Digital Communications

Laboratory

Erlangen – January 2018

Contents

0.1 General Information

Instructor:	Ebrahim Amiri
Class time:	room R4.10
Dates:	Lab #1: — Lab #2: — Lab #3: — Lab #4: — Lab #5: — Lab #6: —
Prerequisites:	Digital Communications (DiCo in WS)
Credits	2.5 ECTS credits (= work load of 75 hours time)
Reference:	Johannes Huber, *Digital Communications*, lecture notes
Lab work:	6 lab exercises
Homework:	6 homework sets due in class before lab work
Grading:	on passed/not passed basis

0.2 Lab Guidelines

In the Lab

There are six labs in this course. Students are required to complete each lab during their assigned time slot (length: approx. 5 hours) on-site. In the lab students work in pairs. At least one teaching assistant will be present to help the students and to check their work.

The work in the lab is to solve the problems marked by an **L** in this manual. An example lab exercise is shown in **L-0.1**.

Lab Exercise L-0.1

In the first lab solve all the lab exercises marked by **L-1.X** (with **X**$\in \mathbb{N}$).

Homework

Students are required to come prepared with the pre-lab reading work. In addition, each of the seven labs has a set of homework problems marked by an **H**. For example in **H-0.1** your first homework is given.

Homework H-0.1

Carefully read the instructions for Lab 1 and solve all the homework problems marked by **H-1.X**!

Collaboration on solving the homework problems is encouraged. Homework assignments *(one per pair)* should be turned in to the teaching assistant in charge no later than the beginning of the assigned time slot. Your assignments should be written neatly, should contain clear and complete solutions, and should be stapled if consisting of multiple sheets.

Grading policy

Grading is done on a passed/not passed basis. In order to pass, at least fair results in all homework sets and lab exercises have to be achieved.

Project 1

Digital Transmission of Data

1.1 Introduction, Background, and Motivation

In this first experiment you will study the basics of digital data transmission. Today, digital transmission techniques are widely used in many common applications, e.g. mobile telephony, digital video broadcasting etc. Some of the advantages of digital transmission techniques over analog schemes like amplitude modulation and frequency modulation are a much better power efficiency and a tremendous flexibility. The focus of this experiment is on the most popular technique for digital data transmission, so-called pulse amplitude modulation (PAM).

The experiment comprises two main parts: first, the structure of a generic digital PAM transmitter is analyzed. Second, the receiver of such a system is studied. The latter requires to establish an entire digital transmission system consisting of transmitter, channel, and receiver. Detailed descriptions and discussions of the optimal design of digital PAM transmission systems are part of any textbook on digital communications, e.g. [Pro00, Hay00]. Here, in the context of this lab course, we use the system model and the notation introduced in the lecture notes to "Digital Communications" by Prof. Huber [Hub11a].

1.2 Purpose

The aim of this experiment is to demonstrate the effectiveness and efficiency of digital transmission schemes. You will learn how to design a transmitter for digital pulse amplitude transmission and how to represent the digital data by complex coefficients. You should understand the design rules and constraints for the basic pulse shape in the time domain and the frequency domain and know how to develop the optimal receiver for digital transmission systems. You will get to know the effects of some transmission disturbances on the received signal.

1.3 Lab Environment

In this experiment you work with a MATLAB-based simulation environment which can model several digital transmission techniques. The lab computers are equipped with powerful analog-to-digital converters (ADC) and digital-to-analog converters (DAC). In the first part of the experiment, just the DAC are used to generate physical signals from the software-based digital transmission model. That is, the digital transmission schemes selected and simulated in the software are analyzed based on "real" voltages and currents.

In the second part, two of the lab computers are combined into a complete digital transmission system comprising transmitter, channel, and receiver. One of the computers acts as the transmitter, i.e., the MATLAB-generated signals are D/A converted and transmitted to the receiver using patch cables (coaxial cables with BNC connectors). The other computer acts as the receiver; the physical signal is A/D converted and then again fed into a MATLAB-based receiver structure. The (AWGN) channel is modeled by an electronic noise generator.

The physical signals can be analyzed using the Tektronix oscilloscopes or the Agilent multimeters.

1.3.1 Transmitter

In order to simulate the transmitter of a digital transmission system the respective MATLAB program has to be started. This done by starting MATLAB and then typing

```
experiment1_transmitter
```

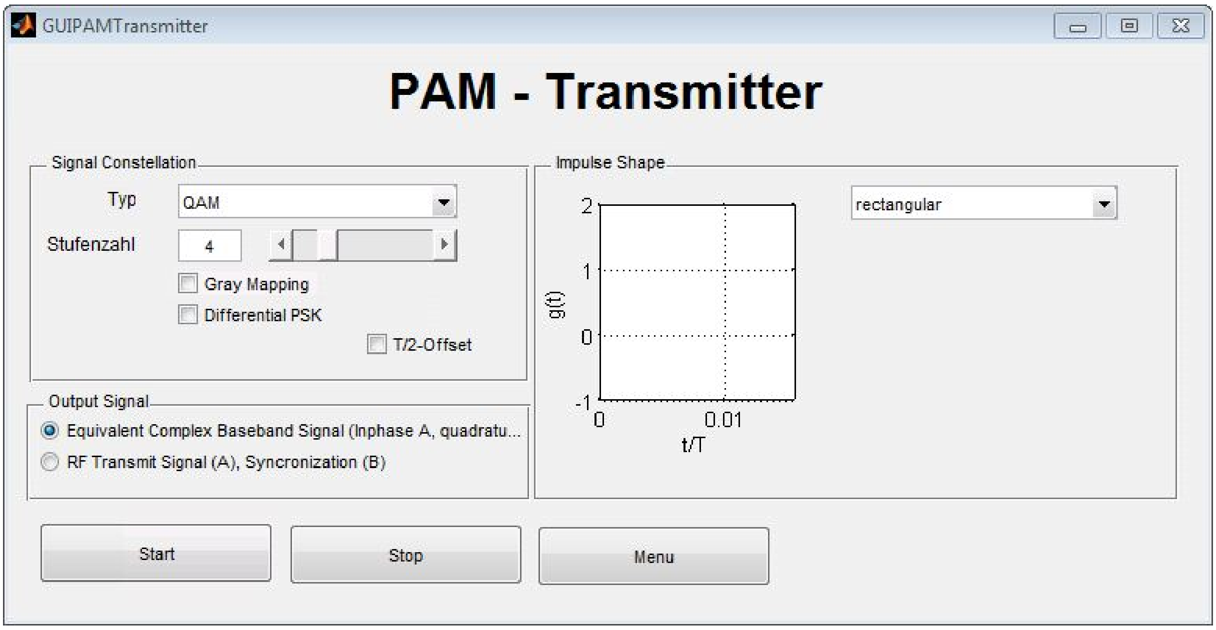


Figure 1.1: MATLAB GUI of PAM transmitter.

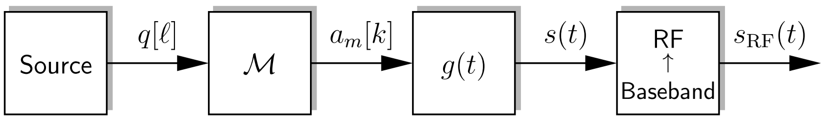


Figure 1.2: Block diagram of transmitter structure of a digital transmission system.

into the command window. Then, you should see a MATLAB GUI as depicted in Fig. 1.1.

The transmitter is built from several individual blocks as depicted in Fig. 1.2. These blocks are subsequently described in some more detail.

1.3.1.1 Data Source

The data source emits a pseudo-random sequence of binary symbols $q[\ell] \in \{0, 1\}$. Zeros and ones occur with equal probabilities.

1.3.1.2 Binary Labeling (Bit Mapping)

As we focus on digital PAM, the binary labeling in block $\mathcal{M}$ assigns binary m-tuples to (complex) amplitude coefficients $a_{m[k]}$.

The system supports several different sets of amplitude coefficients, i.e., signal constellations. In Table 1.1 a list of the implemented signal constellations and constellation sizes is given.

Table 1.1: Supported signal constellations

	Size of Constellation					
Type	2	4	8	16	32	64
ASK (unipolar)	×	×	×	×	×	×
ASK (bipolar)	×	×	×	×	×	×
PSK	×	×	×	×	×	×
QAM		×		×		×
STAR				×		
CROSS					×	

1.3.1.3 Pulse Shaping

The transition from the discrete-time domain signal to the (usually still complex-valued) continuous-time domain signal (complex baseband) in a digital transmission system is performed by the pulse shaping. In PAM transmission the basic pulse shape $g(t)$ is simply weighted with the (complex) amplitude coefficients $a_{m[k]}$. The (spectral) shape of the pulse significantly influences the spectrum of the RF signal; the waveform of the pulse shape in time-domain domain is crucial for a transmission not suffering from inherent interference.

In the simulation environment used in this experiment, the basic pulse shape is still located in the digital domain. That is, the continuous-time signal is simulated by an over-sampled discrete signal.

The equivalent complex baseband signal can be measured at the output of the D/A converters if the respective check box is set. The inphase component is at channel A, the quadrature component at channel B.

1.3.1.4 RF Modulation

The radio-frequency signal is generated from the baseband signal by multiplying with sinusoidal waveforms with the carrier frequency f_c. In the simulation environment this step is also implemented in MATLAB, i.e., even the modulation into a radio-frequency signal is software-based. The physical signal which can be then analyzed at channel A of the D/A converter is the result of a simple D/A conversion of an entirely digitally modeled block diagram. The second output at channel B is a synchronization signal required for the combination of two computers into an entire transmission system.

1.3.2 Receiver

The receiver can be started by typing

```
experiment1_receiver
```

into the MATLAB command window. Please make sure, that you do not run the transmitter and the receiver in parallel using two instances of MATLAB. The MATLAB GUI for the receiver is shown in Fig. 1.3.

The incoming signal at the computer acting as receiver is first A/D converted and then processed in an entirely MATLAB-based receiver structure which is sketched in Fig. 1.4. The individual parts of this block diagram are briefly explained below.

1.3.2.1 RF/Baseband

First, the incoming noisy radio-frequency signal $e_\mathrm{RF}(t)$ is demodulated, i.e., a baseband signal $e(t)$ is generated by multiplying with the carrier and low-pass filtering.

1.3.2.2 Filtering

In the baseband, the next step at the receiver is a (matched) filter to reduce the noise and optimize the SNR for the sampling. The filters impulse response can be chosen from a given selection of pulse shapes.

1.3.2.3 Sampling

After the input filter, the pseudo-continuous signal $a(t)$ is sampled in order to obtain a stream of discrete symbols $d[k]$ on the symbol raster of T_s.

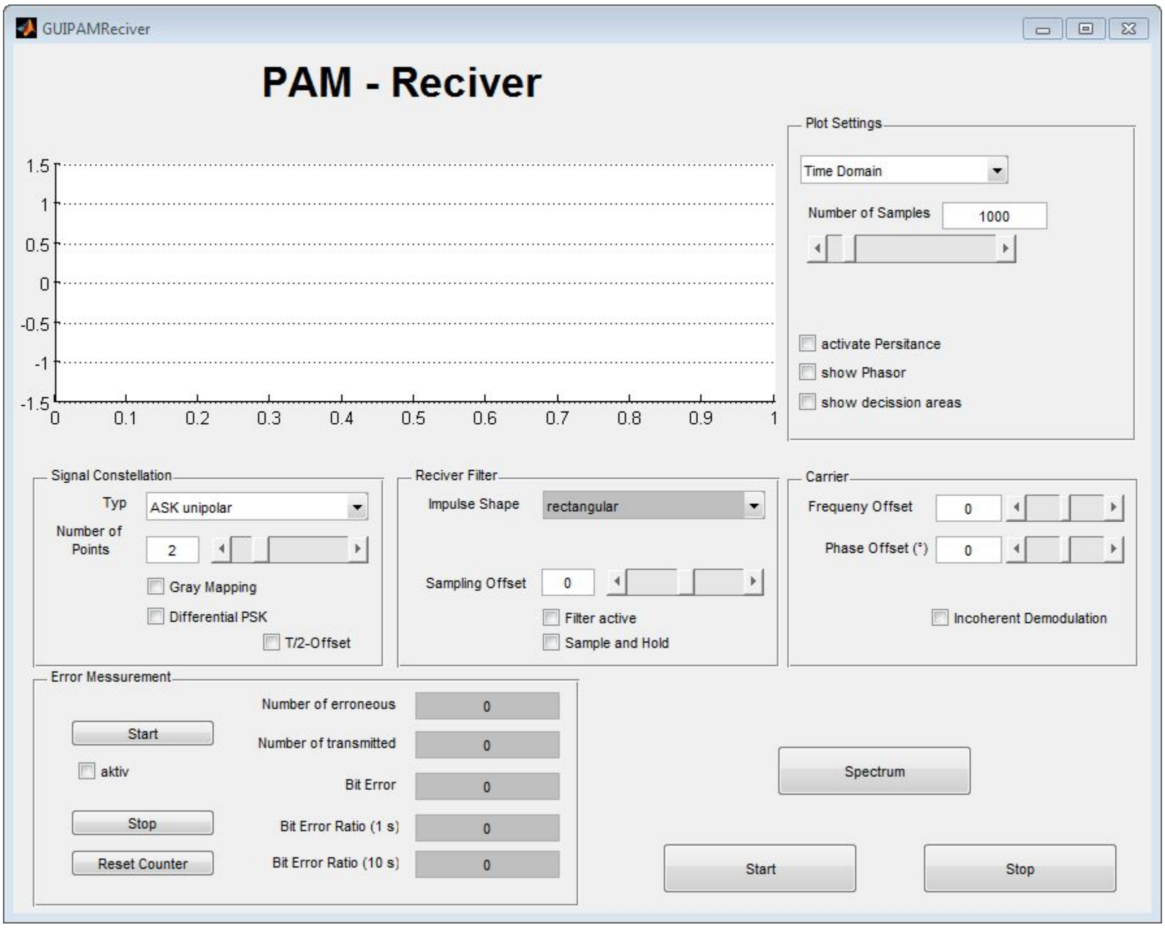


Figure 1.3: MATLAB GUI of PAM receiver.

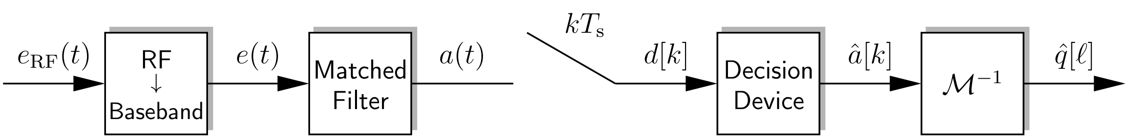


Figure 1.4: Block diagram of receiver structure of a digital transmission system.

1.3.2.4 Decision Device

The time-discrete samples $d[k]$ are then fed into the decision device to obtain estimates $\hat{a}[k]$ on the initial coefficients $a_{m[k]}$.

1.3.2.5 Demapping

The last step is to retrieve the binary data stream from the estimates $\hat{a}[k]$. $\mathcal{M}^{-1}$ is the inverse operation to $\mathcal{M}$ and yields the estimated binary data symbols $\hat{q}[\ell]$.

1.3.2.6 Disturbed Received Signals

At the receiver, we can model the effects of several problems which are likely to occur on the channel. The impact of frequency offsets, phase offset, and mismatched sampling instances can be tested on the received signal.

1.3.2.7 Error Counters

In order to assess the entire transmission scheme consisting of transmitter, channel, and receiver, the bit error introduced on the channel or by the effects mentioned above can be counted. The GUI offers an overall error counter and moving average values, resulting from window sizes of 1 s and 10 s. For reliable BER diagrams, the latter is preferably used.

Readings for Lab 1

[Hay00] Simon Haykin, *Communication systems*, 4th ed., John Wiley & Sons, New York, NY, USA, 2000.

[Hub11a] Johannes B. Huber, *Digital Communications*, Lecture Notes, Erlangen, October 2011.

[Hub11b] ______, *Nachrichtentechnische Systeme*, Vorlesungsskript, Erlangen, Oktober 2011.

