en caso de que no estemos en banda base existe un filtro que recorta la señal donde queremos, antes del filtro receptor, un filtro selector de banda

4 PAM system simulation

In this assignment we will simulate different aspects of Pulse Amplitude Modulation (PAM) communications systems, using Matlab and Simulink. We will start with a simple system and then we will move on to more realistic scenarios, in which the operating conditions are no longer ideal. In particular, we will consider the following impairments:

- Channel Fading
- Additive noise
- Multipath interference
- Phase and frequency offsets between local oscillators of the transmitter and the receiver
- Phase and frequency offsets between the symbol clocks used at the transmitter and the receiver

This analysis will allow us to understand the variety of problems that must be addressed when designing a practical communication system; otherwise, the error rate may be too high, making the system useless. An in-depth study of these problems and their corresponding compensation techniques are left for the digital communication courses in the third and fourth year of the degree.

Important: Questions marked with ★ are mandatory, but are left to the student to complete on his/her own. Those marked with ♣ are optional.

4.1 Simulation of a basic PAM system

In this section, we will work with the Simulink model `bandpass_real_PAM`, which simulates transmission and reception using bandpass PAM with real-valued symbols. Some important features of this model are:

- Transmit and receive filters are square-root raised-cosine (SRRC) pulses. Apart from the noise, the channel is considered ideal.
- Various simulation parameters can be set in the blue block, such as the symbol constellation and the noise variance (power) in the received signal.
- The `General QAM` block employs the minimum distance criterion to estimate the transmitted symbols, and includes their subsequent conversion to bits.

- There is an associated startup file `bandpass_real_PAM_ini.m` that establishes some settings before running the model.
- When running the model you will be able to observe the power spectrum in the spectrum analyzer[1] and the samples at the input of the decision block.
- As you will observe in the spectrum analyzer window, the carrier frequency is set to 10 kHz by default. Thus, in order to perform the simulation correctly, a much higher sampling frequency is needed.

☐ **Question 1**

In this simulation the sampling frequency is determined by the bit rate and the number of samples per symbol period, M, in the transmitter filter. The sampling period is given by $T_s = T/M$, where T denotes the symbol period, which in turn is a function of the bit rate R_b and the constellation used.

a) Obtain the sampling frequency for a bit rate of 1 kbit/s and an oversampling factor $M = 128$.

b) From the simulation parameters, obtain the theoretical values of the bandwidth of the baseband and passband PAM signals. Experimentally check the bandwidth of the transmitted signal (the spectrum analyzer also allows to display the power in Watts.)

c) What is the minimum value of M so that the transmitted signal does not suffer from aliasing? Check your answer by modifying this parameter in the transmit filter (the receiver filter and sampler are automatically adjusted during runtime). Is the simulation of the system correctly implemented? Observe the power spectrum at the input of the receive filter.

d) Repeat part b) for different values of the *roll-off* factor (modify its value both at the transmit and receive filters).

(e) In case we wish to triple the binary rate, describe the possible options and their implications.

■

☐ **Question 2**

a) Let us consider a passband PAM system with a maximum bandwidth (in RF) of 15 kHz and a symbol period of 0.1 ms. If the system must support a bit rate of 30 kbit/s, determine the symbol constellation and the range of suitable *roll-off* values.

[1] The spectrum analyzer does not represent the Power Spectral Density (PSD, typically in dBm/Hz), but rather the power contained within a resolution bandwidth (RBW), which is adjustable. For this reason, it shows power units (dBm, for instance).

b) Now suppose that to support the 30 kbit/s bit rate, while maintaining the same maximum bandwidth, we decide to use a 16-PAM constellation. Obtain the symbol period, T, in ms, as well as the range of appropriate *roll-off* values. What are the advantages and disadvantages of this new system over the previous one?

■

Let us delve deeper into the design of the transmit and the receive filters, which are square-root raised-cosine pulses with the same roll-off factor. Thus, assuming an ideal channel, the effective pulse has a raised-cosine (RC) response (which is a Nyquist pulse) and, moreover, the receive filter is matched to the transmit filter, so that the SNR at its output is maximized.

However, since the raised-cosine pulses (or their square root) are of infinite duration in the time domain, in practice they must be truncated (an operation also known as *windowing*). Typically these pulses are truncated to an integer number of symbol periods, symmetrically around time $t = 0$ (or $n = 0$ in discrete-time). Since the resulting impulse response is non-causal, it is necessary to introduce an appropriate time delay to make them causal. This delay will also affect the signal at the filter output, and this fact must be taken into account in the simulation. Figure 4.1 shows an example of a truncated and time-delayed pulse.

