sqrt{2\frac{E_b}{N_o}}\right]\right\} \end{aligned} \tag{4.22}$$

where E_b denotes the average transmitted energy per bit. Assuming a Tikhonov-distributed phase error [56, 30] the value of the average BEP of the demodulator for a given phase error variance $\sigma^2_{\theta_e}$ is given as

$$P_e = \frac{2}{\pi I_o\left(\frac{1}{16\sigma^2_{\theta_e}}\right)} \int_{-\pi/4}^{\pi/4} \exp\left(-\frac{\cos 4\theta_e}{16\sigma^2_{\theta_e}}\right) P_e(\theta_e)d\theta_e \tag{4.23}$$

For a discrete carrier synchronisation loop, the Tikhonov PDF is given by

$$\rho(\phi) = \begin{cases} \frac{\exp(\rho\cos M\phi}{(2\pi/M)I_o(\rho)} & |\,\phi\,| \leq \pi M \\ 0 & \text{otherwise} \end{cases} \tag{4.24}$$

where ρ is the SNR in the system bandwidth. M is the number of phases for a M-phase Costas loop, and $M = 1$ for PLL-type synchronisers.

Figure 4.17 illustrates the BEP curves of the receiver in the presence of the Tikhonov-distributed phase errors. Note that $\sigma = \sigma_{\theta_e}$, is the phase error variance. From these curves the sensitivity of the Q²PSK signal with respect to imperfect carrier references can be deduced. A possible explanation of this behavior is that the effect of a phase misalignment on the Q²PSK signal is two-fold. On one side the timing error induces adjacent-symbol interference, and on the other hand, the error on the initial phase of the demodulation subcarriers produces a *loss in orthogonality* between the data symbols. All these effects produces excess crosstalk between the parallel data streams, resulting in a degradation in BER noticeable in the graphs of Figure 4.17.

4.3.4 Kalman Phase and Frequency Estimation

The first step after frame and symbol synchronisation has been obtained, is to establish the magnitude of any phase and frequency offset that may be present in the received Q²PSK modem signal.

A feed-forward, DSP based frequency estimation strategy for channels with both frequency and phase uncertainty, will be presented, based on the techniques presented in [69, 70]. The proposed