[Pro00] John G. Proakis, *Digital communications.*, 4th ed., McGraw-Hill, New York, NY, USA, 2000.

1.4 Lab Exercises

1.4.1 Signal Generation at the Transmitter

First, we concentrate on the generation of the pulse-amplitude-modulated signal at the transmitter side. This is done in the equivalent complex baseband, i.e., the radio-frequency signal is not considered in these first steps.

In order to illustrate and analyze the properties of the signal constellations, a "rectangular pulse shape" has to be selected in the respective drop-down menu.

1.4.1.1 Signal Constellations (at the Transmitter)

Homework H-1.1

Calculate (by hand and analytically) the *peak-to-average power ratio (PAPR)* of the equivalent complex baseband signals for the following signal constellations: 2-ASK (unipolar/bipolar), 8-ASK (unipolar/bipolar), 16-QAM, and 8-PSK.

How are the PAPR and the crest factor related?

Note: The *peak-to-average power ratio* is defined as

$$\mathrm{PAPR} = \frac{\max_m\{P_m\}}{\mathrm{E}\{P_m\}},$$

P_m $(m = 1, \ldots, M)$ being the power of the m-th signal point.

Hint: Use the graphical representation of the signal constellations in the ECB to determine the PAPR. The squared distance of a signal point from the origin is proportional to its power P_m.

Lab Exercise L-1.1

Display the signal constellations from Tab. 1.1 on the oscilloscope (X-Y mode).

Lab Exercise L-1.2

Verify your analytical results from **H-1.1**, i.e., measure the PAPR of the physical signals of the signal constellations listed in **H-1.1**.

Note, you can measure voltages (peak, amplitudes, etc.) using the oscilloscope and RMS voltages using the Agilent multimeter. You cannot directly measure the power. In addition you can just measure these quantities in one of the two quadrature components.

Lab Exercise L-1.3

Compare the results for unipolar and bipolar ASK. How do you rate the two techniques in terms of the PAPR? Give some advantages for both variants of ASK.

1.4.1.2 Pulse Shaping

The properties of the transmit signal are–apart from the signal constellation–mainly determined by the employed basic pulse shape $g(t)$. In the following, you will study several basic pulse shapes with respect to their suitability to be used in digital transmission scheme. In particular, the spectra of the pulses shall be analyzed (baseband transmission).

Lab Exercise L-1.4

Study the time-domain inphase component of the equivalent complex baseband signal. First, use a unipolar 2-ASK signal constellation. Compare the signals resulting from a rectangular pulse shape and a cosine-roll-off pulse shape, respectively.

Repeat these steps for a bipolar 2-ASK, a 4-QAM, a bipolar 4-ASK, and a 16-QAM signal constellation.

Lab Exercise L-1.5

Compare the spectra of the ECB signals of a bipolar 2-ASK using a rectangular pulse shape and a cosine-roll-off pulse shape, respectively.

Homework H-1.2

How are the spectra of the radio-frequency and the baseband signal related?

Lab Exercise L-1.6

Verify your results from **H-1.2**.

1.4.1.3 The Radio-Frequency Signal

After the weighting of the basic pulse shape $g(t)$ with the channel coefficient $a_{m[k]}$, the baseband signal has to be transformed into a radio frequency signal centered around the carrier frequency f_c.

Lab Exercise L-1.7

Display and study the radio frequency signal of the following signal constellations in both, time domain and frequency domain: 2-ASK (unipolar/bipolar), 4-QAM, and 16-QAM.

First use a rectangular pulse shape and then a cosine-roll-off pulse shape.

Lab Exercise L-1.8

How does the size of the signal constellation M affect the spectra of the radio signals (Note, the data rate $1/T_\mathrm{b}$ is assumed to be constant)?

1.4.2 (Coherent) Receivers for Pulse Amplitude Modulation

In the following, we study the receiver used in PAM transmission. In particular, the filter at the receiver input is analyzed.

Homework H-1.3

How is the optimal filter at the receiver called? What is the optimization criterion in its derivation?

The combination of the basic pulse shape at the transmitter and the receive filter has to fulfill the so-called Nyquist criterion. Give the definitions of this criterion is both, time and frequency domain and sketch them for an exemplary pulse shape.

1.4.3 Transmission over the AWGN Channel

For the remaining experiments two of the lab systems have to be combined. One system acts as the transmitter, the other one acts as the receiver in a transmission scenario. An illustration for the practical implementation of this combination is given in Fig. 1.5.

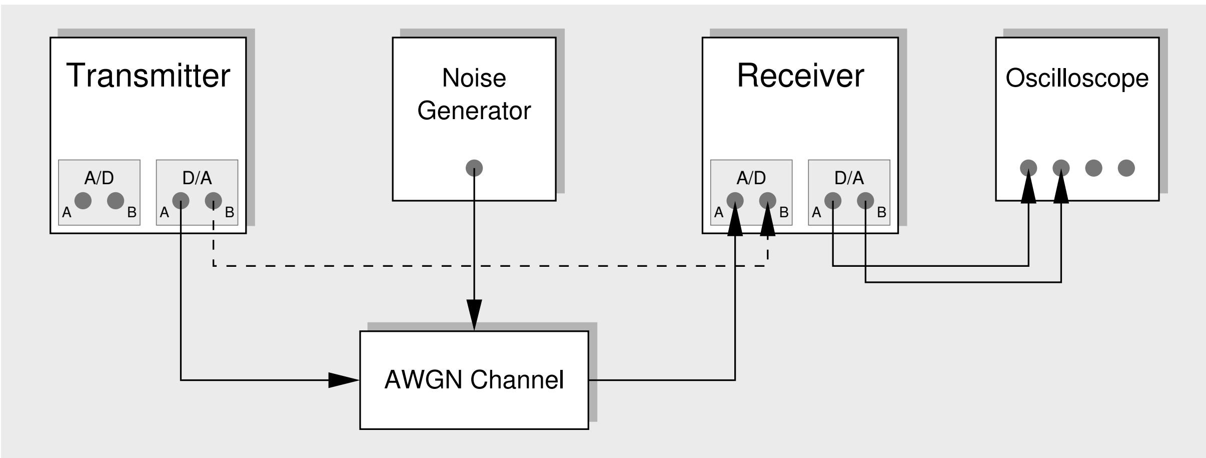


Figure 1.5: Transmission system using two of the lab computers.

1.4.3.1 Signal Constellations at the Receiver

In the following section, we finally want to study the signal constellations at the receiver side. This goal needs some preliminary steps which are described in the following.

Select a 4-QAM signal constellation and the $\sqrt{\cos}$ pulse with a roll-off factor of 0.94 at the transmitter. Study the inphase and the quadrature component of the demodulated (from RF) time-domain signal at the receiver.

Lab Exercise L-1.9

Describe the effect you can observe regarding the amplitude and the phase of the signal. How can you compensate for these effects?

Lab Exercise L-1.10

Activate the appropriate matched filter and study the resulting eye patterns. Identify the best time instance for sampling.

Lab Exercise L-1.11

Now activate the sample and hold check-box and describe the resulting signal. What happens, if you vary the sampling time instance?

Lab Exercise L-1.12

Use the oscilloscope (in X-Y mode) to display the phasors of the *sampled* received signal for 2-ASK (unipolar/bipolar), 4-QAM, 16-QAM, 4-PSK, and 8-PSK.

Lab Exercise L-1.13

How does the additive white Gaussian noise affect the signal constellations? Use a 4-QAM constellation and varying noise powers to demonstrate the effect.

Lab Exercise L-1.14

How do frequency offsets and phase offsets affect the signal constellation? Use a unipolar 2-ASK and a 4-PSK signal constellation and vary the respective parameters in the receiver GUI.

Lab Exercise L-1.15

How does a mismatched sampling affect the signal constellation at the receiver? Use again a 4-QAM signal constellation to study the effects.

1.4.3.2 Measuring the Bit Error Ratio

Lab Exercise L-1.16

Determine the bit error ratio over $E_\mathrm{b}/\mathcal{N}_0$ (range: 0 dB to 15 dB in steps of 2 dB) for the following modulations: 2-ASK unipolar, 2-ASK bipolar, 4-PSK, 8-PSK, 16-QAM. Use the diagram on page 26 to illustrate your results in a single plot.

Lab Exercise L-1.17

Compare the modulation schemes given in **L-1.16** in terms of spectral efficiency and power efficiency for a bit error ratio of $\mathrm{BER} = 10^{-4}$). Use the diagram on page 27 for an illustration.

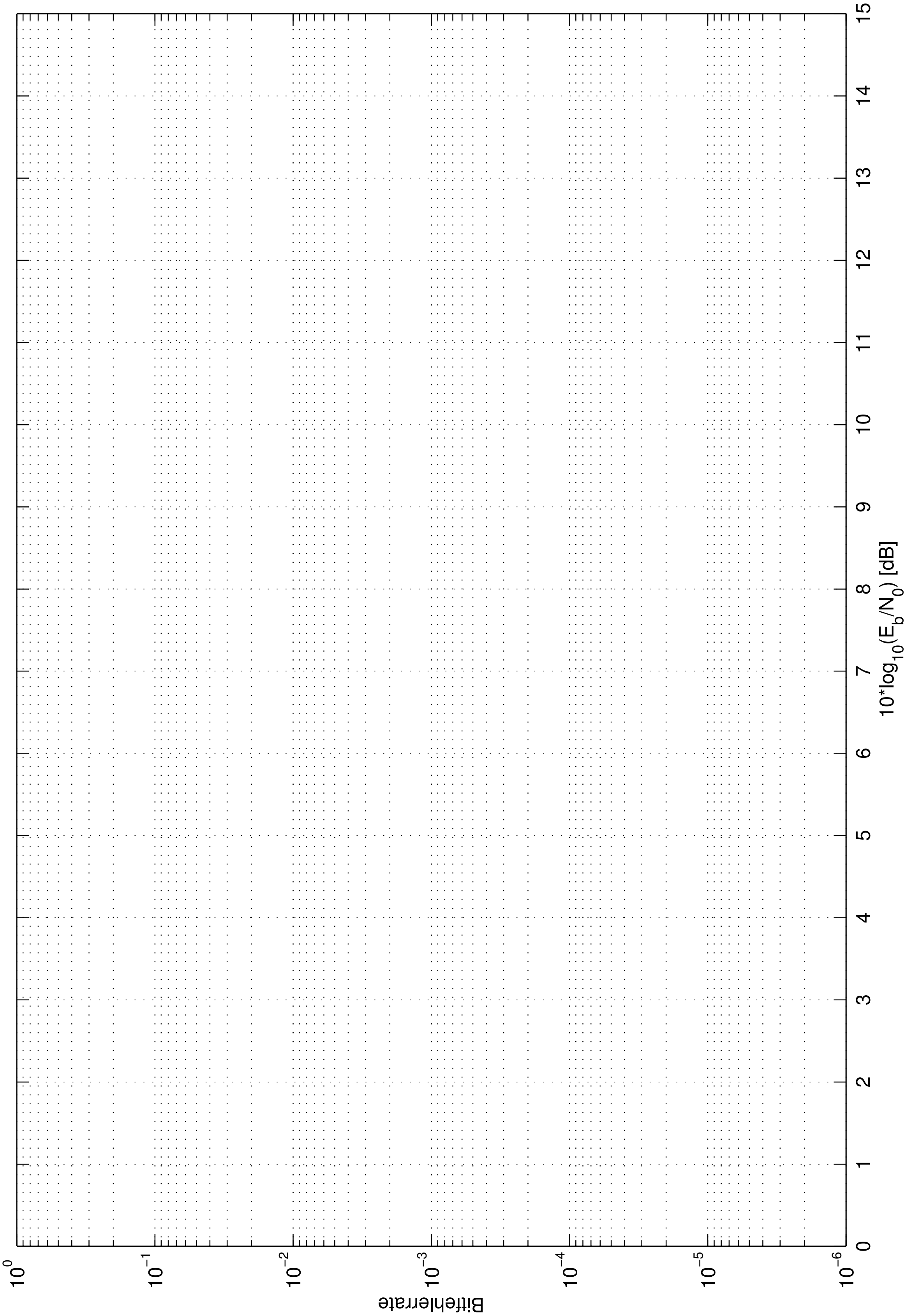


Figure 1.6: Diagram for **L-1.16**.

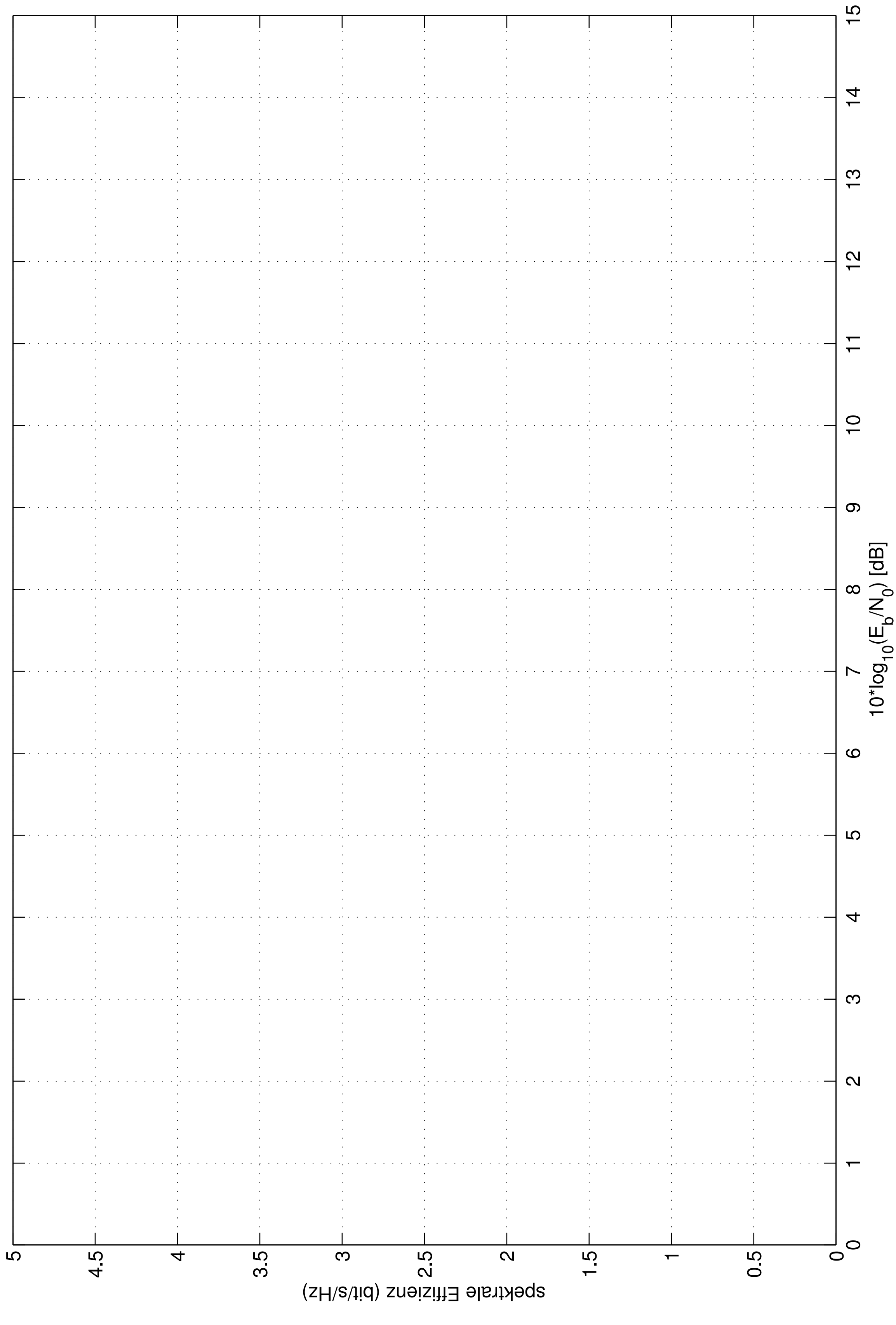


Figure 1.7: Diagram for **L-1.17**.