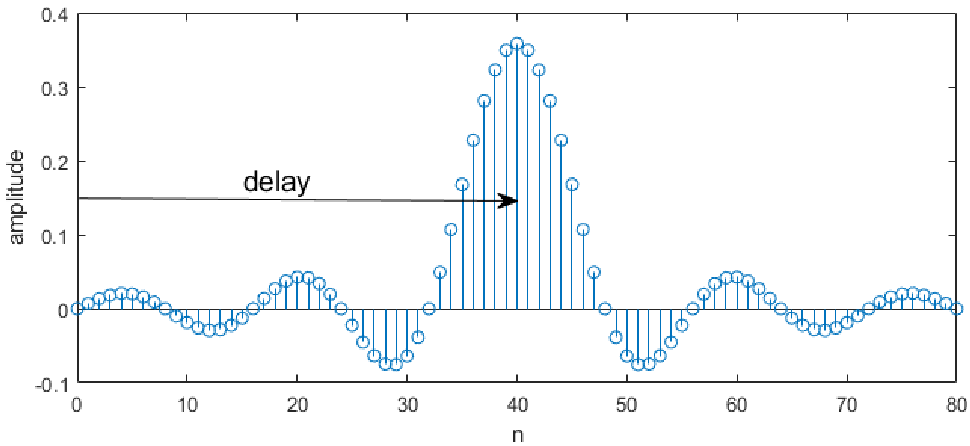

Figure 4.1: Sampled and truncated raised-cosine filter with a length of 10 symbol periods (8 samples per symbol period), including a time delay of 40 samples (half its length) to make it causal.

If we edit the parameters of the transmit pulse, we will observe that there is one of them (*filter span*) that defines the length of the pulse in symbol periods, being the delay introduced by the filter half its length. On the other hand, the convolution of two truncated square-root raised-cosine pulses is not exactly a raised-cosine pulse so that, in practice, some residual intersymbol interference (ISI) will remain, even if we sample at the optimal time instants. This residual ISI will be negligible if the tails of the pulses that have been eliminated after truncation are not significant; this, in turn, will depend on the number of symbol periods retained and on the roll-off factor.

☐ **Question 3**

Open the diagram `RC_group_delay`, which shows the impulse response of a square-root raised-cosine pulse and the convolution of two of these pulses.

a) Observe the impulse response of these pulses, with a length of 10 symbol periods, and the following roll-off factors: $0, 0.2, 0.5, 1$ (you must change this parameter in both filters). For each case, discuss whether they can be regarded as Nyquist pulses.

b) Estimate the delay at the output of every filter, expressed in symbol periods and in samples.

c) Repeat the previous parts for a pulse length of 4 symbol periods.

■

☐ **Question 4**

Once again, consider the diagram `bandpass_real_PAM` with oversampling factor $M = 128$.

a) Estimate the delay introduced by this block diagram, in symbol periods, from the time instant in which a symbol is generated until the time instant in which the corresponding sample at the input of the decision device is obtained.

b) Discuss whether the previous delay must be taken into account when assessing system performance.

c) Adjust now the transmit and receive filters to have a length of 4 symbol periods each, and a roll-off factor of 0.3. Set the noise variance to zero and run the simulation. Explain what happens to the samples at the input of the decision block. Is the bit error rate (BER) of 0.5 realistic?

■

4.2 Eye diagram and soft decisions

Typically, the eye diagram is obtained by observing the signal at the receive filter output in an oscilloscope, synchronizing its time basis with the symbol clock and adjusting the horizontal sweep to cover an integer number of symbol periods in the screen. This results in a representation in which many consecutive signal segments of equal length overlap. In Simulink we will simply use the `Eye Diagram` block.

☐ **Question 5**

a) Insert the block `Eye Diagram` into the model `bandpass_real_PAM` and connect it to the output of the receive filter. Follow your instructor's directions to set the represented time interval to 3 symbol periods. For the time being, set the noise variance to zero.

b) How does the roll-off factor α affect the shape of the eye diagram? In a system with Additive White Gaussian Noise (AWGN), in which the sampling instants were slightly deviated from the optimum, what value of α do you think would be more desirable to minimize the error probability? What price would we be paying in return?

c) Now observe the eye diagrams corresponding to time intervals of 4, 5 and 6 symbol periods. ■

Another useful way to assess the quality of the communication link is to plot the samples at the input of the decision device with respect to time. These samples are estimates of the transmitted symbols and are known as *soft decisions*. In order to plot soft decisions, we simply run the simulation and then execute `plot(1:length(srx), real(srx),'.');` in Matlab.

□ **Question 6**

If the system works properly, soft decisions should take values close to the transmitted data symbols: check that this is the case. To help with visualization, it may be more convenient to plot only a few initial values of the soft decisions and not all of them. Do you think there are samples of `srx` that should be discarded in the resulting graph? ■

4.3 Additive Noise

From a theoretical point of view, we typically assume that the noise introduced by the channel is additive, white and Gaussian with power spectral density $S_n(f) = N_0/2$ W/Hz. As a consequence, the power of the noise process is infinite in theory, although as soon as the noise passes through a filter, it will become band-limited and its power will be finite. In practice, the noise can be regarded as white if its PSD is approximately constant over a wide enough bandwidth.

♣ **Question 7**

In the two scenarios of Fig. 4.2, the process $n_c(t)$ is white with PSD $N_0/2$.

For Scenario (a):

a) Derive the autocorrelation function $R_n(\tau) = E\{n(t+\tau)n^*(t)\}$ as well as the power of $n(t)$. Is $n(t)$ a white process?

b) Obtain the autocorrelation function $R_n[m] = E\{n_{k+m}n_k^*\}$ and the power of the sequence n_k. What condition must f_s satisfy for the resulting sequence n_k to be white? (We say that the sequence is white if every pair of values uncorrelated, that is, $E\{n_k n\ell\} = 0$ for all $k \neq \ell$).

c) Express the bandwidth B as a function of the value of f_s from part (b).