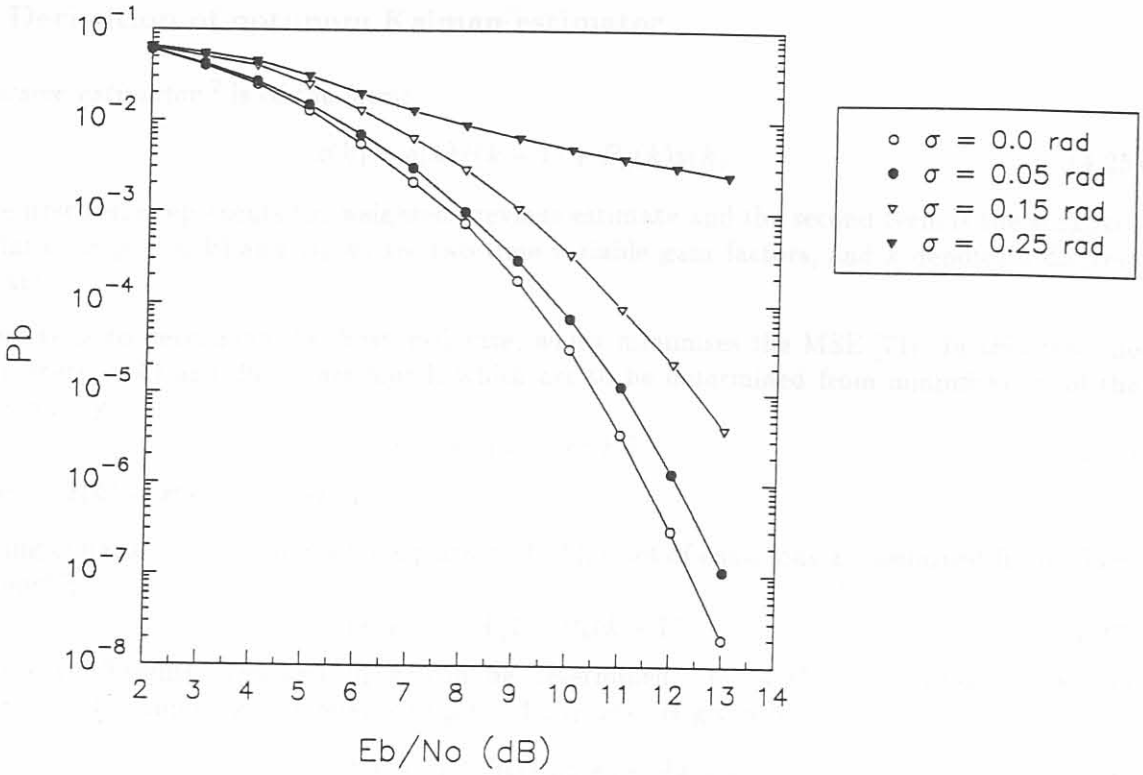

Figure 4.17: BEP sensitivity of Q²PSK to a Tikhonov-distributed phase error.

method is ideally suitable for application in burst-mode communications, where rapid acquisition is required.

Kalman filters have been used for applications in the signal processing field, when very fast identification or estimation of parameters is needed. It is known that the Kalman filter is a minimum Mean Square Error (MSE) estimator when the received signal is contaminated by Gaussian noise, and the observables and the parameters to be estimated are linearly related.

In this section two different, but indeed similar, phase and frequency correction procedures will be analysed.

Firstly the derivation of a optimum Scalar Kalman Estimator (SKE) is considered, which is capable of estimating a scalar time-varying process. In the following section a carrier phase and frequency correction strategy is introduced, which first of all eliminates the phase error and then tracks the frequency information with the derived SKE. In general, a tracking mechanism that unwraps the phase modulated incoming signal before attempting to estimate the unknown phase offset, may result in a better phase estimator than the one that estimates the phase information too. Finally, a dual phase and frequency tracking mechanism, based on the method of Kim [69], will be analysed. For both these correction strategies a step-by-step procedure will be presented.

4.3.4.1 Derivation of optimum Kalman estimator

The *recursive estimator* [2] is of the form:

$$\hat{x}(k) = a(k)\hat{x}(k-1) + B_k(k)y(k) \tag{4.25}$$

where the first term represents the weighted previous estimate and the second term is the weighted current data sample. $a(k)$ and $B_k(k)$ are two time-variable gain factors, and k denotes a discrete time instant.

The objective is to determine the 'best' estimate, which minimises the MSE [71]. In this case the two parameters, $a(k)$ and $B_k(k)$ are found, which are to be determined from minimisation of the MSE, ϵ, given by

$$\epsilon = p(k) = E[e^2(k)] \tag{4.26}$$

where $e(k) = \hat{x}(k) - x(k)$ is the error.

Substituting equation (4.25) for $\hat{x}(k)$ in equation (4.26), a set of equations are obtained from which the relationship

$$a(k+1) = A\left[1 - B_k(k+1)\right] \tag{4.27}$$

between the coefficients $a(k)$ and $B_k(k)$ can be determined. In (4.27), A denotes the system parameter [71]. By applying this relationship to (4.25), $\hat{x}(k)$ is given by

$$\hat{x}(k) = A\hat{x}(k-1) + B_k(k)[y(k) - A\hat{x}(k-1)] \tag{4.28}$$

The first term, $A\hat{x}(k-1)$, represents the best estimate of $\hat{x}(k)$ without any additional information, and it is therefore a *prediction* based on past observations. The second term is a *correction* term depending on the difference between the new data sample and the observation estimate, $\hat{x}(k) = A\hat{x}(k-1)$, multiplied with a variable gain factor, $B_k(k)$.

