The Model of the Low Rate Telemetry Communication System for Matlab-Simulink

Jiri SPACEK, Miroslav KASAL

Dept. of Radio Electronics, Brno University of Technology, Purkyňova 464/118, 612 00 Brno, Czech Rep.

jiri.spacek@email.cz, kasal@feec.vutbr.cz

Abstract. *This article is dedicated to the model of low rate telemetry system, which has been developed for Matlab-Simulink environment. The purpose of this model is a research of the low rate telemetry transmission reliability in those cases where the modulation scheme carrier-subcarrier is used. This modulation scheme is widely used in case of the interplanetary spacecrafts. The main purpose of the model is a research of the effects of AWGN and phase noise especially for very low value of E_b/N_0. Effects can be evaluated for the whole transmission system or for its components parts. The model described is very versatile and it can be easily modified or expanded.*

Keywords

Simulation, data transmission system, telemetry, carrier-subcarrier, BPSK/BPSK, very low E_b/N_0, AWGN, phase noise.

1. Introduction

In the last few years the research of nearby planets of Solar system is underway. In that research automatic space probes are used.

The space probe transmits - predominantly in the X-band (~8.4 GHz) - low rate telemetry data, usually $R = 4 \div 1000$ bps, [1]. The carrier phase modulation generally has the peak modulation index between (40 - 80) deg, i.e. it is the residual carrier modulation, but using the suppressed carrier modulation is also possible. The modulation signal for the deep space probe carrier modulator is totaled by the BPSK modulated (square- or sine-) wave subcarrier, and the modulation signal is created by data in the NRZ-L format, [2]. The concatenated code is used for telemetry data forward error correction. Outer code uses a Reed-Solomon RS(255,223) and inner code uses a convolutional code (r = 1/6, k = 15).

Telemetry signals are received by the *Deep Space Network* with very sensitive receivers and 34-meters high-efficiency dish antennas.

Under special conditions, there is a theoretical possibility to receive, or at least indicate, deep space probe sig-nal with relatively small aperture antenna, but received signals are extremely weak - almost lost in the noise - and receiving tract must therefore be perfectly adjusted.

Computer modeling can be relatively quick and cheap way of determining influence for each subsystem of the receiving system. For this purpose, the presented model for Matlab-Simulink environment has been developed.

The presented model allows studying the influence of Additive White Gaussian Noise (AWGN) and Phase Noise (PN) for low rate telemetry transmission with modulation scheme BPSK/BPSK, i.e. with the suppressed carrier. The model does not contain any forward error correction. Nevertheless, it can be easily modified and then used for studying other effects or other modulation schemes with subcarriers.

In this article the developed model is described at the block level. Then the process for model blocks parameters design is mentioned. Some of achieved results for modulation scheme BPSK/BPSK are also presented.

2. Model of Communication System

The block diagram of the developed model is presented in Fig. 1.

Random data stream, $S \in \{\,0\,;\,1\,\}$ is generated in the block *Bernoulli Binary Generator*. User can specify the probability of a zero and also source bit rate. The bit rate R_b is given by

$$R_b = 1/T_b \tag{1}$$

where T_b is user-defined sample time. Next the data level conversion $\{\,0\,;\,1\,\} \Rightarrow \{\,-1\,;\,1\,\}$ with block *Lookup Table.1* is done.