In Scenario (b) the filter is a square-root raised-cosine filter roll-off factor α, corresponding to a digital communication system with symbol period T.

d) Derive the autocorrelation function and power of $n(t)$. Is $n(t)$ a white process?

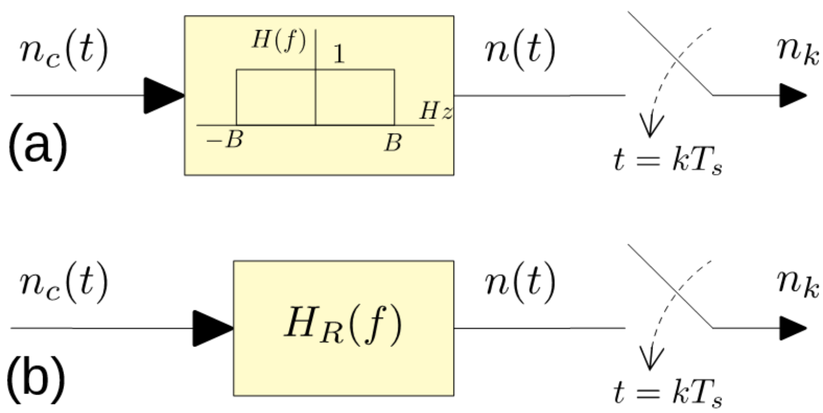

Figure 4.2: Noise filtering and sampling.

e) Obtain the autocorrelation function and power of the sequence n_k. What condition must f_s satisfy for the resulting sequence n_k to be white?

■

In our simulations we will model the noise samples introduced by the channel as a discrete-time Gaussian sequence whose PSD is constant in the range $\omega \in [-\pi, \pi]$ rad (equivalent to $[-f_s/2, f_s/2]$ Hz in the continuous frequency domain), so that its power is uniformly distributed over this interval.

♣ Question 8

Consider again the `bandpass_real_PAM` diagram with the default parameter values. Set the noise power to 0.001 (1 mW).

a) Taking into account the value of the sampling frequency, obtain the PSD of the noise samples in W/Hz and dBm/Hz.

b) Calculate the noise power over a bandwidth of RBW Hz (the *resolution bandwidth* indicated in the spectrum analyzer window). Divide the resulting power into its positive and negative frequency contributions.

c) Express the previous two contributions in dBm and compare them to the noise level observed in the spectrum analyzer.

d) Estimate the noise power at the output of the receiver filter, which has an amplitude gain of $\sqrt{2}$. Does the estimated power depend on the roll-off factor?

e) Observe the effect of the noise on the eye diagram, trying different values for the noise power.

■

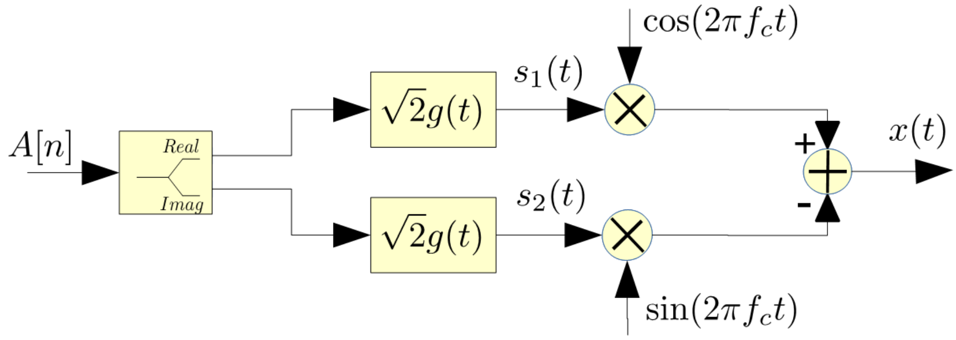

Figure 4.3: QAM Modulator

4.4 Quadrature Amplitude Modulation (QAM)

Figure 4.3 represents the structure of a QAM modulator that generates the modulated signal from the in-phase and quadrature components ($s_1(t)$ and $s_2(t)$, respectively).

★ Question 9

In the diagram `puzzle`, organize the blocks and make the appropriate connections to perform the complete simulation (modulator and demodulator) of a 4-QAM (also known as QPSK) system. Note that the transmit and receive filters include an amplitude gain of $\sqrt{2}$ to compensate for the attenuation introduced at the modulation and the demodulation stages.

a) Taking into account the value of the sampling frequency, compute the noise PSD in W/Hz for a noise power of 0.1 W.

Estimate the noise power at the output of the receive filters.

c) Observe the effect of the noise in the eye diagram, trying different values for the noise power.

■

☐ Question 10

The diagram `bandpass_complex_PAM` is completely equivalent to the one you have just built.

a) Obtain the expression of the transmitted signal for a sequence of complex symbols $A[k]$ and a transmit filter $g(t)$.

b) Check that the simulation of this new diagram is equivalent to the previous one. For a noise power of 0.1 W, compare the error probabilities obtained with both models.