The realisation of equation (4.28) is depicted in Figure 4.18.

Derivation of expressions for $B_k(k)$ and $p(k)$ are derived in [71]. These expressions are repeated here for sake of convenience:

The variable Kalman gain factor

$$\begin{aligned} B_k(k) &= \frac{p(k-1)}{p(k-1) + \sigma_\nu^2} \\ &= \frac{1}{1 + \sigma_\nu^2/p(k-1)} \end{aligned} \tag{4.29}$$

and the MSE,

$$p(k) = \sigma_\nu^2 . B_k(k) \tag{4.30}$$

where σ_ν^2 denotes the noise variance (or power). Finally, equations (4.28), (4.29) and (4.30) constitute a complete computationally viable estimation algorithm. Substituting $\theta_I(k)$ for $y(k)$, the current input sample, and $\hat{\theta}(k)$ for $\hat{x}(k)$, the best estimate, this estimator can be summarised as follows:

- Estimate over one symbol period:

$$\hat{\theta}(k) = \hat{\theta}(k-1) + B_k(k)\left[\theta_I(k) - \hat{\theta}(k-1)\right] \tag{4.31}$$

[2] Also known as recursive filter.

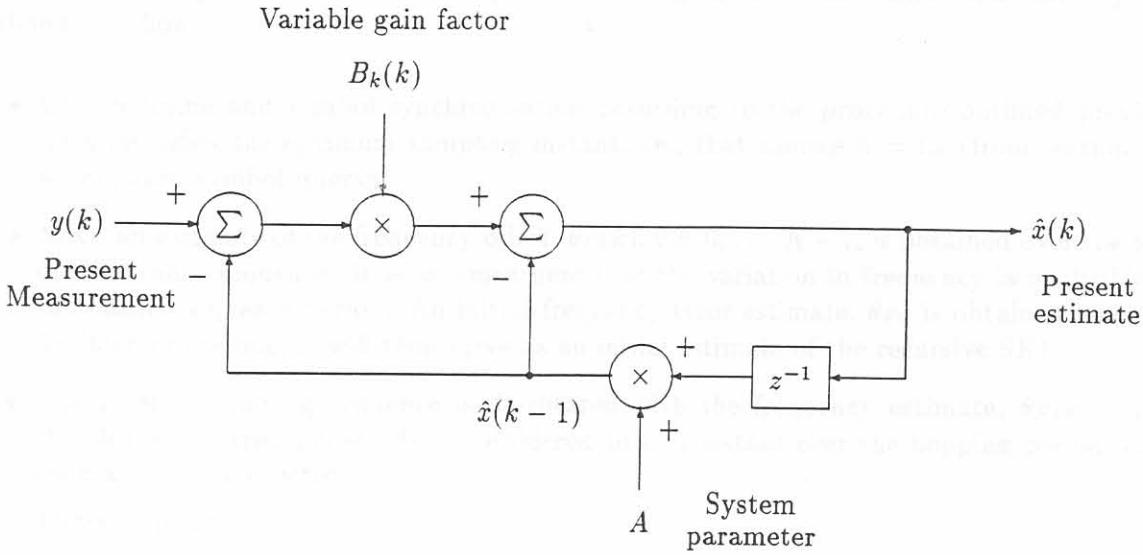

Figure 4.18: Optimum recursive scalar Kalman estimator.

- Calculate the variable Kalman gain:

$$B_k(k) = \frac{p_1(n)}{p_1(n)+\sigma_\nu^2} = \frac{p(0)}{k.p(0)+\sigma_\nu^2} \tag{4.32}$$

where

$$p_1(k) = p(k-1) = p(k \mid k-1) \tag{4.33}$$

- Derive the MSE:

$$p(k) = p_1(k)[1 - B_k(k)] \tag{4.34}$$

Note that, with $k \to \infty$, $B_k(k) \to 0$ and $\hat{\theta}(k) = \hat{\theta}(k-1)$, which indicates that the estimator has stabilised on the best estimate of the required parameter, θ, i.e., further measurements will not improve this estimate.