Project 2

Implementation of Transmitter and Receiver in MATLAB

2.1 Introduction, Background, and Motivation

In lab 1 a whole PAM system with different signal constellations has been analysed. The effects of some transmission disturbances on the received signal have been measured using oscilloscopes and multimeters and should be understood at this point. Furthermore, the advantages and disadvantages of each constallation should have become clear.

In this experiment, a complete digital communication system shall be implemented, where the whole system shall work on one computer using MATLAB without any additional hardware components. The script data '*./lab2/simulation.m*' will be the framework of this simulation, where all the following MATLAB functions will be called from. The missing MATLAB functions for transmitter, channel and receiver shall be implemented step by step by the student. To evaluate the performance of the simulation setup, in addition, a function calculating the bit error rate (BER) shall be implemented. At the end of this lab, the BER-$E_\mathrm{b}/\mathcal{N}_0$-curves if different transmission settings shall be visualized and compared to the well-known curves from the lectures.

2.2 Purpose

The aim of this experiment is to get more into detail of a digital communication system. By building the code of the different steps of transmitter, channel and receiver, the student will get a better insight into their functionalities.

2.3 Lab Environment

```
%% Simulation Parameters
    %PAM_type = '4QAM';
    PAM_type = '8ASKbipolar';

    GrayMappingOn = 0; % 0 = off, 1 = on
    EbN0_dB = 10;
    numberOfBits = 12e4;
    f_b = 1e3; % symbol/baud rate
    oversamplingFactor = 4; % should be at least 4
    f_c = 5e3; % carrier frequency; should be ca. 5 times f_s
    f_s = f_c*oversamplingFactor;

%% Creation of a random bit stream
    traBits = round(rand(1,numberOfBits));

%% Transmitter
    traSignal = transmitter(bitBits, PAM_type, ...
    GrayMappingOn, f_b, f_c, f_s);

%% Channel
    recSignal = channel(traSignal, EbN0_dB, M, f_s, f_b);

%% Receiver
    recBits = receiver(recSignal, PAM_type, ...
    GrayMappingOn, f_b, f_c, f_s);

%% Calculate BER
    BER = calculateBER(traBits, recBits)
```

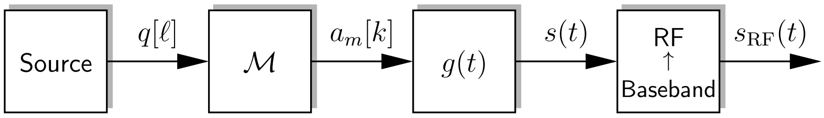


Figure 2.1: transmitter structure of a digital transmission system.

The code of the simulation script ('*./lab2/simulation.m*') is listed in this section. The code starts with setting the parameters. Especially the parameter *oversamplingFactor* is important. It will be described in subsection 2.3.1. Then, a random bit vector is created with the bit length set before. Subsequently the function calls of transmitter, channel and receiver are listed. The last part calculates the BER.

2.3.1 Oversampling factor

Transmitting data on a physical channel means to make use of D/A and A/D converter. So, the signals are not only time discrete but also time continuous. MATLAB works on vectors and matrices and cannot handle time continous signals. Even so, simulating a time continous signal is also possible in MATLAB: The continuous signal is simply approximated by working with an oversampled signal. By default, the parameter *oversamplingFactor* is set to 4.

Remark: In the following, the time continous signal is described as a function of the time continous value t, i.e. $s_{\mathrm{RF}}(t)$. In the context of MATLAB the signal should be a function of a time discrete value, i.e. $s_{\mathrm{RF}}[t]$. Using a *oversamplingFactor* of 4 means, that the system sampling rate f_s is 4· f_c. Thus, the values $s_{\mathrm{RF}}[t]$ can be seen as sampled values at sampling rate f_s.

2.3.2 Transmitter

The block diagram of the transmitter is the same as in experiment 1, see Fig. 2.1. Please have a closer look at the sections describing the transmitter units in experiment 1. In the following, *Bit Mapping* using Gray Mapping will be described in detail.

The job of *Bit Mapping* (block $\mathcal{M}$ in Fig. 2.1) is to map a M binary m-tuple to (complex) amplitude coefficients $a_m[k]$. This can be done in a natural way or in a more sophisticated

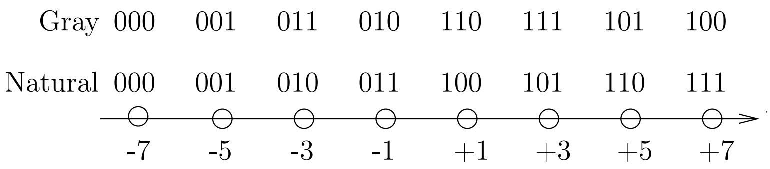


Figure 2.2: Constellation diagram for bipolar 8-ASK

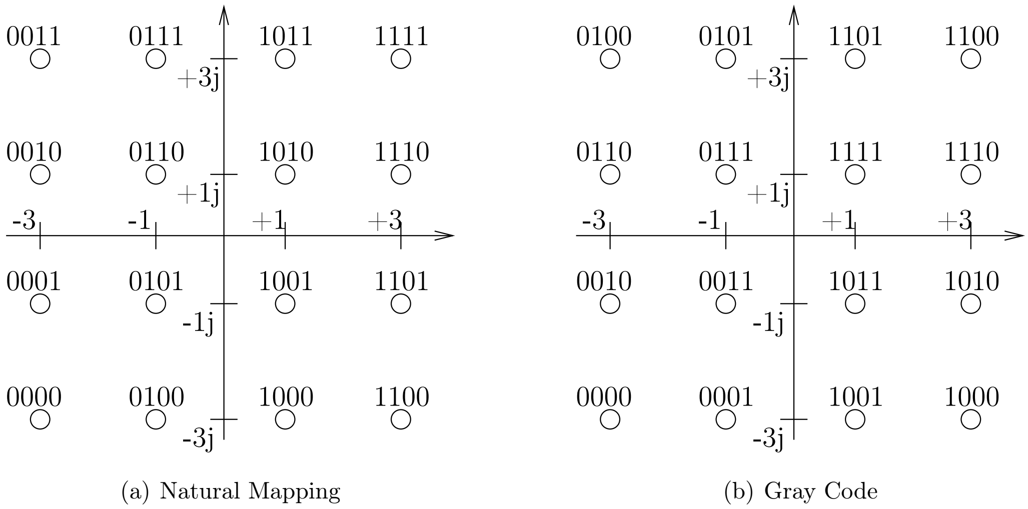


Figure 2.3: Constellation diagram for 16-QAM

way: the Gray Code. The resulting constellation diagrams are shown in Fig. 2.2 for (bipolar) 8-ASK and in Fig. 2.3 for 16-QAM.

If a symbol was detected at receiver-side which was not the transmitted one, the probability of detecting the neighbor of the transmitted symbol is much higher than detecting the others. By using the Natural Mapping neighboring signal points might differ in more than one bit value. This is not the case by using the Gray Code. There, detecting the direct neighbor yields to only one bit error in the whole m tuple.

2.3.3 Channel

The channel is kept simple in this experiment. The signal is just degraded by additive white Gaussian noise (AWGN) which has a constant power spectral density with the two-sided power spetral density denoted as $\mathcal{N}_0/2$. So, the block diagram is also simple, see Fig. 2.4. Adding the noise can be described with:

$$e_{\mathrm{RF}}(t) = s_{\mathrm{RF}}(t) + n(t) \tag{2.1}$$

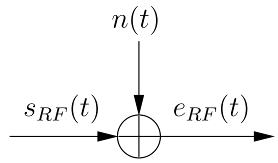


Figure 2.4: AWGN channel

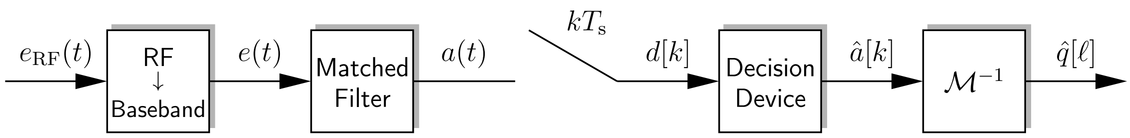


Figure 2.5: receiver structure of a digital transmission system.

2.3.4 Receiver

In a consistent way, the block diagram of the receiver is also the same as in experiment 1, see Fig. 2.5. Please have a closer look at the sections describing the receiver units in experiment 1. In this experiment, no Matched Filter is used, since a rectangular has been used to perform pulse shaping.

2.4 Lab Exercises

2.4.1 Transmitter

Lab Exercise L-2.1

Copy the file './lab2/simulation.m' into a subfolder 'working'.

Lab Exercise L-2.2

Create the MATLAB file *transmitter.m* in the working folder with the following head:

```
function traSignal = transmitter(bitvector, ...
    PAM_type, f_b, f_c, f_s)
```

This function will be filled in the following step by step.

2.4.1.1 Mapping

Homework H-2.1

The bit stream 10 11 00 01 00 11 is given. Map these bits onto complex PAM constellation symbols using bipolar 8-ASK and 16-QAM, first with Natural Mapping, then using Gray Code (see Fig. 2.2 and 2.3). Write the results in following table.

	Natural Mapping	Gray Code
bipolar 8-ASK		
16-QAM		

Lab Exercise L-2.3

Create the MATLAB file *mapBitsToSymbol.m* in the working folder with the following head:

```
function PAM_symbols = mapBitsToSymbols(bitvector, PAM_type)
```

Write the code which maps the bits inside the vector *bitvector* onto complex symbols according to input string *PAM_type* using Natural Mapping. Possible values for the input variable *PAM_type* can be '8ASKbipolar' and '4QAM'. Add the function call into the MATLAB file *transmitter.m*.

Lab Exercise L-2.4

Create the MATLAB file *mapBitsToSymbol_Gray.m* in the working folder with the following head:

```
function PAM_symbols = mapBitsToSymbols_Gray(bitvector, PAM_type)
```

Write the code equivalent to the function *mapBitsToSymbol.m*. It shall map the bits onto complex symbols, but with Gray Code. Add the call of this function to the function *transmitter*. During the later simulation run, take care, that one of these two mapping functions is commented out.

2.4.1.2 Pulse shaping

Lab Exercise L-2.5

The MATLAB file *pulseShape.m* is already in the working folder and ready to use. Add the following function call into the MATLAB file *transmitter.m*.

```
ecb_signal = pulseShape(PAM_symbols, f_b, f_s)
```

2.4.1.3 Modulation

Homework H-2.2

Write down the block diagram of the modulation of the complex PAM symbols to get the high frequency signal with carrier frequency f_c (last block in Fig. 2.1). A

simple filter with an rectangular impulse answer is used for pulse shaping. Complete the following equation:

$$s_{\mathrm{RF}}(t) =$$

Homework H-2.3

Given the following ECB signal:

$$s(t) = \begin{cases} +3 - 5i & \text{for } 0 \leq t < T \\ -1 - 1i & \text{for } T \leq t < 2T \\ +7 + 3i & \text{for } 2T \leq t < 3T \\ 0 & \text{else} \end{cases}$$

Calculate the HF signal after modulation for $0 \leq t < 3T$

Lab Exercise L-2.6

Create the MATLAB file *modulate.m* in the working folder with the following head:

```
function hf_signal = modulate(ecb_signal, f_c, f_s)
```

Write the code which takes the ecb signal and modulates it onto the carrier frequency f_c using sine and cosine. Add the function call into the MATLAB file *transmitter.m*.

2.4.2 Channel

Homework H-2.4

Given a signal vector $\mathbf{x}$ in MATLAB, write a pseudo code calculating the mean power of this signal.

Homework H-2.5

Given the calculated mean power $\bar{S}$ of the transmit signal $s_{\mathrm{RF}}(t)$, the symbol rate f_c and the modulation constellation size M. How can the energy per information bit E_{b} be calculated?

Homework H-2.6

A MATLAB Gaussian noise vector shall be added to the analogue transmit signal given these the two-sided power spectral density $\mathcal{N}_0/2$ and the system sampling rate f_s. How is the mean power of this noise vector calculated given these parameters

---

Lab Exercise L-2.7

Create the MATLAB file *channel.m* in the working folder with the following head

```
function recSignal = channel(traSignal, EbN0_dB, M, f_s, f_b)
```

Write the code which adds the Gaussian noise given the parameters of the function head.

2.4.3 Receiver

Lab Exercise L-2.8

Create the MATLAB file *receiver.m* in the working folder with the following head:

```
function recBits = receiver(recSignal, PAM_type, ...
f_b, f_c, f_s)
```

This function will be filled step by step. It is called in the MATLAB script *simulation.m*.

2.4.3.1 Demodulation

Homework H-2.7

Write down the block diagram of the demodulation of the received signal to get the complex baseband signal $e(t)$ (first block in Fig. 2.5).

Lab Exercise L-2.9

Create the MATLAB file *demodulate.m* in the working folder with the following head:

```
function demodSignal = demodulate(recSignal, f_c, f_s)
```

Write the code which takes the hf signal and demodulates it into ecb domain. Take care that both signal parts (real and imaginary part) are taken into account. Add the call of this function to the function *receiver*.

2.4.3.2 Matched Filter / Downsampling

Lab Exercise L-2.10

The MATLAB files *MatchedFilter.m* and *Downsample.m* are already in the working folder and ready to use. So, there is no need to take care of correct synchronisation. Add the following function call into the MATLAB file *receiver.m*.

```
filtered_signal = MatchedFilter(ecb_signal, f_b, f_s);
PAM_symbols = Downsample(filtered_signal, f_s/f_b);
```

2.4.3.3 Decision Device / Demapping

Homework H-2.8

The received PAM symbols are degraded by noise. What are the decision boundaries for 16-QAM in the AWGN case? Fill them into Fig. 2.3 using dotted lines.

In the following, the transmitted bit stream is 00 01 11 10 10 11. Consequently, the transmitted 16-QAM symbol sequence using Natural Mapping is $\{-3-1\text{j}; 3+1\text{j}; 1+3\text{j}\}$. The transmitted 16-QAM symbol sequence with Gray Code is $\{-1-3\text{j}; 3+1\text{j}; 1-1\text{j}\}$.

At the receiver-side, the received 16-QAM symbol sequence is degraded by additive white Gaussian noise. The complex noise part can be described by the sequence {+1.5-0.7j; 0.2-1.1j; -1.1+0.5j} which is added to the transmitted symbols.

Homework H-2.9

Calculate the estimated 16-QAM symbols $\hat{a}[k]$ for both received signal sequences!

Homework H-2.10

Demap both sequences to bit streams using the decision boundaries defined in the homework and calculate the bit error rate! What is better in this case: Natural Mapping or Gray Code? Why is it better?

Lab Exercise L-2.11

Create the Matlab file *mapSymbolsToBits.m* in the working folder with the following head:

```
function bitvector = mapSymbolsToBits(PAM_symbols, PAM_type)
```

Write the code which first maps the estimated PAM points $\hat{a}[k]$ to the nearest possibly transmitted point and second map these values to bits again according to input string *PAM_type* and with Natural Mapping. Possible values for the PAM_type can be '4QAM' and '8ASKbipolar'. Add the call of this function to the function *receiver*.

Lab Exercise L-2.12

Create the Matlab file *mapSymbolsToBits_Gray.m* in the working folder with the following head:

```
function bitvector = mapSymbolsToBits_Gray(PAM_symbols, PAM_type)
```

Write the code equivalent to the demapping without Gray Code in the function *receiver*. It shall map the received PAM points to the nearest possibly transmitted point and map these values to bits again according to input string *PAM_type*. Add the call of this function to the function *receiver*.

2.4.4 BER calculation

Homework H-2.11

Having the transmitted bit stream and the received bit stream. How can the bit error rate be calculated?

Homework H-2.12

How many bits have to be simulated to calculate reliable value for a bit error rate of less than 10^{-6}?

Lab Exercise L-2.13

Create the Matlab file *calculateBER.m* in the working folder with the following head:

```
function BER = calculateBER(traBits, recBits)
```

Write the code which calculates the bit error of the communication system.

Lab Exercise L-2.14

Run the simulation script with different parameters such as number of bits, PAM type, etc. and check if everything works as you expect.