Consider now the diagram `lowpass_equi_PAM` .

c) Under what conditions is this diagram equivalent to the bandpass simulation?

d) What is the main advantage of this new scheme?

e) Relate the power of the real and imaginary parts of the noise in this scheme with the noise power in the two previous diagrams.

f) Without changing the number of samples per symbol period (oversampling factor), adjust the remaining parameters of the system so that its operation is equivalent to that of the previous diagram (assuming that the conditions of part c) are met).

■

4.5 Discrete-time equivalent channel

As we already know, in the PAM system the samples at the input of the decision block are given by:

$$\begin{aligned} q[n] &= \sum_{k=-\infty}^{+\infty} A[k]p((n-k)T) + z[n] \\ &= A[n] \star p[n] + z[n], \end{aligned}$$

where $p[n]$ and $z[n]$ are, respectively, samples from the equivalent pulse $p(t)$ and from the noise at the output of the receive filter, taken at symbol rate. This expression is valid for both real and complex constellations, with the only caution that in the latter case the noise is also complex. Therefore, if we are only interested in modelling $q[n]$, which suffices, for example, to estimate the probability of error, we can use the scheme in Figure 4.4, the so-called *discrete-time equivalent channel*. Sometimes, depending on $p(t)$, this scheme can be further simplified, as we will show below.

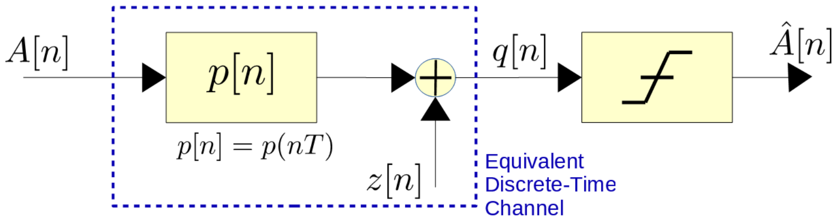

Figure 4.4: Equivalent discrete-time channel model.

□ Question 11

Consider again the diagram `lowpass_equi_PAM`. Notice that the raised-cosine block generates $M = 16$ samples per symbol period and that, later, we discard all but one of those M samples, which is the input to the decision block. Save the diagram with a different name, and then make the appropriate modifications (also to its corresponding initialization file (a copy of `lowpass_equi_PAM.ini`)) in order to estimate the same BER but working at the symbol rate. ■

We have just shown that, under certain conditions, we can estimate the symbol error rate (SER) or the bit error rate (BER) by simply generating the sequence $q[n] = A[n] + z[n]$ and applying it to the decision block. Let us use this concept to make multiple simulations of a communication system and estimate its SER and BER for different noise powers. In this way, we will be able to plot the typical error probability curves as a function of the ratio E_s/N_0 or E_b/N_0.

SER_BER_4PAM.m

```
L = 1e6; % number of transmitted bits.
VARN = logspace(-1,2,30); % 30 log-spaced samples between 0.1 and 100.
ser = []; ber = [];
conste = [-3 -1 3 1]; % 4-PAM
N = length(conste);
Es = sum(abs(conste).^2)/N; % Average symbol energy
Eb = Es/log2(N); % Average bit energy
for i = 1:length(VARN),
  i,
  varnoise = VARN(i); % noise power
  b = randn(L, 1) > 0; % bit sequence
  m = bits2pam(b); % symbol sequence
  z = m + sqrt(varnoise)*randn(size(m)); % soft decisions
  simb = quantalph(z,conste)'; % hard decisions
  mdec = pam2bits(simb); % decoded bits
  ser(i) = mean(m ~= simb); % symbol error rate
  ber(i) = mean(b ~= mdec); % bit error rate
end
EsN0 = Es./(2*VARN); % Es/N0 ratio
EbN0 = Eb./(2*VARN); % Eb/N0 ratio
% SER as a function of Es/N0
figure (1); semilogy(10*log10(EsN0), ser); grid,
title('Symbol Error Rate (SER)');
xlabel('Es/N0 (dB)'); ylabel('SER'); legend('Simul.');
```

☐ **Question 12**

The script `SER_BER_4PAM.m` contains the code to simulate a 4-PAM system (using the discrete-time equivalent channel model) for different noise powers and to estimate its SER and BER. It also plots the SER as a function of E_s/N_0 in dB. Now you should complete the code to plot:

a) the estimated and theoretical SER as a function of E_s/N_0 in dB, both in the same figure.

b) the estimated and theoretical SER as a function of E_b/N_0 in dB, both in the same figure.

The BER is limited to $\frac{1}{m}\mathrm{SER} \leq \mathrm{BER} \leq \mathrm{SER}$, where m is the number of bits per symbol.

c) Explain why these bounds apply.

d) Plot the estimated BER and both the upper and lower bounds as a function of E_b/N_0 in dB, all in the same figure.

e) Does the BER depend on the bit-to-symbol mapping that has been chosen? For a given constellation and noise power, could we reduce the BER by changing this mapping? Replace the instructions `bits2pam(b)` and `pam2bits(simb)` with `bits2pam(b,1)` and `pam2bits(simb,1)`, respectively, and repeat the previous experiment. Explain what you observe.

■

4.6 Flat-fading channel

In practice, the channel will seldom be as benign as in the previous simulated models. One of the possible impairments introduced by the channel is the so-called **flat fading**, characterized by fluctuations in the channel attenuation over time. The term *flat* refers to the fact that all frequency components of the transmitted signal are affected by the same attenuation[2]. This phenomenon is very common in wireless systems due to events such as changes in atmospheric conditions.