Equations (4.31-4.34) together form the recursive algorithm. The procedure is to find $B_k(k)$ from equation (4.32). Then, from the stored previous value $\hat{\theta}(k-1)$ and the new data sample θ_I, calculate $\hat{\theta}(k)$ from equation (4.31). This procedure continually generates the best linear estimation of $\hat{\theta}(k)$, and at the same time it provides the corresponding MSE, $p(k)$, which can be monitored to assess the convergence of the recursive algorithm. Note that $p(k) \to 0$ for m very large.

4.3.4.2 Correction Strategy A: Single Estimator

This method has been proposed by Aguirre and Hinedi [72], where the phase information is removed by manipulating the samples in a direct manner. This is done by performing a simple cross-product between the in-phase and quadrature sample, producing modified samples which are then fed into a recursive SKE, which estimates the time-varying frequency information.

The step-by-step procedure in estimating and correcting the frequency offset and carrier phase is outlined as follows:

- Obtain frame and symbol synchronisation according to the procedure outlined previously. This identifies the optimum sampling instant, i.e., that sample $k = i_m$ (from section 4.3.1) within each symbol interval.
- Next, an estimate of the frequency offset, $\hat{\theta}_F(k), k = 0, \ldots, K-1$, is obtained over the length of the header sequence. It is assumed here that the variation in frequency is negligible over the header sequence period. An initial frequency error estimate, $\hat{\theta}_{F0}$ is obtained and stored for later processing; it will then serve as an initial estimate of the recursive SKE.
- The received training sequence is de-rotated with the frequency estimate, $\hat{\theta}_F(k)$, so that the absolute carrier phase offset, considered to be constant over the hopping period, can be estimated and corrected.

  These steps are:

  - Repeat the correlation process, by using the de-rotated received training sequences with the reference training sequence. This corresponds to a correlation between the received training sequence and absolute phases of the reference training sequence. The initial carrier phase rotation, $\theta_o(i_m)$, is then calculated from (4.13) in section 4.3.1. This value is stored and later utilised in the establishment of decision boundaries of the Q²PSK detection scheme.
  - All samples of the received I and Q baseband signals, associated with each two-dimensional modulator, are de-rotated with the fixed carrier phase offset, $\theta_o(i_m)$.
- Finally, after having obtained an initial Doppler estimate during the preamble period, the recursive SKE, can be initiated in order to track and improve the possibly time-variant frequency error during the data period of the FH burst.

4.3.4.3 Correction Strategy B: Dual Estimator

In this section the analysis of a adaptive dual phase and frequency tracking loop will be carried out. This loop was introduced by Kim [69], an *adaptive dual digital phase-locked loop* DPLL. By close investigation of the tracking loop (Figure 2, [69]), one can identify two separate loops, each in the form of a recursive Scalar Kalman Estimator (SKE) depicted in Figure 4.18, for tracking the phase and frequency variations, respectively. The first SKE is used to track frequency variation, while the second, utilising the estimated frequency is used to track phase variation.

The block diagram in Figure 2, [69] is repeated in Figure 4.19, but redrawn to indicate the two estimators more clearly. Henceforth, the name Dual Phase Frequency Kalman Estimator (DPFKE) is adopted for this configuration.

From Figure 4.19, the closed loop transfer function of the DPFKE loop is given by

$$\frac{\theta_P(z)}{\theta_I(z)} = \frac{(B_l(t) + B_k(t)(z-1) + B_l(t)B_k(t)}{z - (1 - B_l(t))(z - (1 - B_k(t)))} \tag{4.35}$$

From equation (4.35), it is clear that the loop is stable as long as $0 < B_k(t), B_l(t) < 1$. The time-varying parameters (Kalman gain factors) $B_k(t)$ and $B_l(t)$, control the bandwidth of the frequency detection and phase detection loops, as well as the overall bandwidth of the DPFKE.