Up to now, the simulation script is ready to run one simulation. However, to compare different settings, it is common to simulate with different $E_{\mathrm{b}}/\mathcal{N}_0$ values and save the BER in a vector. Then, BER / $E_{\mathrm{b}}/\mathcal{N}_0$ curves can be plotted for the selected settings. Therefore, the simulation script has to be adapted to the form on the following page.

```
%% LOOP PARAMETERS
EbN0_dB_Vector = 0:1:15;
BER_Vector = zeros(1,length(EbN0_dB_Vector));

%% Simulation Parameters
    %PAM_type = '4QAM';
    PAM_type = '8ASKbipolar';

    GrayMappingOn = 0; % 0 = off, 1 = on
    %EbN0_dB = 10;    % this is now commented out
    numberOfBits = 12e4;
    f_b = 1e3; % symbol/baud rate
    oversamplingFactor = 4; % should be at least 4
    f_c = 5e3; % carrier frequency; should be ca. 5 times f_s
    f_s = f_c*oversamplingFactor;

%% Creation of a random bit stream
    traBits = round(rand(1,numberOfBits));

%% LOOP START
curLoop = 0;
for EbN0_dB = EbN0_dB_Vector
    curLoop = curLoop + 1;

%% Transmitter
    traSignal = transmitter(traBits, PAM_type, ...
    GrayMappingOn, f_b, f_c, f_s);

%% Channel
    recSignal = channel(traSignal, EbN0_dB, M, f_s, f_b);

%% Receiver
    recBits = receiver(recSignal, PAM_type, ...
    GrayMappingOn, f_b, f_c, f_s);

%% Calculate BER
    BER = calculateBER(traBits, recBits)

%% LOOP END
    BER_Vector(curLoop) = BER;
end
```

Lab Exercise L-2.15

Add the simulation loop lines in your simulation script and run the simulation for the two PAM types and the configuration with and without Gray mapping. Do not forget to change the name the specific BER vector at the end of the code for the plots.

Lab Exercise L-2.16

Plot the four BER curves in Matlab using the functions 'figure', 'semilogy(x,y)' and 'hold on'. You can label the figure with the function xlabel, ylabel, legend, title. Discuss the curves in your group and then with your supervisor.

Lab Exercise L-2.17

Fill the values into the diagram on page 56 and compare your curves with the curves from theory on page 52.

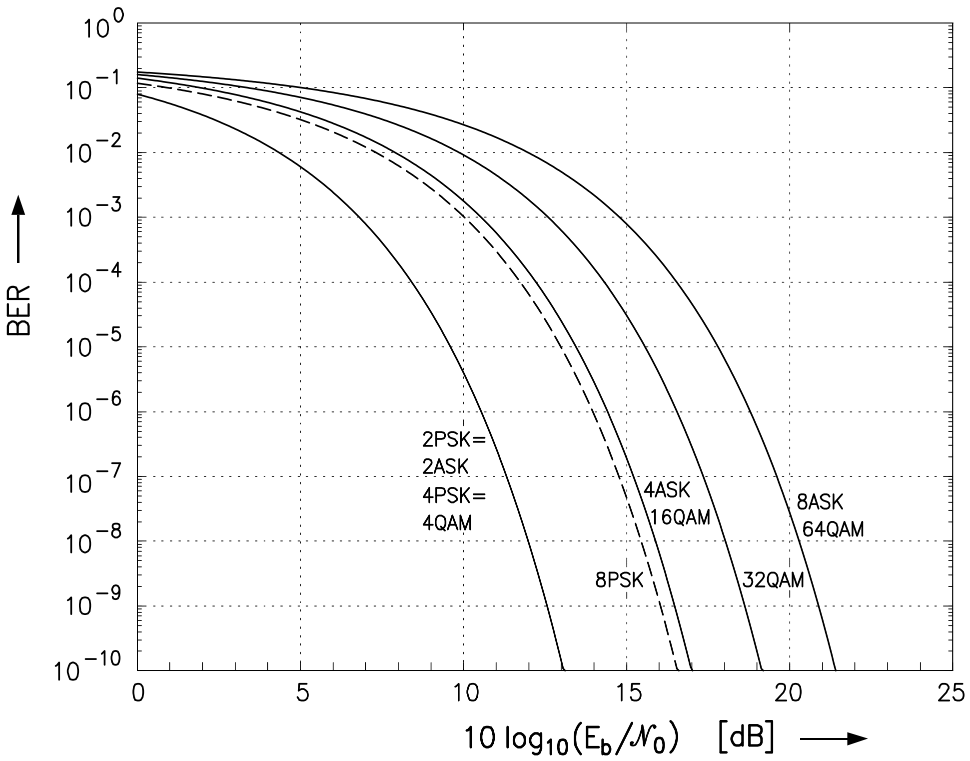
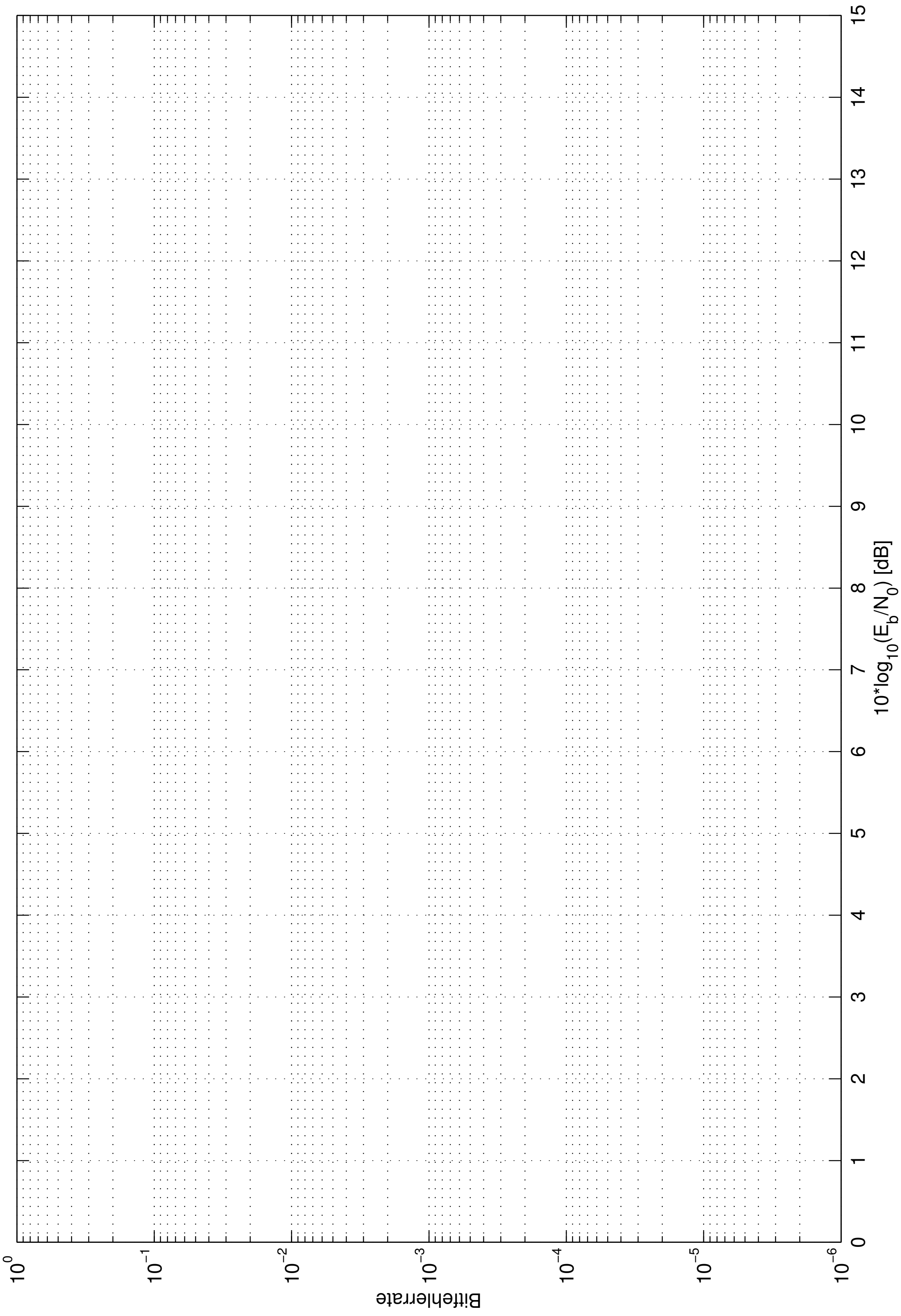


Figure 2.6: Bit error rates (BER) of different PAM schemes.

.


0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
10*log10(Eb/N0) [dB]
10^0
10^−1
10^−2
10^−3
10^−4
10^−5
10^−6
Bitfehlerrate

Project 3

Variants of PAM-Transmission Schemes

3.1 Introduction, Background, and Motivation

The third experiment will be about variants of PAM transmission. You will encounter different well-known and widely applied schemes of PAM-transmission and explore and compare their properties.

In the first experiment you already studied the building blocks of a standard PAM-transmission scheme and their function and properties. You further studied the building blocks in the second experiment, when you implemented them yourselves. In this experiment you will find the same building blocks, but of course some modifications will be done to reach certain goals. These goals may be to optimize spectral efficency, reduce complexity, or other desirable properties. You will design your own basic pulse shapes and investigate their impact on the transmission system. You will also learn about some well-known variants of PAM transmission schemes and their connection and motivations.

3.2 Purpose

This experiment is supposed to provide an overview of variants of PAM and show the plurality of it. Furthermore the connections between the variants shall be explored.

3.3 Lab Environment

The lab environment is the same as in experiment 1.

3.4 Lab Exercises

Figure 3.1 depicts a standard PAM transmission scheme, which is the foundation of all variants found in this experiment. One fundamental building block is the basic pulse shape and its influence is oftentimes underestimated.

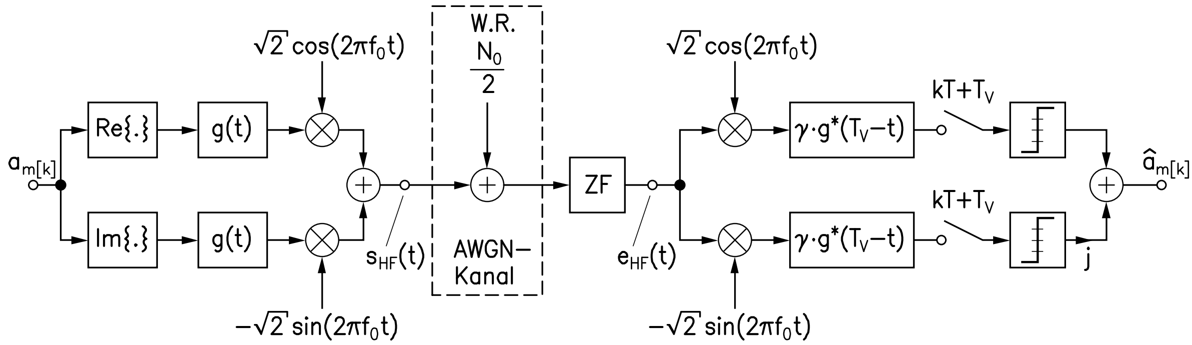


Figure 3.1: Standard PAM transmission scheme

3.4.1 Basic Pulse Shape

To understand the influence of the basic pulse shape we want to examine the impact of different basic pulse shapes on the standard PAM system. In experiment 2 the basic pulse shape was always set to be rectangular.

Homework H-3.1

What property do basic pulse shapes usually possess if intersymbol interference-free transmission is desired? State the name and explain what this property means.

Homework H-3.2

Sketch the following basic pulse shapes in time domain: Rectangular, cosine with roll-off slope and one pulse of your choice, which does *not* fulfill the property from homework 1. How does the cosine pulse change for different roll-off slope factors α. For the pulse shape of your choice also provide a formula.

Lab Exercise L-3.1

In the MATLAB -file “Eigener Impuls” you can define your own impulse. “T” stands for the symbol interval and “t” for the time variable. You can also use variable “Alpha” if you want to define a roll-off slope. Please make sure your impulse is defined in the intervall from “0” to “T”. Compare the impulse shape the transmitter shows to your sketch you prepared.

Lab Exercise L-3.2

Compare the spectra at the receiver, while transmitting 8ASK (bipolar) signals using different pulse shapes and roll-off slope factors $\alpha = 0, 0.3, 0.6, 1$. (Use the built in MATLAB function at the receiver). Measure the bandwidth of the signals.

Lab Exercise L-3.3

Compare the eye-patterns at the receiver, while transmitting 8ASK (bipolar) signals using different pulse shapes and different roll-off slope factors α. (Use the built in MATLAB function at the receiver)

Lab Exercise L-3.4

Determine the bit error ratio versus $E_\mathrm{b}/\mathcal{N}_0$ (range: 0 dB to 15 dB in steps of 2 dB) for 8PSK using the following basic pulse shapes: rectangular, raised cosine with $\alpha = 0.3$ and your own one. Use the diagram on page 64 to illustrate your results in a single plot.

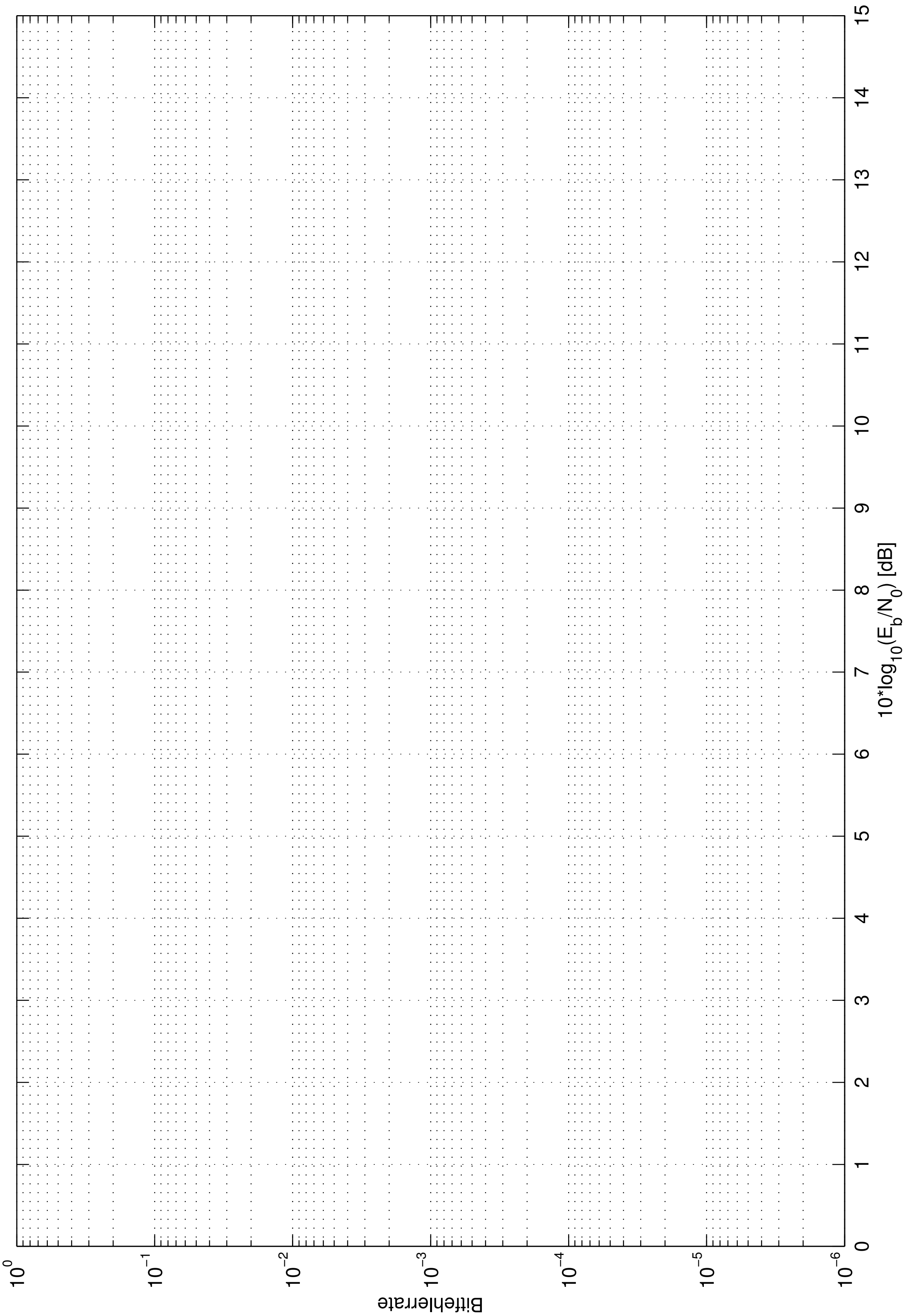


Figure 3.2: Diagram for **L-3.4**.