♣ **Question 13**

The diagram `bandpass_real_PAM_flat_fading` simulates a 4-PAM system where the channel introduces an attenuation that can be varied at runtime by means of a scroll bar. It also includes a switch that allows resetting BER estimation during runtime. In this way, the BER can be estimated for different attenuation values without restarting the simulation. By default, the noise power is preset to a very small value (10^{-6}).

a) Check the effect of channel attenuation on the simulation performance.

b) In the absence of other error sources, what would be the maximum attenuation that the channel could introduce without causing detection errors?

[2] When this is not the case and different frequency components suffer different attenuation, the corresponding term is *frequency-selective fading.*

c) If the channel provided a gain higher than one, could errors occur? If so, for what gain values? ■

♣ **Question 14**

The diagram `lowpass_real_PAM_flat_fading_AGC` simulates the same 4-PAM system as above, but in the baseband (therefore we can use a much lower sampling frequency). Additionally, an Automatic Gain Control (AGC) block has been included in the receiver to ensure a prescribed signal power at the receiver input. The AGC estimates the signal power at its input and, using an adaptive algorithm, adjusts its gain. The adjustment speed of the AGC depends on the *step size*, a parameter of the algorithm.

a) Check that the system is working properly. Observe the influence of the step size on the algorithm performance.

b) If the AGC perfectly compensated the channel attenuation, do you think that its BER would be identical to that of the system without attenuation (and without AGC)? ■

4.7 Simulation of a more realistic system

We will now simulate a bandpass PAM system, using the model `bandpass_complex_PAM_channel`, which will allow us to study the influence of important sources of error that are present in practice, as follows:

- **Channel response**. So far, except for the noise, we have considered the channel to be ideal. In this new simulation we will be able to model the channel as the response of an FIR filter.
- **Carrier recovery**. In practice, the frequency and phase of the carrier used in the receiver are estimated from the received signal. The model we will use does not perform carrier recovery, but it does simulate the effect of a frequency and/or phase error between the local oscillators used at the transmitter and the receiver.
- **Clock recovery**. In practice, the estimation of the time instants to sample the signal at the matched filter output is done by the receiver based on the received signal. This clock recovery stage is not included in the model either, but we will be able to observe the influence of estimation inaccuracies. For example, the value of the symbol interval T may be a little different at the receiver with respect to that used by the transmitter.

In the `Global` block of the model, we can modify the appropriate parameters to observe the influence of the different impairments. The errors in carrier frequency and symbol period estimates is modelled as a variation of these parameters in the receiver, and they are specified as a percentage with respect to the transmitter values. Thus, for example, a carrier frequency deviation of 1% is modelled as a receiver frequency of $1.01f_c$. Similarly, a deviation of 15% in the symbol period is equivalent to using a period of $1.15T$ in the receiver.

In order to not complicate the simulation, the parameters related to synchronism are always rounded to an integer number of samples, so there may be small discrepancies between the values set for these parameters and those actually used in the simulation. Thus, for example, if the number of samples per symbol period is $M = 128$, a sampling phase deviation of 15% will be rounded to a sampling time shift of 19 samples.

The influence of the different impairments on system performance will be analyzed separately next.

4.7.1 Multipath

Multipath interference occurs whenever the signal reaches the receiver after travelling through several different paths, thus suffering different attenuations and delays. As a consequence, not all spectral components of the signal experience the same attenuation. In this case the channel can be modelled as a linear filter with an appropriate frequency response[3].

As seen in the classroom sessions, for a baseband PAM system the effective pulse is given by

$$p(t) = g(t) \star h(t) \star f(t)$$

where $g(t)$, $f(t)$ and $h(t)$ are the transmit pulse, receive filter, and channel impulse response, respectively. Clearly, even assuming that the convolution of the transmit pulse and the receive filter is Nyquist, the effective pulse $p(t)$ need not be Nyquist for some channel impulse responses, and therefore the receiver will be affected by intersymbol interference (ISI).

☐ Question 15

Consider a baseband PAM system where the transmit pulse and receiver filter are both square-root raised-cosine pulses, whereas the channel response is:

$$h(t) = \delta(t) - 0.2\delta(t - mT),$$

where m is an integer and T is the symbol period.

a) Derive the discrete-time equivalent channel response, $p[n] = p(nT)$.

b) In the absence of noise, express the samples at the input of the decision block in terms of the sequence of data symbols.

c) Discuss the two previous points when m is not an integer.

[3] If, in addition, the attenuation and/or delays of the different paths are changing over time, the resulting effect is known as *frequency-selective fading*.

■

For a bandpass PAM system, the effective pulse is given by

$$p(t) = g(t) \star h_e(t) \star f(t)$$

where $h_e(t)$ is the channel response shifted to baseband, that is, $h_e(t) = h(t)e^{-j2\pi f_c t}$. $h_e(t)$ is known as the *equivalent lowpass channel* and, in general, will be complex-valued. Therefore, its frequency response will not be conjugate symmetric with respect to the zero frequency.