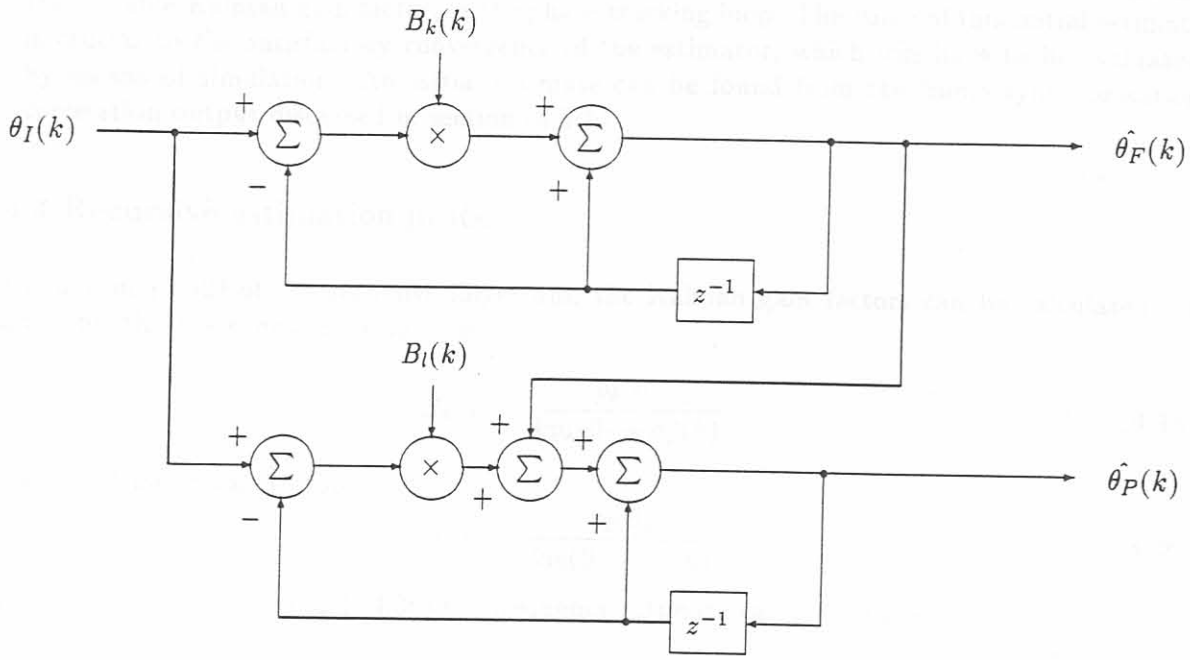

Figure 4.19: Dual Frequency and Phase Kalman Estimator (DPFKE).

From Figure 4.19, the dual frequency and phase estimator outputs at time instant, k, can be expressed as

$$\hat{\theta}_F(k) = \hat{\theta}_F(k-1) + B_k(k)\left[\theta_I(k) - \hat{\theta}_F(k-1)\right] \tag{4.36}$$

and

$$\hat{\theta}_P(k) = \hat{\theta}_P(k-1) + \hat{\theta}_F(k) + B_l(k)\left[\theta_I(k) - \hat{\theta}_P(k-1)\right] \tag{4.37}$$

The DPFKE has the capability of estimating the input frequency immediately, without any limitation in pull-in range. After the initial phase and frequency acquisition, it requires a short header period, since it aligns its phase adaptively based on the minimum MSE criterion at each given time instant [73].

Next, a step-by-step procedure will be given in estimating the phase and frequency offsets by means of the DPFKE. This dual estimator is very similar to the previous single Kalman estimator and almost the same steps are followed, with only slight changes concerning the phase estimation.