3.4.2 Offset-QAM

In order to minimize variations of the envelope of the transmit signal Offset-QAM has been invented. In order to achieve this property the quadrature component of the signal is delayed by half a symbol duration (T/2) as depicted in Figure 3.3.

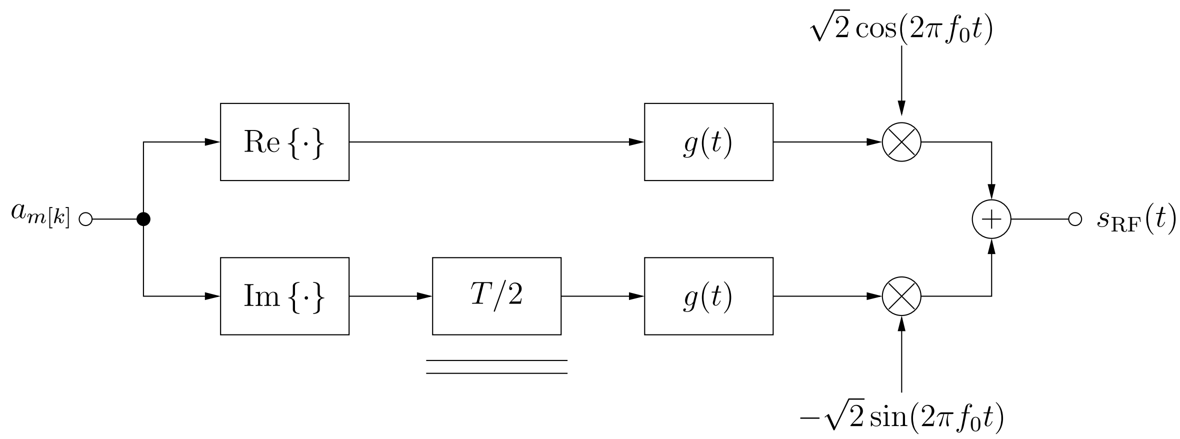


Figure 3.3: Offset-QAM transmission scheme

Homework H-3.3

Why is it useful to minimize variations of the envelope of the transmit signal?

Lab Exercise L-3.5

Compare the phasors of a 4QAM transmission scheme with and without offset (time-delay) at the transmitter as well as at the receiver. At the transmitter you can check "Offset" on and off. At the receiver use MATLAB to display the phasor.

Lab Exercise L-3.6

How do different basic pulse shapes with different parameters α influence the phasor?

Lab Exercise L-3.7

Using the possibility of defining your own impulse define a basic pulse shape of positive sinusodial shape, more precisely a sinus from "0" to "π" fitted into the intervall from "0" to "T". Transmit Offset-4QAM using this basic pulse shape and display the phasor of the signal. What shape does it have?

Lab Exercise L-3.8

What crest factor does the transmit signal possess?

Lab Exercise L-3.9

What is the name of this well-known transmission scheme you just put together?

3.4.3 Gaussian Minimum Shift-Keying

For the following exercises use the basic pulse shape "MSK". As MSK does possess a constant envelope of the transmit signal, all the transmitted information is contained in the phase of the signal and not in the amplitude. Therefore MSK can also be interpreted as binary Frequency-Shift-Keying (FSK).

Homework H-3.4

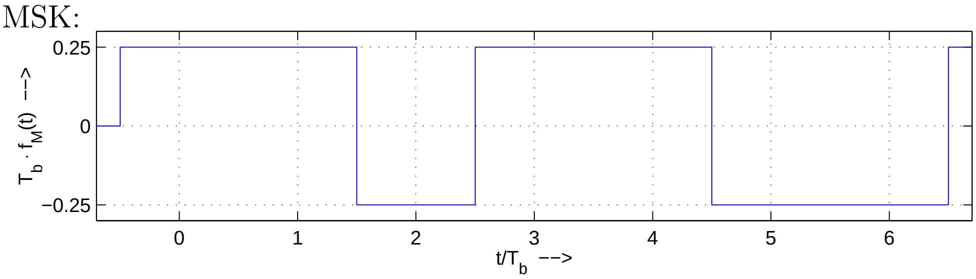
In order to revisit this equivalence between MSK and binary FSK please fill out the solution sheet for the given sequence. (You will also find an example in the DiCo script)

Lab Exercise L-3.10

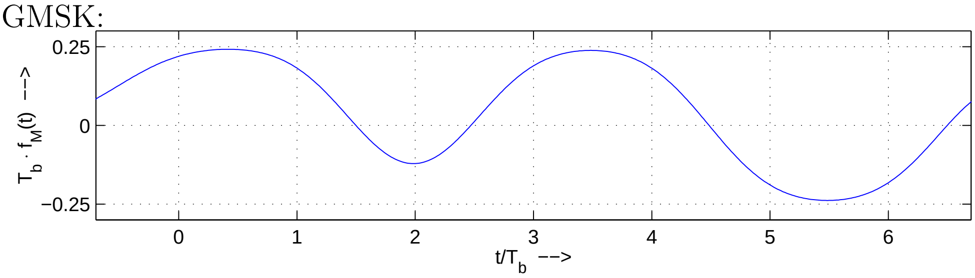
At the receiver compare the spectrum of a standard 4QAM signal and an MSK signal.

As you may have realized the changes of frequency in MSK/binary FSK are very abrupt. A softer slope in the frequency over time would result in a smaller spectrum. In order to reach that goal a filter can be applied, which softens the abrupt rectangular changes in frequency into Gaussian sloped ones. The name of this scheme is "Gaussian Minimum Shift-Keying" (GMSK).

- MSK:



- GMSK:



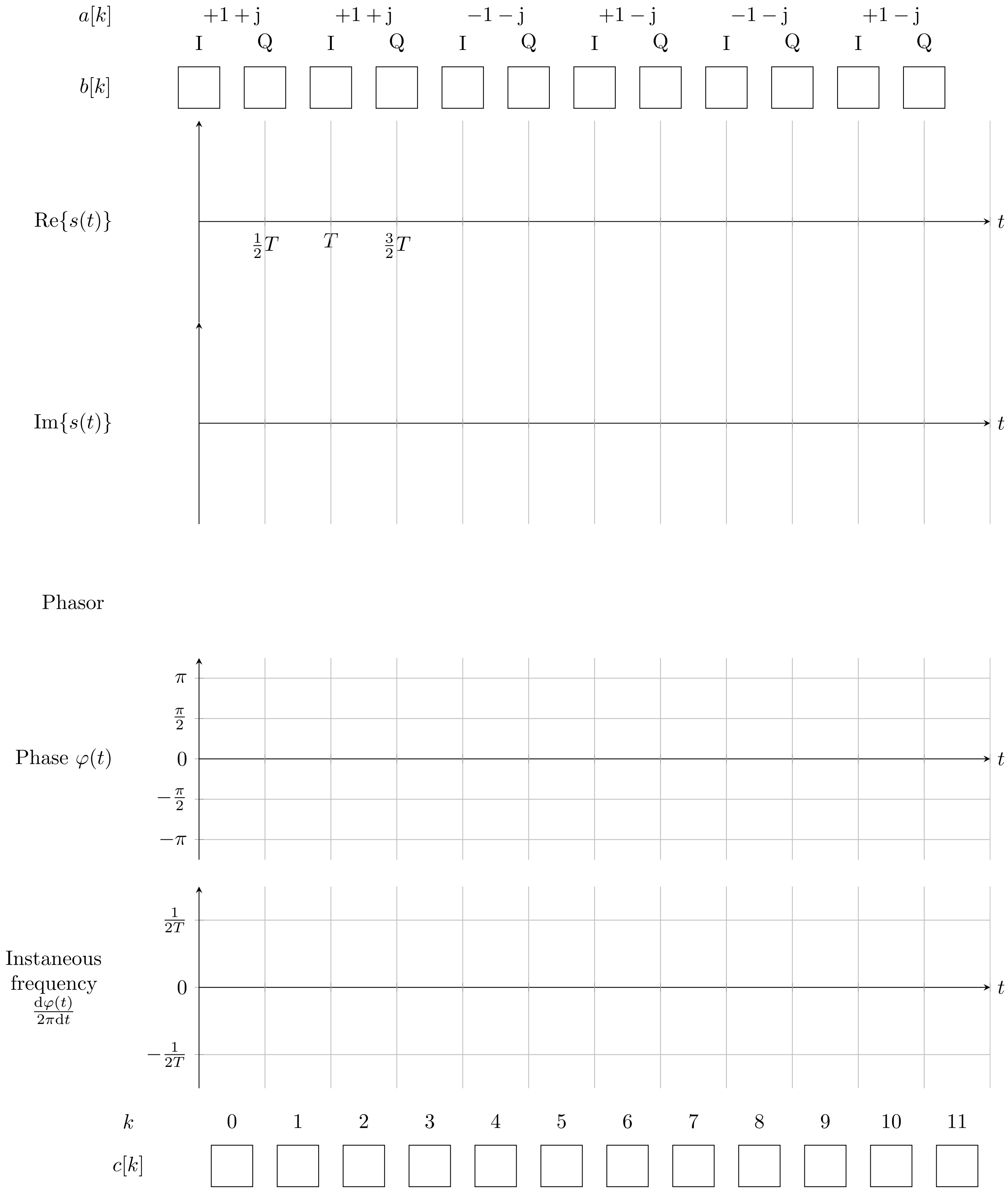
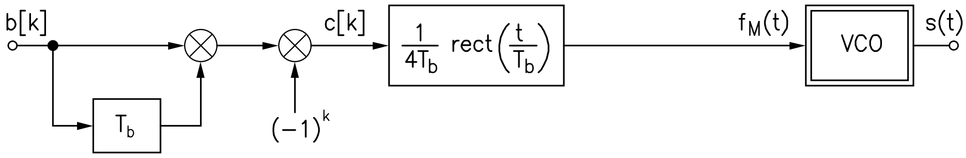


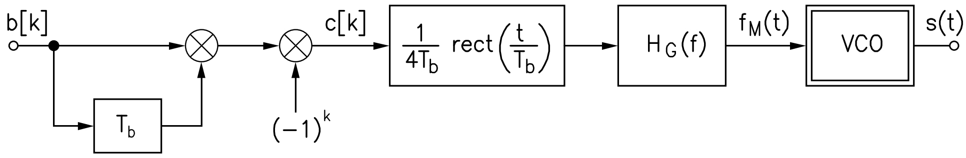
Figure 3.4: Solution sheet for exercise 4

Modulation:

MSK:



GMSK:



with

$$\boxed{H_{\mathrm{G}}(f) = \mathrm{e}^{-(f/B_{\mathrm{b}})^2 \cdot \ln 2/2}}$$

GSM: $B_{\mathrm{b}} \cdot T_{\mathrm{b}} = 0,3$

Parameter: $B_{\mathrm{b}} \cdot T_{\mathrm{b}}$

Lab Exercise L-3.11

Use the basic pulse shape GMSK. Display the phasor of the transmit signal and compare it to the phasor of MSK.

Lab Exercise L-3.12

What influence does the parameter $B_{\mathrm{b}} \cdot T_{\mathrm{b}}$ have on the bandwidth and the phasor of the received signal? How does the eye-pattern change?

Lab Exercise L-3.13

Compare standard 4QAM, MSK and GMSK in terms of spectral efficiency as well as the opening of the eyepatterns.

3.4.4 "Carrierless" Amplitude and Phase Modulation

Instead of first filtering the signal with the basic pulse shape $g(t)$ and then mixing it into the RF-band using carrier frequency f_0 one can also combine these two blocks by not defining $G(f)$ in the baseband but shifting it to a higher frequency (=carrier frequency). If we wanted to use in-phase and quadrature components we would have to use two orthogonal transmit pulses, one for each component. Therefore we will just use bipolar ASK (onedimensional).

Lab Exercise L-3.14

> Define a basic pulse shape for transmission in the CAP scheme. Make sure you check the "CAP" check-box to tell the transmitter you are now using CAP.

Lab Exercise L-3.15

> How would the orthogonal basic pulse shape for the quadrature component look like?

Lab Exercise L-3.16

> Compare the spectra, eye-patterns and transmission errors for 8ASK using CAP and standard transmission.

Project 4

OFDM

4.1 Introduction, Background, and Motivation

In many (digital) transmission schemes the channel cannot be modeled by a simple AWGN channel but the channel is also dispersive. This means that the transfer function of the channel is not only a scalar as for the AWGN channel, but depends on the frequency. The channel introduced intersymbol interference. Orthogonal frequency-division multiplexing (OFDM) is a well-known and popular technique to cope with these frequency-selective channels.

4.1.1 Orthogonal Frequency-Division Multiplexing

For this lab, we follow the description and definition of OFDM as given in [Hub11]. The basic idea of OFDM can be visualized by splitting the frequency-selective transfer function of the dispersive channel into a number of quasi-constant subbands. These subchannels are then non-dispersive and can be used in parallel. They do not introduce any intersymbol interference.

The basic pulse shapes used for the parallel transmission over these subchannels have to be orthogonal in the frequency domain. At the receiver a bank of matched filters prevents interference between the individual subchannels. This general approach to OFDM is is shown in Fig. 4.1.

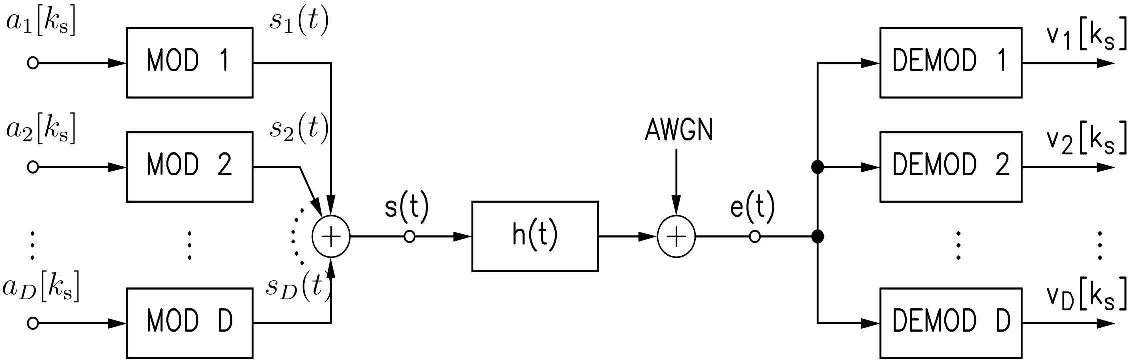


Figure 4.1: Basic principle of OFDM

For the practical implementation of an OFDM system, a different approach is preferred over the individual parallel modulators. Instead, the inverse discrete Fourier transform (IDFT) is use to transform blocks of D symbols from the frequency domain to the time domain. These symbols are then modulated by a single basic pulse shape $g(t)$. At the receiver the matched filter wrt. $g(t)$ is applied an then the block of D receive symbols is transformed again into the frequency domain using a DFT.

The individual blocks generated at the transmitter are separated by a cyclic prefix to avoid interblock interference. In Fig. 4.2 this block-based description of OFDM is illustrated. Finally, the OFDM transmission over independent parallel subchannels can be described by just using individual scaling factors per channel.