♣ Question 16

Now consider a band-pass PAM system with the same filters and channel as the previous question.

a) Obtain the response $h_e[n] = h_e(nT)$.

b) Particularise the previous result for the case where the product $f_c \cdot T$ is an integer.

c) Derive the equivalent discrete-time channel response, $p[n] = p(nT)$.

■

□ Question 17

Open the model `bandpass_complex_PAM_channel`. For a very small noise power (say 10^{-6}), $R_b = 1000$ and $f_c = 10^4$, set the remaining parameters to their ideal values.

a) Set the vector `[ 1  zeros(1,127) 0.2 ]` as the FIR channel coefficients. Explain what you observe in the simulation.

b) Now modify the binary rate so that $R_b = 1024$. Explain what you observe now in the simulation.

After finishing this question, set again $R_b = 1000$. ■

♣ Question 18

Open the diagram 'bandpass_complex_PAM_channel'. For a very small noise power (10^{-6}), $R_b = 1000$ and $f_c = 1e4$, set the other parameters to their ideal values. Run the simulation so that all the necessary variables are initialized (you can stop it immediately). This defines two vectors $ch1$ and $ch2$ that correspond to channels with weak and strong ISI, respectively.

a) With the help of `fvtool`, observe the impulse and frequency responses of both channels. In both cases, measure and compare the maximum peak-to-peak variation of its magnitude.

b) In the block diagram set the channel coefficients to the vector $ch1$. Run the simulation for noise powers of 10^{-6} and 10^{-3}. Observe the graphs and the BER value in both cases.

c) Repeat section b) with channel $ch2$. What conclusions do you draw from the results?

■

4.7.2 Carrier Phase Errors

As we already discussed in Assignment 2, for a DSB-SC system the impact of a constant phase error ϕ between the transmitter and receiver local oscillators is the attenuation of the received signal by a factor equal to the cosine of that deviation. In the absence of other degradations, the input to the decision block is then given by $q[n] = A[n] \cdot \cos\phi$.

☐ **Question 19**

Consider the model `bandpass_complex_PAM_channel` with noise power 10^{-6} and an ideal channel.

a) Run the model for different values of the phase error. Observe their effect on the receiver.

b) Derive the maximum phase error ϕ for which, in the absence of noise, no symbol errors occur. Validate your answer via simulation.

■

4.7.3 Carrier Frequency Errors

As shown in Assignment 2, in the presence of a small frequency error f_Δ between the transmitter and receiver local oscillators, a residual modulation remains in the DSB-SC demodulated signal, so that the input to the decision block is given by $q[n] = A[n] \cdot \cos(2\pi f_\Delta nT)$.

☐ **Question 20**

- Run the model `bandpass_complex_PAM_channel` for different values of the frequency error (0.01%, for example), and observe the effect on the receiver.
- For an error of 0.02%, what is the period, in samples, of the residual modulation present in $q[n]$? Validate your answer via simulation.

■

4.7.4 Sampling Period and Sampling Phase Errors

★ **Question 21**

Consider the model `bandpass_complex_PAM_channel`:

a) Run the simulation for different values of sampling phase deviation (between 0% and 100%). Explain what you observe.

b) Repeatedly run the simulation with a very small noise power and a deviation of $\pm 1\%$ and $\pm 2\%$ in the symbol period. Stop the simulation to be able to observe how the input to the decision block "collapses periodically". Why do you think this happens? Can you relate the period of these "collapse events" to the simulation parameters?

■

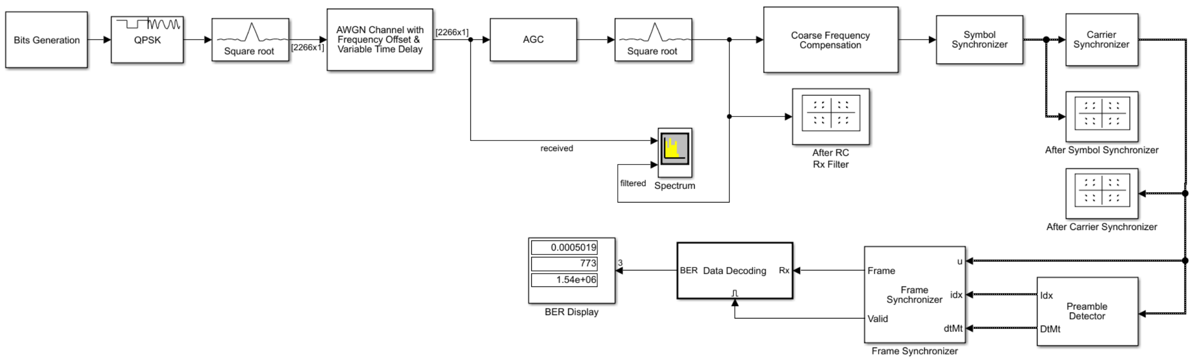

Figure 4.5: QPSK transmitter and receiver with carrier and clock recovery.