- Obtain frame and symbol synchronisation according to the procedures previously outlined.
- Next, an estimate of the frequency offset, $\hat{\theta}_F(k), k = 0, \ldots, K-1$, is obtained over the length of the preamble sequence. Furthermore, an initial frequency error estimate, $\hat{\theta}_{F0}$ is obtained and stored for later processing. It will be used to initialise the recursive DPFKE.
- The received training sequence is de-rotated with the frequency estimate, $\hat{\theta}_F(k)$, so that the absolute carrier phase offset, considered a constant over the hopping period, can be estimated and corrected. Recall, from the previous method that a second cross-correlation was needed to estimate this carrier phase offset. However, in this estimator the tracking of this information is interleaved with that of the frequency estimation. The problem is that no initial estimate

of the phase information is available at this stage. This is needed to obtain an initial value for the variable Kalman gain factor, in the phase tracking loop. The value of this initial estimate is crucial to the satisfactory convergence of the estimator, which will have to be evaluated by means of simulation. An initial estimate can be found from the frame synchronisation correlation output discussed in section (4.3.1).

4.3.4.4 Recursive estimation process

From equation (4.32) of the recursive algorithm, the Kalman gain factors can be calculated, at instant k, for the frequency tracking loop:

$$B_k(k) = \frac{p_k(0)}{kp_k(0) + \sigma_\nu^2(k)} \tag{4.38}$$

and for the phase tracking loop:

$$B_l(k) = \frac{p_l(0)}{kp_l(0) + \sigma_\nu^2(k)} \tag{4.39}$$

Then from equations (4.31) and (4.36) the frequency estimate can be obtained:

$$\hat{\theta}_F(k) = \hat{\theta}_F(k-1) + B_k(k)\left[\theta_I(k) - \hat{\theta}_F(k-1)\right] \; ; \; k = 1, 2, \ldots \tag{4.40}$$

with MSE from equation(4.34):

$$p_k(k) = \frac{p_k(0)}{1 + kp_k(0)/\sigma_\nu^2} \tag{4.41}$$

Similarly equations (4.31) and (4.37) produce the phase estimate:

$$\hat{\theta}_P(k) = \hat{\theta}_P(k-1) + \hat{\theta}_F(k) + B_k(k)\left[\theta_I(k) - \hat{\theta}_P(k-1)\right] \; ; \; k = 1, 2, \ldots \tag{4.42}$$

with MSE from equation(4.34):

$$p_l(k) = \frac{p_l(0)}{1 + kp_l(0)/\sigma_\nu^2} \tag{4.43}$$

By plotting $p_k(k)$ and $p_l(k)$ as a functions of the iteration index k, an indication of the convergence of the recursive loops of the estimators can be obtained. These MSE curves can be smoothed by calculating the running average:

$$\tilde{p}_k(k) = \beta\tilde{p}_k(k-1) + (1-\beta)p(k)_k \tag{4.44}$$

and

$$\tilde{p}_l(k) = \beta\tilde{p}_l(k-1) + (1-\beta)p(k)_l \tag{4.45}$$

starting with $\tilde{p}_k(k-1) = \tilde{p}_l(k-1) = p(0)$ at instant, $k = 1$.

This concludes the discussion of the proposed carrier phase and Doppler frequency error estimation and correction procedures for the Q²PSK modem. The carrier phase and frequency tracking procedures designed and analysed in the foregoing sections have been presented in [74].

4.4 CONCLUDING REMARKS: CHAPTER 4

The research presented in the foregoing was motivated by the need for carrier frequency and phase recovery for Q^2PSK in a mobile communications environment. One of the main shortcomings of the Kalman based carrier estimators is that the acquisition characteristics of these estimators are highly dependent on the initial estimate. The question now is to what extend the initial estimates will be suitable for effective operation. To answer this question, a thorough performance evaluation is needed to verify the correctness and also the effectiveness of these procedures. In Chapter 7, these tracking procedures will be evaluated by means of simulation.