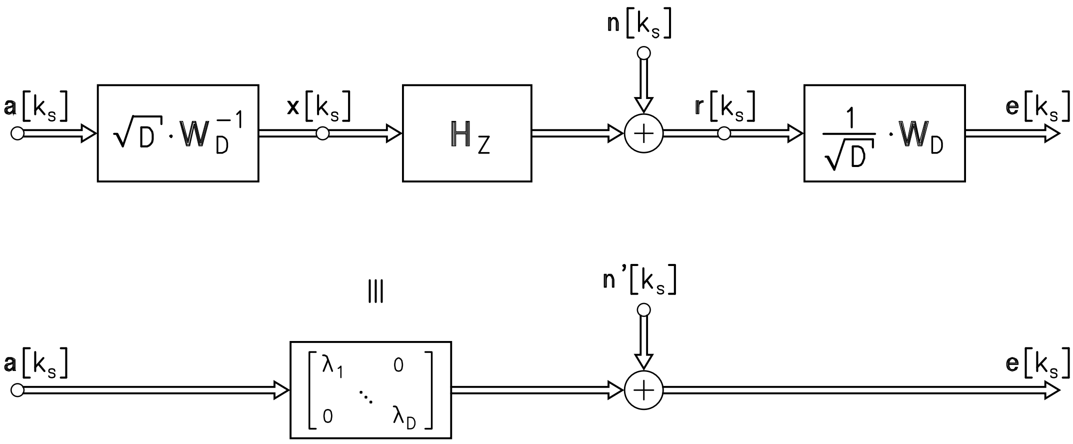


Figure 4.2: Transmission over independent subchannels

4.1.2 Bit Loading

The varying scaling factors λ_i per subchannel lead lead to varying signal-to-noise ratios over the subchannels. If these different SNRs are known at the transmitter, so-called bit loading can be used. That is, the data rate per subchannel is adapted to the subchannel conditions.

One of the potential approaches to adapt the individual data rates is “rate allocation according to the channel capacity“. $R = \sum_{i=1}^{D} \mathrm{ld}(1 + \frac{1}{N_i * \gamma})$ A detailed explanation of this bit loading algorithm will be given in class.

4.2 Purpose

The aim of this experiment is to demonstrate the OFDM and a simple bit loading algorithm. You should understand why OFDM is widely used in dispersive channels. In addition you should understand why bit loading can useful in OFDM systems.

4.3 Lab Environment

The lab environment is identical to that of Chapter 2.

Readings for Lab 4

[Hub11] Johannes B. Huber, *Digital Communications*, Lecture Notes, Erlangen, October 2011.

4.4 Lab Exercises

In this Lab we will extend the simulation script from Chapter 2. The aim is a simulation of an OFDM system.

The first change in the transmission scheme is the dispersive channel.

Lab Exercise L-4.1

There is a function `dispersive_channel()` which models the dispersive channel. Add the frequency-selective channel to your transmission chain before the addition of the white noise. How does the bit error ratio change?

Homework H-4.1

How can you observe the influence length of the dispersive channel.

In this Lab we use an all-digital representation of the transmission scenario. Thus, the influence length of the channel can not be measured in seconds, but in samples only.

Lab Exercise L-4.2

Determine the length of the impulse response (in samples) of the given dispersive channel.

4.4.1 OFDM Transmitter

We now want to transmit the data using an OFDM system.

Homework H-4.2

Draw the block Diagram of a general OFDM transmitter.

Homework H-4.3

The OFDM system shall be designed to operate in the base band (like in DSL)? Are there any constraints on the signal?

In order to construct the OFDM transmitter, it is reasonable to split it into smaller blocks.

Lab Exercise L-4.3

Save the file `old transmitter.m` as `ofdm_transmitter.m`. The bitstream has to be split in $D = 16$ substreams. Use the given function `map_bits_qam(bitvector, M)` for the mapping for each substream on symbols.

Lab Exercise L-4.4

Use the inverse discrete Fourier transform to generate the OFDM signal out of the symbols for each subchannel.

Keep in mind that the system should transmit in the base band.

Lab Exercise L-4.5

Proof that your signal is real valued.

Lab Exercise L-4.6

Add the cyclic prefix. The length of the cyclic prefix should be optimized for the given dispersive channel.

Next, we analyze the transmit signal.

Lab Exercise L-4.7

In order to analyze the OFDM transmit signal, we generate a signal resulting from 1,200,000 Bits. Use the function `write_to_card(Block)` to generate an output of the D/A converter. Now analyze the OFDM signal in time and frequency domain using the oscilloscope and spectrum analyser.

Lab Exercise L-4.8

Study the distorted signal after the dispersive channel and the AWGN with the oscilloscope and spectrum analyser.

4.4.2 OFDM Receiver

Homework H-4.4

Draw the block diagram of a generic OFDM receiver.

Lab Exercise L-4.9

Save the file `old receiver.m` as `ofdm_receiver.m`.

Lab Exercise L-4.10

Remove the cyclic prefix from the signal.

Lab Exercise L-4.11

Use the Fourier transform to generate the OFDM signal out of the symbols for each subchannel. Keep in mind that the system should transmit in the base band.

Lab Exercise L-4.12

In order to equalize the effect of the scaling factors in the subchannels, we apply a so-called frequency domain equalization (FEQ). The FEQ is provided in the function `feq(Symbols)`.

Lab Exercise L-4.13

The bitstream has to be split up for the 16 individual subchannel. Use the given function `demap_bits_qam(Symbols, M)` for the mapping of individual stream.

After the transmission the BER can be calculated.

Lab Exercise L-4.14

Modify the function `calculateBER.m` to match the needs for your OFDM system and save it as the new file `calculateBER_OFDM.m`.

Homework H-4.5

The dispersive channel has an influence on the SNR. How can the SNR be computed at the receiver? How can it be calculated per subchannel?

Lab Exercise L-4.15

Write a function to compute the SNRper subchannel and plot the SNR.

Lab Exercise L-4.16

Plot the SNRper subchannel.

Now you know the SNR per subchannel. It is possible to use bit loading to achieve higher data rates if you use modulations of a higher order in subchannels with an high SNR.

4.4.3 Bit Loading

Lab Exercise L-4.17

Extend your OFDM transmitter and receiver to support bit loading, i.e., varying data rates per subchannel. Determine the data rates of the 16 subchannel by hand!

Lab Exercise L-4.18

Write a block of the loaded signal to the D/A converter and analyze the signal in time and frequency domain.

Project 5

Signal Space Representation

5.1 Introduction

This chapter will introduce the implementation of modulation schemes with signal elements. Here we will focus on memory-less modulation schemes such as Frequency Shift Keying (FSK). Those modulation schemes with inherent encoding, such as Continuous Phase Modulation (CPM) are not considered.

5.2 Purpose

In this exercise the student will extend his existing transmission scheme to use a set of signal elements. Thereby the implementation of other modulation schemes, such as Frequency Modulation (FSK) becomes easy. It will become obvious that a memory-less modulator is used to connect the signal numbers – an abstract mathematical description of information – with signals that can be transported over a physical channel. A matched filter (MF) receiver (also known as correlation or vector receiver) will demodulate the transmitted signal which is degraded by additive noise due to the AWGN channel. We will see that with the use of signal elements the set of modulator and MF receiver can easily be described by a single set of signal elements. We will also learn that with the use an orthogonalization algorithm the complexity of the receiver may be reduced.

5.3 Lab Environment

The lab environment is equal to those of the prior exercises. We will extend existing implementations to include the features of signal space representation. In order to later distinguish your implementations of different lab courses we will copy existing files instead of editing them.

5.4 Signal Space Representation

In general transmission schemes contain several blocks such as an encoder, a mapper, and a modulator. We will focus on uncoded transmission so we can neglect the encoder. Basically the modulator is memory-less and contains a set of – not necessarily orthogonal – signal elements (Fig. 5.1).

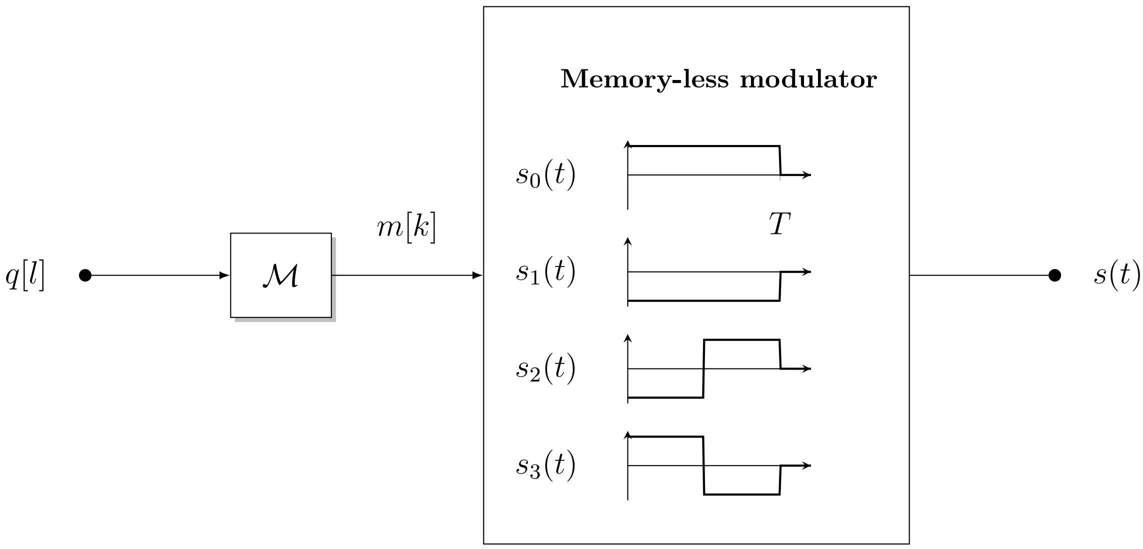


Figure 5.1: Block diagram of a transmitter. The binary data sequence $q[k]$ is mapped to signal numbers $m[k] \in \{0, 3\}$ which selects the signal elements.

Instead of weighting your basic pulse shape $g(t)$ with ± 1 as it is done for binary PAM transmission, we now have a signal space $\mathcal{S}$ with a set of signal elements of a duration T from which the corresponding signal element is selected for each time instance. The resulting transmit signal can be written as

$$s(t) = \sum_{k=-\infty}^{+\infty} s_{m[k]}(t - kT) \tag{5.1}$$

where $m \in \{0, \cdots, (M-1)\}$ describes the signal number at time instance k and $s_{m[k]}(t)$ is the corresponding signal element.

5.4.1 Orthogonality

An important condition is that signal elements $s_m(t)$ must fulfill the temporal orthogonality condition [Hub11]

$$\frac{1}{E_{i\ell}} \int\limits_{-\infty}^{+\infty} s_i(t+kT) \cdot s_\ell^*(t) \,\mathrm{d}t = \underline{\underline{\delta_{0k}}}\,, \quad \forall i, \ell \in \{0,\ 1, \ldots,\ (M-1)\}, \tag{5.2}$$

with $E_{i\ell}$ being the energy of the cross correlation of the signal elements $s_i(t)$ and $s_\ell(t)$ and δ_{ij} being the Kronecker-symbol:

$$\delta_{ij} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j. \end{cases} \tag{5.3}$$

In general the signal elements are not limited in time but we will focus on time-limited signals which are by definition temporally orthogonal.

Although this temporal orthogonality is fulfilled the signal elements might not be mutual orthogonal. If the double orthogonality condition is also fulfilled the signal elements are also mutually orthogonal! Throughout this lab course signal elements are described by $s_i(t)$ with $i \in \{0, \ldots, (M-1)\}$ unless they fulfill the double orthogonality. Else signal elements are described by $g_i(t)$ and are orthonormal basis vectors of the signal space, i.e.

$$\frac{1}{E_g} \int\limits_{-\infty}^{+\infty} g_l(t+kT) \cdot g_\ell^*(t) \,\mathrm{d}t = \underline{\underline{\delta_{l\ell} \cdot \delta_{0k}}}\,, \quad \forall i, \ell \in \{0,\ 1, \ldots,\ (M-1)\}. \tag{5.4}$$

Each of these continuous-time signals $s_i(t)$ or $g_i(t)$ will be represented as a vector in discrete-time k with $s[k] = s(kT_s)$. Each signal element can be represented by a vector with T/T_s elements (T/T_s must be an integer) and which contains one symbol. Thus the quotient T/T_s describes the number of samples per symbol.

The orthogonality condition must then be described in discrete-time giving

$$\frac{1}{E_g} \sum_{k=-\infty}^{+\infty} g_l[k] \cdot g_\ell^*[k] = \underline{\underline{\delta_{l\ell} \cdot \delta_{0k}}}\,, \quad \forall i, \ell \in \{0,\ 1, \ldots,\ (M-1)\}. \tag{5.5}$$

If all signal elements or basis vectors of the signal space are summarized into a single vector we get

$$\boldsymbol{g}(t) \stackrel{\text{def}}{=} [g_0(t), g_1(t), \ldots, g_{M-1}(t)] \qquad \text{(continuous-time)} \tag{5.6}$$

$$\boldsymbol{g}[k] \stackrel{\text{def}}{=} [g_0[k], g_1[k], \ldots, g_{M-1}[k]]\,. \qquad \text{(dicrete-time)} \tag{5.7}$$

In case of discrete-time the resulting matrix $\boldsymbol{g}[k]$ has the dimension

$$\text{Number of signal elements} \times (T/T_s). \tag{5.8}$$

The double orthogonality condition can then be tested with:

$$\frac{1}{E_g} \sum_{k=-\infty}^{+\infty} \boldsymbol{g}[k] \cdot \boldsymbol{g}^*[k] = \underline{\underline{\boldsymbol{I}_{M\times M}\delta_{0k}}} \tag{5.9}$$

5.4.2 Orthogonalization

Although in most digital transmission scheme the signal elements are not mutually orthogonal the same signal space can be represented using orthonormal basis vectors.

The goal is to find basis vectors $g_i(t)/g_i[k]$ (which are in most cases not unique for a given set of signal elements) and its linear factors $s_{i,l}$ so that each signal element can be represented as a *linear combinations* of weighted basis vectors [Hub11]:

$$s_i[k] = \sum_{l=0}^{D-1} s_{l,i} \cdot g_l[k] \tag{5.10}$$

Introducing a matrix-vector notation, we have

$$\boldsymbol{s}[k] = [s_0[k], \ldots, s_{D-1}[k]]\,, \tag{5.11}$$

$$\boldsymbol{g}[k] = [g_0[k], \ldots, g_{D-1}[k]] \text{ and} \tag{5.12}$$

$$\boldsymbol{S} = \begin{bmatrix} s_{0,0} & \cdots & s_{0,(M-1)} \\ \vdots & s_{\ell,i} & \vdots \\ s_{(D-1),0} & \cdots & s_{1,(M-1)} \end{bmatrix} \tag{5.13}$$

which results in

$$\boldsymbol{s}[k] = \boldsymbol{g}[k] \cdot \boldsymbol{S}. \tag{5.14}$$

At this point it should be clear that there might suffice $D \leq M$ orthonormal basis vectors $g_i(t)$ to span the signal space.

This representation is a linear transformation of basis vectors into signal elements. This process can be reversed. The orthonormal basis vectors can be represented as linear combinations of *orthogonalization factors* $g_{i,l}$ and the signal elements themselves:

$$g_l[k] = \sum_{i=0}^{M-1} g_{i,l}\, s_i[k]. \tag{5.15}$$

To find a set of orthonormal basis functions we will focus on the so called Gram-Schmidt Procedure (GSP) which is described by a recursive algorithm [Hub11] and calculates the

- *Linear Factors* (linear factor matrix $\boldsymbol{S}$) and the
- *Orthonormal Basis Function* $g_i(t)/g_i[k]$

A set of M row vectors $s_0[k], s_1[k], \ldots, s_{M-1}[k]$ representing the signal elements over time k is given. The goal is to find a set of orthonormal basis vectors $g_0[k], g_1[k], \ldots, g_{D-1}[k]$ with unknown dimensions $D \leq M$ and linear factors $s_{\ell,i}$ such that

$$s_i[k] = \sum_{\ell=0}^{D-1} s_{\ell,i} \cdot g_\ell[k]. \tag{5.16}$$

The recursive algorithm works as follows:

1. Calculate the first basis vector using the first signal element vector

$$g_0[k] = \frac{1}{\sqrt{\sum\limits_k s_0[k] \cdot s^*[k]_0}} s_0[k]$$

Number of basis vectors determined so far: $l = 1$

Linear factors for $\boldsymbol{s}_0$:

$$s_{0,0} = \sqrt{\sum_k s_0[k] \cdot s_0^*[k]}, \quad s_{\ell,0} = 0 \text{ for } \ell = 1, 2, \ldots, D-1 \tag{5.17}$$

$$\boldsymbol{S} = \begin{bmatrix} \sqrt{\sum\limits_k s_0[k] \cdot s_0^*[k]} & \cdots \\ 0 & \\ \vdots & \ddots \\ 0 & \end{bmatrix} \tag{5.18}$$

$$\boldsymbol{g}[k] = \begin{bmatrix} \frac{1}{s_{0,0}} s_0[k], & \cdots \end{bmatrix} \tag{5.19}$$

Dimension D still undetermined.