4.8 Other stages of a digital communication system

Consider the block diagram in figure 4.5, which performs (baseband) simulation of a (bandpass) QPSK-modulation based digital communication system. This figure is an expanded version of the block diagram `commqpsktxrx`, whose different components will be discussed next. Some aspects are novel with respect to the models used up to this point:

- The bit source includes a header that is periodically transmitted, inserted among the data bits. Data bits are randomized using a 'scrambler' in order to prevent long runs of identical bits from occurring (such long runs would have a negative impact on clock recovery). Of course, the receiver locates and removes headers, and it also applies the reverse process ('descrambling') to the data bits.
- In addition to the typical additive white Gaussian noise (AWGN), the channel also introduces a frequency error and a variable delay.
- The receiver includes the tasks of carrier recovery (in two stages) and clock recovery[4], which will compensate for the frequency and phase errors introduced by the channel, and estimate the appropriate sampling instants to obtain the soft decisions.

☐ Question 22

Open the diagram `commqpsktxrx` [5]. Follow your instructor's directions to analyse the parameters of the model. Run the simulation and interpret the results. ■

[4]Carrier and clock recovery techniques are part of the third-year course syllabus *Principles of Digital Communications.*

[5]Use openExample('comm/QPSKTransmitterAndReceiverSimulinkExample') in recent Matlab versions.

4.9 Real system with QPSK modulation

Next, we describe the two hardware devices to be used in this section for the transmission and reception of digitally modulated signals. Both devices will be controlled through Simulink, as certain parts of the modulation and demodulation processes will be implemented in software (*Software Defined Radio*, SDR).

A large portion of this section has been taken from the book [1], which is available free of charge at `http://www.desktopsdr.com`, and is highly recommended for students willing to deepen in the study of digital communications and interested in SDR.

4.9.1 The transmitter

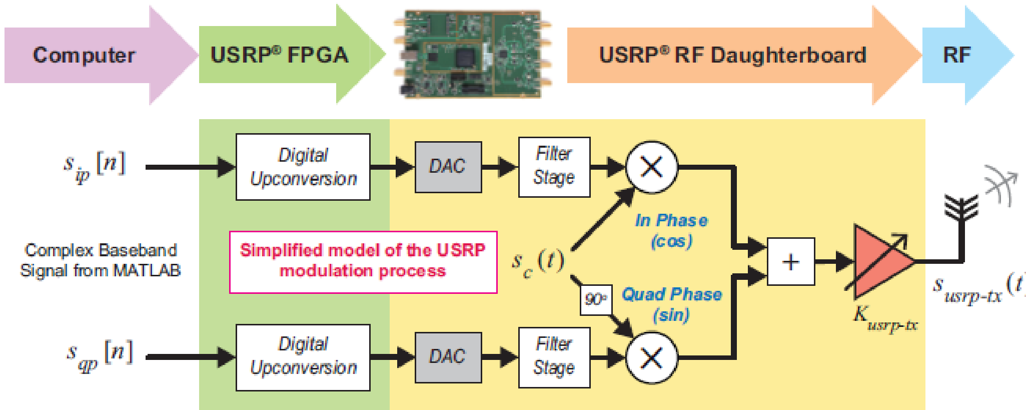

Figure 4.6: Block diagram of the USRP transmitter [1].

The hardware we will use to synthesize the transmit signals generated in Simulink belongs to the family of the *Universal Software Radio Peripherals* (USRP), which use programmable FPGAs for the transmission and the reception procedures. It is manufactured by *Ettus Research* and it is known as USRP N210. Its technical characteristics can be consulted at `https://www.ettus.com/product/details/UN210-KIT`. For this assignment, it suffices to understand its operation from a conceptual point of view, as shown in Figure 4.6, where you can observe how the modulated signal is generated from any complex baseband signal, and then amplified and fed to the transmit antenna.

Figure 4.7 shows an example where the complex baseband signal may correspond to a QPSK modulation or a DPSK[6]. The data bits represent the indefinite transmission of an ASCII message ('Hello World!') along with a three-digit decimal counter (from 000 to 100). The binary sequence is preceded by a header (expanded Barker sequence) to form 170-bit frames as the output of the

[6]Differential Phase Shift Keying – In this case the information resides in the phase difference between consecutive symbols.

block *ASCII Transfer Binary Source*. Then, each pair of bits is transformed to its corresponding QPSK symbol, so that at the transmit filter output we obtain the digitally modulated baseband signal. The next block *USRP Tx Prep* increases the sampling frequency of the baseband signal (this is done by generating new samples via interpolation), and the resulting sequence is the input to the *SDRu Transmitter* block. This block is part of the *Communications System Toolbox Support Package for USRP®* library, which communicates with the USRP via an Ethernet connection. Some of its parameters are: the IP address of the device, the transmitter center frequency (in Hz, in the 25 MHz – 1.75 GHz range), the frequency offset of the local oscillator (LO) for an intermediate frequency stage, and the interpolation factor. Besides, the transmitter gain setting governs the analog and digital gains of the USRP.

In this assignment we will use a center frequency of 914 MHz, an LO frequency of 250 kHz and an interpolation factor of 500. The transmitter gain can vary from 10 to 20 dB.