2. Moving on to the next vector $s_i[k]$ $(i = 1, 2, M-1)$:

   a) Linear factors regarding the l basis vectors computed so far:

$$s_{\ell,i} = \sum_k s_i[k] \cdot g_\ell^*[k] \tag{5.20}$$

$$\boldsymbol{S} = \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & \cdots \\ 0 & s_{1,1} & s_{1,2} & \ddots \\ 0 & 0 & s_{2,2} & \ddots \\ 0 & 0 & 0 & \ddots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{5.21}$$

$$\boldsymbol{g}[k] = \left[\left(\tfrac{1}{s_{0,0}} s_0[k]\right), \quad \left(\tfrac{1}{s_{0,1}} s_0[k] + \tfrac{1}{s_{1,1}} s_1[k]\right), \quad \cdots \right] \tag{5.22}$$

   b) Now test if current signal element can already be represented with the existing basis vectors:

$$\Delta_i = \underbrace{\sum_k s_i[k] \cdot s_i^*[k]}_{\text{energy of the signal element}} - \underbrace{\sum_{\ell=0}^{l-1} |s_{\ell,i}|^2}_{\text{energy of the existing basis vectors}} \tag{5.23}$$

      i. $\Delta_i = 0$: The current signal element $s_l[k]$ can fully be represented by a linear combination of existing basis vectors. No new basis vector is needed.

$$s_{\ell,i} = 0 \text{ für } \ell = l, l+1, \ldots, D-1$$

      ii. $\Delta_i > 0$: There is energy left, that cannot be represented by a linear combination of basis vectors. Thus a new basis vector has to be introduced.

$$g_l[k] = \frac{1}{\sqrt{\Delta_i}} \left(s_i[k] - \sum_{\ell=0}^{l-1} s_{\ell,i} g_\ell[k] \right) \tag{5.24}$$

      Determine the new linear factor for $s_i[k]$:

$$s_{l,i} = \sqrt{\Delta_i}, \quad s_{\ell,i} = 0 \text{ for } \ell = l+1, l+2, \ldots, D-1 \tag{5.25}$$

      Increment number of basis vectors: $l \to l+1$

3. As long as $i < M-1$, repeat step 2 with vector $s_{i+1}[k]$ (recursion)

4. Orthogonalization finished. Dimensionality: $D = l$

The result is a set of new orthonormal basis vectors which describes the identical signal space as the original set of signal elements, and a set of linear factors for each basis.

The above description uses an “incremental” order to select the next signal element in the progress. Alternatively one can choose that signal element that has the maximum energy Δ_i that cannot be represented as linear combination of basis vectors. This modification is later on called “sorted-by-energy”.

Readings for Lab 5

[Hub11] Johannes B. Huber, *Digital Communications*, Lecture Notes, Erlangen, October 2011.

[Kam08] Karl-Dirk Kammeyer, *Nachrichtenübertragung*, 4 ed., B. G. Teubner, Stuttgart, March 2008.

[Pro00] John G. Proakis, *Digital communications.*, 4th ed., McGraw-Hill, New York, NY, USA, 2000.

5.5 Lab Exercises

5.5.1 Transmission with signal elements

We define the signal elements as depicted in Fig. 5.2. There are three different transmission schemes with signal elements $s_{1,i}(t)$, $s_{2,i}(t)$ and $s_{3,i}(t)$, $i \in \{0, \dots, 3\}$ defined. Each set of signal elements describe its own transmission scheme connecting signal numbers to waveforms.

On the other hand one can interpret the set of signal elements as a multiplexing scheme where multiple users (in our case four users) can transmit data at the same time using one individual signal element.

Homework H-5.1

Assign the signal elements to the following multiplexing schemes

- Time Division Multiplex
- Frequency Division Multiplex
- Code Division Multiplex

and justify your decision!

We now neglect the ability to implement a multiplexing scheme using signal elements and focus on the so call point-to-point transmission. This means, one transmitter with a set of signal elements and one receiver which also knows the signal elements. By selecting a signal element at each time instant the transmitter can communicate a sequence of information symbols.

Lab Exercise L-5.1

Implement a transmitter that uses the signal elements that describes a code multiplexing transmission scheme! Copy your existing implementations of a transmitter and extend it so that you can use it in the following form:

```
signalElements = generateSignalElements(type, overSamplingFactor);
transmittedSignal = transmitter_SE(bitStream, ...
                                   signalElements);
```

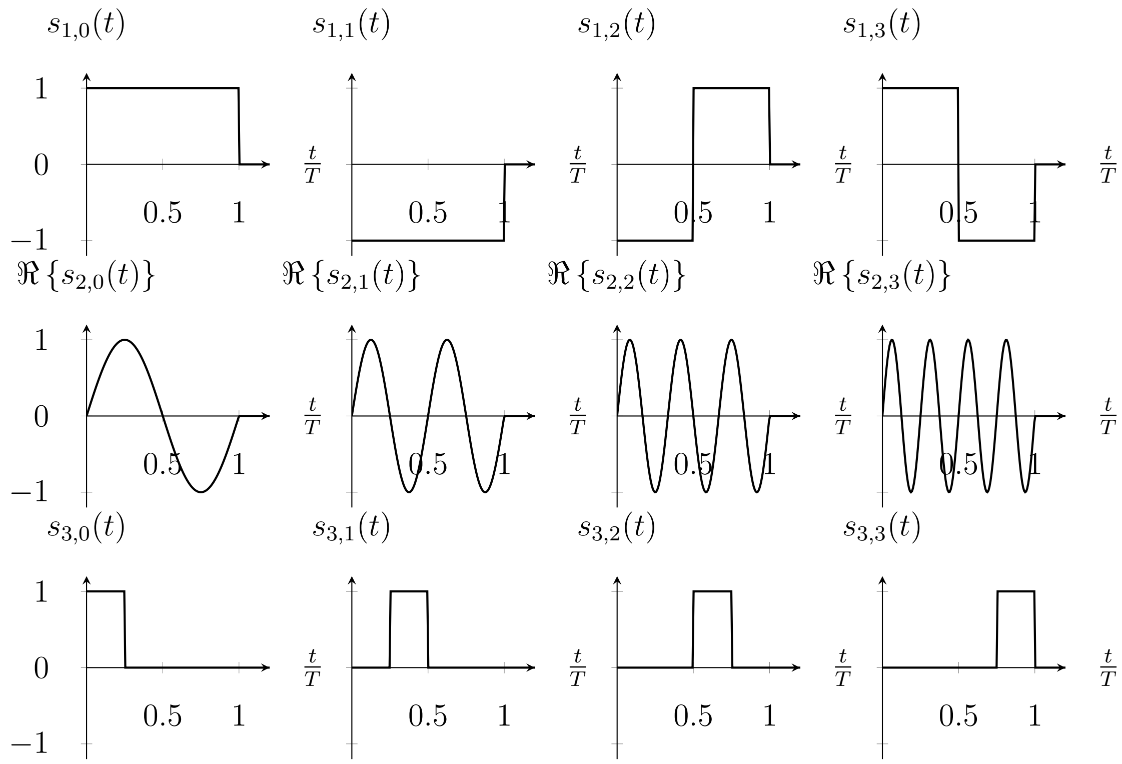


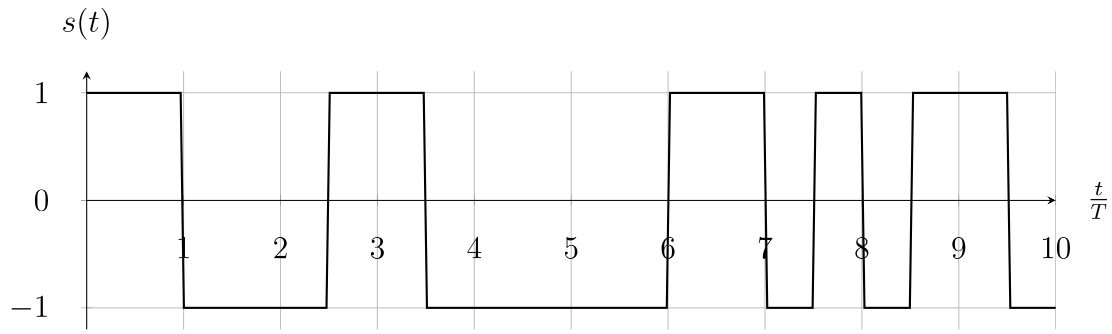
Figure 5.2: Three different transmissions schemes and its signal elements

The parameter `type` is a string that describes which set of signal elements are used. Note that these signal elements are generated by a separate function and stored in the variable `signalElements` which are then used in the transmitter. Proceed as follows:

1. Familiarize yourself with the functions `bi2de` and `de2bi` and use these within your transmitter and receiver (later).
2. Write a function `generateSignalElements`. Generate your set of signal elements depending on the parameter `type`!
3. Take a look at the signal elements (real and imaginary party, absolute value). Use an `oversamplingFactor` of 128 for testing purpose.
4. In the transmitter, map the sequence of binary data to the signal numbers! (use natural labeling)
5. Make sure the mapping can address all signal elements!
6. Generate the transmit signal $s(t)$ by concatenation!

Homework H-5.2

Determine the sequence of signal numbers for the following data sequence which uses the signal numbers $s_{1,0}(t), \ldots s_{1,3}(t)$!



Lab Exercise L-5.2

Generate the given transmit signal with your implementation (ECB-domain).

Homework H-5.3

Using your knowledge in signal space representation, what signal elements exist for a 4-ary PAM with a rectangular pulse-shaping? And how many possibilities exist to get from binary data streams to signal numbers? Also

- sketch all signal elements in one diagram!
- give the equation to generate each signal element!
- determine the number of orthonormal basis functions are needed to represent all signal elements? (No calculation necessary!)

5.5.1.1 Vector Receiver

Our transmission system now contains a transmitter which selects for each time instant t a signal element of a M dimensional set of signal elements $\mathcal{S} = \{s_0(t), s_1(t), \dots, s_{M-1}(t)\}$. By concatenating signal elements we generated the transmit signal which is transmitted over an AWGN channel.

A coherent Maximum Likelihood detection can be achieved by a correlation receiver as depicted in Fig. 5.3. Here, the received signal $r(t)$ is uses as an input signal for a vector receiver, or matched filter bank which consists of the matched filter for each signal element in the signal space $\mathcal{S}$. This vector receiver correlates the received signal $r(t)$ with each signal element $s_i(t)$. After sampling, we get a correlation vector $\boldsymbol{d}[k]$ which constitutes a **sufficient statistics** on the receiver input signal with respect to the transmitted sequence of signal numbers $m[k]$. The decision for a signal number is then simply made by picking the **maximum in the correlation vector**.

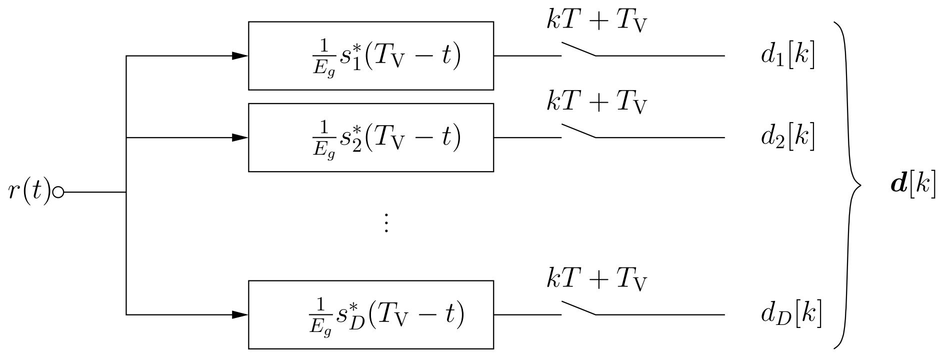


Figure 5.3: Correlation Receiver (also: Vector Receiver) [Hub11]

Lab Exercise L-5.3

Implement a vector receiver using the signal elements defined in exercise L-5.1! We recommend to first implement the filter bank and the correlation using the MATLAB table with the signal elements. Take a look at the results of this correlations before implementing the sampling. After locating the maximum value in the correlation vector compare the sequence of signal numbers with those at the transmitter!

Lab Exercise L-5.4

Implement a function

```
ber = function SigSpaceTrans(binData, sigElements, SNR)
```

which performs a complete transmission using predefined signal elements. Proceed as follows:

1. Use your transmitter to generate the transmit signal $s(t)$ and the signal elements!
2. Perform a transmission over an AWGN channel using the predefined noise variance!
3. Use the noisy signal as input signal for the vector receiver!
4. Sample the signal and decide for the most likely signal numbers!
5. Reverse the mapping from signal numbers to a binary sequence!
6. Count the number of bit errors made!

Lab Exercise L-5.5

Implement a simulation of the transmission for different $\frac{E_s}{N_0}$ and plot the resulting error probabilities over the SNR! Use a data block with a length of at least 10^5. Reduce the `oversamplingFactor` to around 10.

5.5.2 Gram-Schmidt Procedure

We now consider the Gram-Schmidt Procedure to orthogonalize the signal elements. Therefor we use the MATLAB-table which contains the signal elements.

Lab Exercise L-5.6

Implement the Gram-Schmidt Procedure in MATLAB!

Homework H-5.4

Use the Gram-Schmidt Procedure to orthogonalize the following set of signal elements:

$$\begin{aligned}
\boldsymbol{a} &= [\ \ 1,\ \ 1,\ 1,\ \ 1\ ] \\
\boldsymbol{b} &= [\ -1,\ -1,\ 1,\ -1\ ] \\
\boldsymbol{c} &= [\ \ 2,\ \ 2,\ 0,\ \ 2\ ]
\end{aligned}$$

Lab Exercise L-5.7

Compare the result of your homework with the result of your implementation!

Lab Exercise L-5.8

Orthogonalize the signal elements from exercise 1! Sketch the resulting orthonormal signal elements $g_i(t)$! How many basis functions are necessary?

Lab Exercise L-5.9

Discuss how to describe the original signal elements $s_i(t)$ using the orthonormal basis vectors $g_i(t)$! Rebuild the original signal elements in MATLAB.

Homework H-5.5

Describe in your own words how the receiver in task 5.4 has to be modified so that a matched filter bank with less filters can be used!

Homework H-5.6

If the orthogonal basis do not contain the zero-vector, which basis vector would you use for your reduced complexity receiver and which would you neglect? Does the option "incremental" and "sorted-by-energy" influence the decisions?

5.5.3 Frequency Shift Keying

In a M-ary frequency shift keying modulation (FSK) [Kam08, Pro00] scheme the signal elements are defined as

$$s_i(t) = \begin{cases} \sqrt{E_g/T}\ \mathrm{e}^{\mathrm{j}(2\pi i\, h\, t/T + \varphi_0)} & \text{for } t \in [0,\ T) \\ 0 & \text{for } t \notin [0,\ T) \end{cases} \tag{5.26}$$

For the time-discrete version it holds $t = kT_s$:

$$s_i[k] = \begin{cases} \sqrt{E_g/T}\ \mathrm{e}^{\mathrm{j}(2\pi i\, h\, (kT_s)/T + \varphi_0)} & \text{for } k \in \{0, 1, \ldots, (T/T_s - 1)\} \\ 0 & \text{else} \end{cases} \tag{5.27}$$

with a modulation index $h \in \mathbb{R}$ which describes the maximum phase shift and the symbol numbers $i = 1,\ 2, \ldots,\ M$. The transmit signal $s(t)$ can then again be described as a concatenation of signal elements.

Homework H-5.7

> Determine the modulation index h and the alphabet size M of an FSK scheme which uses the signal elements $s_{2,0}(t)$,. . . ,$s_{2,3}(t)$ in Fig. 5.2 (real value of $s_{2,i}(t)$ plotted)!

Lab Exercise L-5.10

> Generate a FSK signal elements for $M = 4$ and $h = \frac{1}{4}$! Plot them and describe the result!

Lab Exercise L-5.11

> Determine the orthonormal basis functions for the signal space in exercise 10! Describe the resulting basis vectors!

Homework H-5.8

> Which sets of parameters in FSK result in orthogonal signal elements?

Project 6

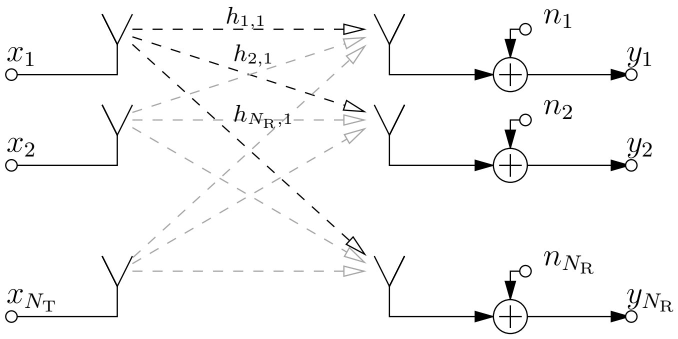
Signal Processing in MIMO Systems

6.1 Introduction, Background, and Motivation

Modern digital communication systems employ multiple antennas at the transmitter and the receiver side. Two basic principles can be distinguished which constitute the benefits of multiple-input/multiple-output (MIMO) systems compared to single antenna systems: On the one hand the multiplexing gain, i.e., the increased data rate due to transmission of independent data streams over multiple transmit antennas. On the other hand, the possibility to observe several independent copies of the transmit signal, e.g., through the use of multiple receive antennas, and thus increased robustness to shadowing and noise due to diversity.

In this lab, an introduction to signal processing for MIMO communications is given, motivating the potentials and the benefits of the techniques enabled through the use of multiple antennas at transmitter and/or receiver.

To this end, we consider the discrete-time equivalent model of digital pulse-amplitude modulation transmitted over a flat-fading MIMO channel with AWGN as depicted below.



Aspects such as pulse-shaping, up-/down-conversion to and from the carrier frequency, as well as matched filtering are hidden in this equivalent model, and implicitly assumed to operate perfectly synchronized. With these assumptions, adopting a compact vector/matrix-notation we define the transmit signal as

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_{N_\mathrm{T}} \end{bmatrix} \tag{6.1}$$

and the received signal as

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_{N_\mathrm{R}} \end{bmatrix} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{6.2}$$

where $\boldsymbol{n}$ is a $(N_\mathrm{R} \times 1)$-vector collecting the additive white Gaussian noise samples on each receive antenna and $\boldsymbol{H}$ denotes the $(N_\mathrm{R} \times N_\mathrm{T})$-dimensional MIMO channel matrix given by

$$\boldsymbol{H} = \begin{bmatrix} h_{1,1} & \dots & h_{1,N_\mathrm{T}} \\ \vdots & \ddots & \vdots \\ h_{N_\mathrm{R},1} & \dots & h_{N_\mathrm{R},N_\mathrm{T}} \end{bmatrix} \tag{6.3}$$

The transmit symbols x_i, $i = 1, \dots, N_\mathrm{T}$, are taken from an arbitrary complex-valued signal constellation (here, we will only consider square QAM constellations).

We begin with studying the special case of single-input/multiple-output (SIMO) transmission, i.e., the transmitter communicates a single data stream using a single transmit antenna ($N_\mathrm{T} = 1$) to a receiver which is equiped with multiple receive antennas (N_R). The results are compared to a conventional single-input/single-output (SISO) system. Next, we consider the dual case of multiple-input/single-output (MISO) transmission ($N_\mathrm{T} \geq 1$, $N_\mathrm{R} = 1$) and briefly review a technique called space-time codes. We will then extend our considerations to include multiple antennas at transmitter and receiver side and implement and discuss two different (receiver-side) equalization strategies for this MIMO system. To this end, we first implement a generic system model of MIMO transmission using QAM.

The notation and organization of this lab closely follows [?]; the corresponding course and its lecture notes are not required for preparation of this lab.

Noteworthy, multi-antenna systems are only one, but probably the most intuitive example for MIMO systems; other communication systems, as, e.g., code-division multiple-access (CDMA) systems, can be modeled in a similar way. The studied principles of this lab are thus applicable in a broad class of digital communication systems.

6.2 Lab Environment

The MIMO transmission system considered in this lab is implemented in MATLAB. Different from previous labs, all simulations operate in the discrete-time equivalent system of digital transmission. Pulse shaping and modulation is not required; functions of previous labs are not reused.

Readings for Lab 6

6.3 Lab Exercises

6.3.1 System Model

Homework H-6.1

Give an equation for the variance of the transmit symbols for the case of a M-ary square-QAM constellation. Assume that the signal points are taken from the grid

$$\mathcal{A} \subseteq \{\pm 1 \pm \mathrm{j},\ \pm 3 \pm \mathrm{j},\ \pm 1 \pm 3\mathrm{j},\ \pm 3 \pm 3\mathrm{j}, \dots\}\ . \tag{6.4}$$

Lab Exercise L-6.1

Implement a function

```
x = GetQAM(r, c, M)
```

which generates an $r \times c$ matrix $\boldsymbol{x}$ containing randomly drawn M-ary QAM symbols taken from $\mathcal{A}$.
Generate a vector $\boldsymbol{x}$ of reasonable length and determine the variance of the QAM symbols.
Compare this result with the analytical one from the lecture notes.

Lab Exercise L-6.2

Implement a function

```
a_hat = QuantQAM(z, M)
```

which quantizes the elements of the matrix $\boldsymbol{z}$ to M-ary QAM symbols.
Use the MATLAB-function `round` and note the finite size of the constellation.

In this lab we will only consider the following channel model:

- the elements $h_{m,n}$ of the MIMO channel matrix $\boldsymbol{H}$ are independent zero-mean unit-variance complex-Gaussian distributed

- the elements of the additive noise vector are independent zero-mean complex-Gaussian distributed with equal variance σ_n^2.

Homework H-6.2

Interprete the assumptions of this channel model and briefly explain the reasoning behind.

Homework H-6.3

Give simple methods, which generate a random channel realization $\boldsymbol{H}$ and a realization of the noise vector $\boldsymbol{n}$ according to this model in MATLAB.

In this lab, we restrict ourselves to the performance measure symbol error rate (SER).

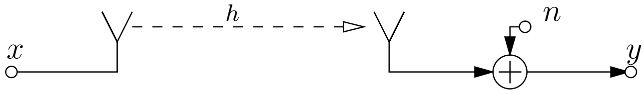
Homework H-6.4

Give a simple method to calculate the SER in MATLAB given the QAM transmit symbols and its estimates.

We now have implemented three major building blocks of a generic MIMO transmission: the transmitter generating a sequence of QAM symbols transmitted over multiple antennas, the MIMO AWGN channel, and the decision device quantizing the equalized symbols into QAM symbols.

6.3.2 SISO

As a reference, we first implement a conventional SISO system ($N_\mathrm{R} = N_\mathrm{T} = 1$), as depicted below.



As in the generic MIMO scenario we assume the fading coefficient h to be complex-Gaussian distributed with zero mean and unit variance.

Homework H-6.5

State the name of the distribution of the magnitude of the fading coefficient, i.e., the channel gain, and give a formula for its probability density function (pdf).

Lab Exercise L-6.3

Write a simulation script which generates and transmits a sequence of QAM symbols according to this setup.
The results shall be averaged over a large number of channel realizations. Include a `for`-loop in order to simulate the SER for different SNRs and plot the resulting SER-vs.-SNR curve. Take care that the SNR is correctly set. Measure the SER for 4-QAM and an SNR range of $-10, 5, \ldots, 50$.

Note: Several (L) symbols can be transmitted over a single channel realization simultaneously by constructing a $(1 \times L)$-dimensional transmit vector.

Lab Exercise L-6.4

Estimate the pdf of the channel gain (using the built-in MATLAB-function `hist`) and compare it to the analytical results from problem 5.

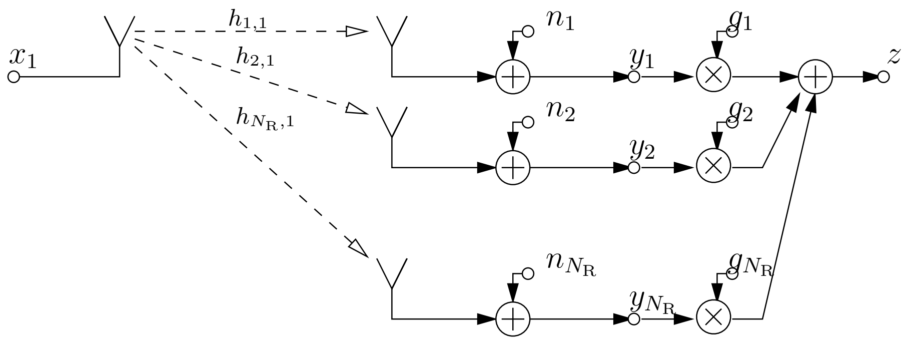
6.3.3 SIMO

We now consider the case of transmission of a single data stream (transmit symbols x with variance σ_x^2) using a single transmit antenna ($N_\mathrm{T} = 1$) to a receiver equiped with multiple (N_R) receive antennas.

Grouping the receive symbols into a vector, we have

$$\boldsymbol{y} = \boldsymbol{h}x + \boldsymbol{n} \tag{6.5}$$

Linearly combining the signals observed at the antennas as depicted below (combining coefficients g_n) offers to significantly increase robustness to noise (diversity gain); there is no multiplexing gain compared to the SISO system.



Assuming channel state information at the receiver side, the optimum (w.r.t end-to-end SNR) combining strategy for this SIMO setup is maximum ratio combining (MRC) which sets $g_n = h_n^*$.

Homework H-6.6

Write the signal processing of MRC compactly in vector/matrix-notation and give an interpretation of the underlying principle.

Lab Exercise L-6.5

Extend your simulation setup to incorporate multiple antennas at receiver side and implement MRC. Measure the SER for 4-QAM, $N_\mathrm{R} = 2$, and an SNR range of $-10, 5, \ldots, 50$.
For comparison, the SER of both schemes SISO and SIMO shall be recorded.

Note that the combined receive signal has to be scaled to match the expected input range of the decision device.

Lab Exercise L-6.6

Which effect is observed for an increasing number of receive antennas.
Run simulations for $N_\mathrm{R} = 4$, and 8 and also for different QAM constellations to verify your observations.

Using MRC, the equivalent effective channel $z = \|\boldsymbol{h}\| x$ can be introduced, i.e., the effective channel gain calculates to $\|\boldsymbol{h}\|^2$.

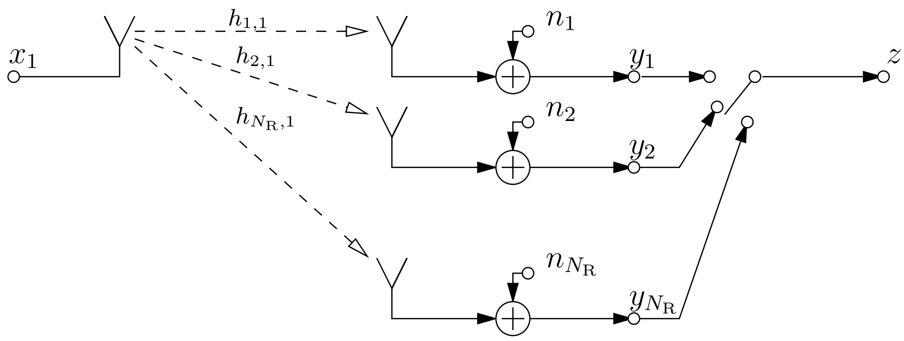
Homework H-6.7

State the name of the distribution of the effective the channel gain and give its mean and variance.

Lab Exercise L-6.7

Estimate the pdf of the channel gain (using the built-in MATLAB-function `hist`) and compare it to the analytical results and the pdf of the channel gain of SISO transmission.

A simpler scheme to exploit the benefits of multiple antennas at the receiver is so-called antenna selection (AS), as depicted below.



In AS, for each time step, the best antenna is selected; its receive symbol is used as a decision variable.

Homework H-6.8

Explain the advantages of AS over MRC with respect to hardware implementation complexity.
Which antenna should be selected in each time step?

Lab Exercise L-6.8

Extend your simulation script to incorporate AS and compare the resulting performance with MRC and the SISO system for different values of N_R.
Which effect do you observe?

6.3.4 MIMO

In the final part of this lab, we investigate systems employing multiple antennas at both transmitter and receiver side. Compared to the previous systems, such schemes offer the potential to have both, a multiplexing gain (increased data rate through the transmission of independent data symbols on the antennas for each time step) and a diversity gain (increased robustness to noise).

In vector/matrix-notation, the MIMO receive signal can be written as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} \tag{6.6}$$

Similar to equalization of inter-symbol interference in digital communication over dispersive channels, the straight-forward approach for equalization of the interference induced by the use of the same transmit medium (time/frequency) is so-called zero-forcing linear equalization (ZF-LE). In case of square systems ($N_\mathrm{T} = N_\mathrm{R}$) the receive signal is processed with the inverse of the MIMO channel matrix, yielding

$$\boldsymbol{z} = \boldsymbol{H}^{-1}\boldsymbol{y} \tag{6.7}$$

In case of non-square systems, the left-pseudo inverse (Moore-Penrose inverse) has to be employed, yielding

$$\boldsymbol{z} = \left(\boldsymbol{H}^\mathsf{H}\boldsymbol{H}\right)^{-1}\boldsymbol{H}^\mathsf{H}\boldsymbol{y} \tag{6.8}$$

Lab Exercise L-6.9

Write a new simulation script which implements MIMO transmission with ZF-LE and run simulations for 4-QAM using $N_\mathrm{T} = N_\mathrm{R} = 4$.
Which diversity order is obtained using this simple equalization strategy?
Note: Several symbols can be transmitted over a single channel realization simultaneously, by letting $\boldsymbol{x}$ represent a matrix of dimension $N_\mathrm{T} \times$ number of channel uses.

Similar to equalization of inter-symbol interference in digital communication over dispersive channels, linear equalization is also possible in a minimum mean-squared error (MMSE) sense. In this case, we have

$$\boldsymbol{z} = \left(\boldsymbol{H}^\mathsf{H}\boldsymbol{H} + \frac{\sigma_n^2}{\sigma_x^2}\boldsymbol{I}\right)^{-1}\boldsymbol{H}^\mathsf{H}\boldsymbol{y} \tag{6.9}$$

Lab Exercise L-6.10

Extend your simulation script incorporating MMSE-LE and compare the results with ZF-LE for different parameter settings.

The optimum detection scheme for MIMO transmission is maximum-likelihood detection. As the receive signal $\boldsymbol{y}$ is corrupted by AWGN, this is equivalent to finding the noise-free signal point with minimum Euclidean distance to the receive signal point, i.e.,

$$\boldsymbol{x}^{\mathsf{ML}} = \operatorname{argmin}_{\tilde{\boldsymbol{x}}} \|\boldsymbol{y} - \boldsymbol{H}\tilde{\boldsymbol{x}}\|^2 \tag{6.10}$$

There are various algorithms to solve the ML detection problem with moderate computational complexity. Here, we only consider the simple strategy of a full search over all candidate signal points.

Homework H-6.9

Calculate and discuss the number of candidate signal points for M-QAM transmitted over N_T transmit antennas.

Lab Exercise L-6.11

Implement a separate function for ML detection based on the full-search approach and include it in your simulation script.
Compare the results to ZF/MMSE-LE for $N_\mathrm{T} = N_\mathrm{R} = 2$ and 4-QAM.