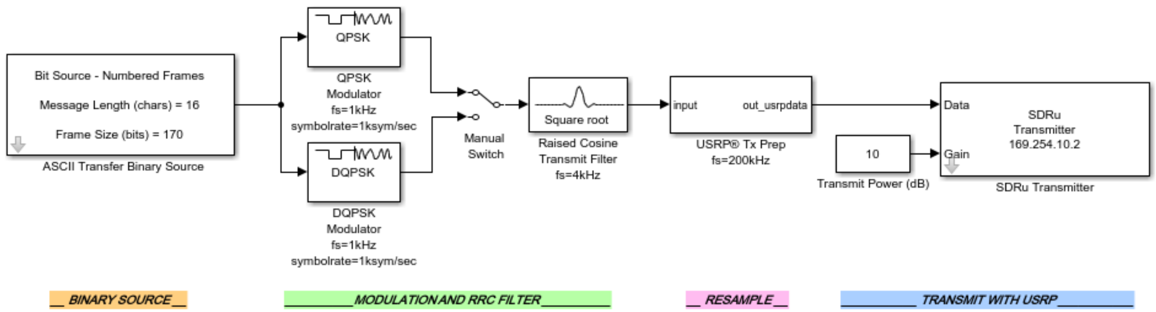

Figure 4.7: QPSK Transmitter with USRP.

□ Question 23

Now you will work with the diagram `usrp_QPSK_ascii_message`. Before you open it, you need to use the Matlab command *addpath()* to indicate where the 'rtlsdr_book_library' is. From the block diagram, determine the values of the bit and symbol rates. Explain the meaning of the parameters in the block that generates the square-root raised-cosine pulse. How many samples of this pulse are generated per symbol? ■

4.9.2 The receiver

As receiver, we will use the RTL-SDR device already seen in the previous assignment. The Simulink model of the receiver is shown in Figure 4.8.

The first block is the *RTL SDR Receiver* with input parameters for frequency and gain tuning. Within this block, we can specify the sampling rate, in this case 240 ksamples/s, and the number of samples per frame (4096). The signal is then decimated (i.e., the sampling rate is reduced by

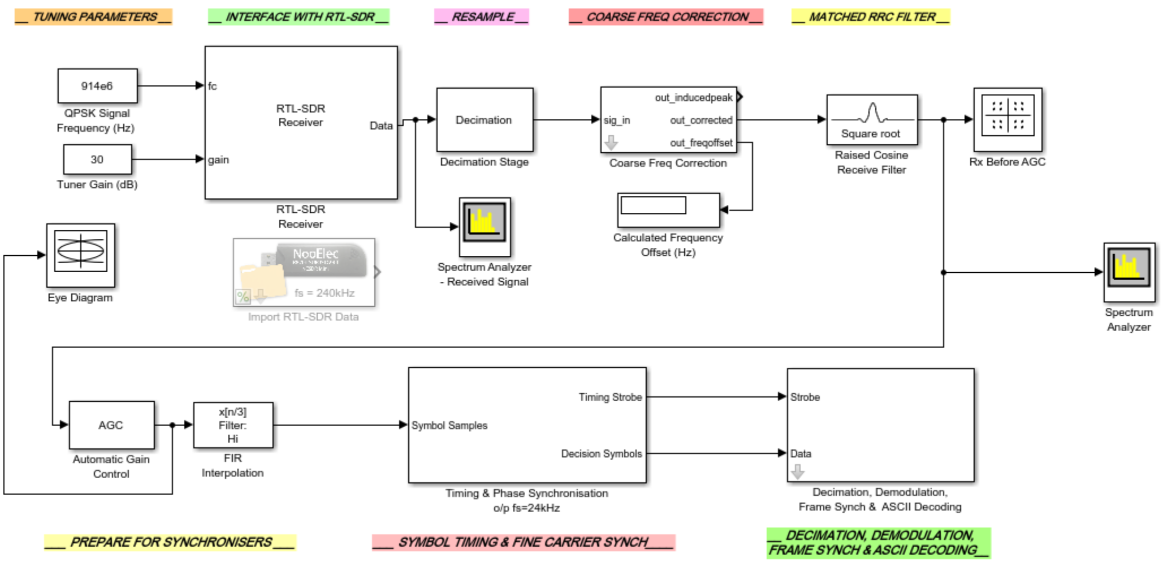

Figure 4.8: QPSK Receiver using RTL-SDR.

some factor), and a coarse frequency correction is carried out. Then, the signal passes through the matched filter.

Next, the automatic gain control (AGC) and the interpolation blocks prepare the signal for the subsequent stages of carrier and clock recovery. The final block includes the decision device, frame alignment, and the decoding of the transmitted message. A detailed study of all these blocks is out of the scope of this course, but it is interesting to understand them conceptually to get an idea of the complexity of the receiver.

☐ Question 24

Open the diagram `rtlsdr_QPSK_ascii_message`. Use the command *addpath()* if necessary. Execute this diagram to observe:

- The spectrum of the received signal in the baseband.
- The constellation at the output of the matched filter.
- The spectrum of the output of the matched filter. Does it correspond to what you would expect?
- The eye diagram after the automatic gain control.
- The constellation after the symbol-rate sampler.

- The compensated constellation.

To observe the decoded symbols it is necessary to activate the diagnostic display (at the centre of the bottom edge of the model) while the simulation is running. The model can be also run in the absence of hardware, from data stored in a file. To do so, simply remove the RTL-SDR block and connect the *Import RTL-SDR Data* block in its place.

■

Bibliography

[1] R. W. Stewart, K. W. Barlee, and D. S. W. Atkinson. *Software Defined Radio Using MATLAB & Simulink and the RTL-SDR.* Strathclyde Academic Media, 2015.