As a baseband filter the Square root Raised Cosine Transmit Filter is used. The user can specify rolloff factor α, length of filter impulse response and the upsampling factor N. This factor determines the ratio between the filter output data sampling frequency $f_{vz\,RC\,out}$ and the source bit rate, so
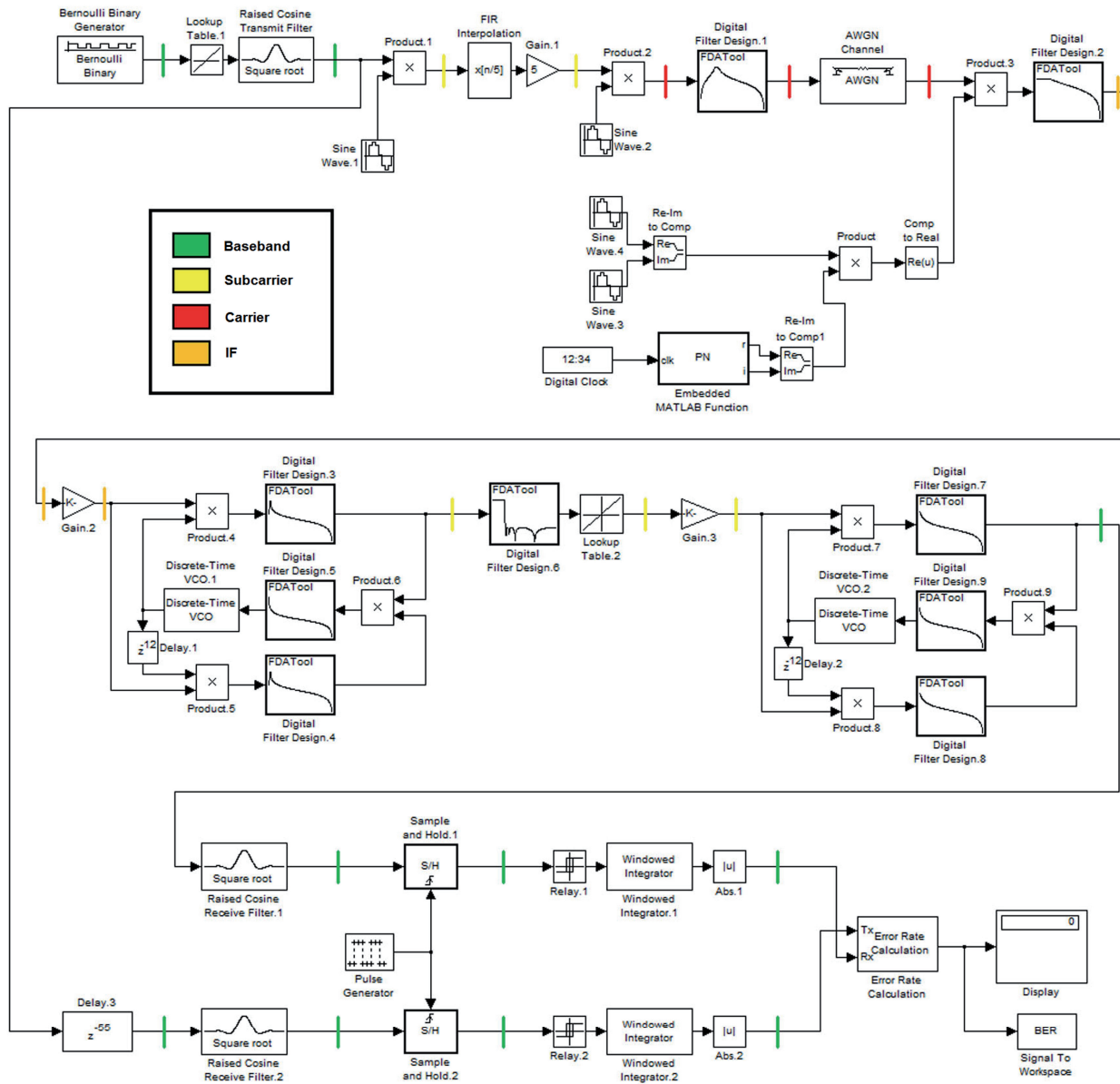
$$N = f_{vz\,RC\,out} / R_b. \tag{2}$$

Fig. 1. The block diagram of developed model.

Output data sampling frequency $f_{vz\,RC\,out}$ is equal to sampling frequency for signal modulated on subcarrier $f_{vz\,sub}$. We use different sampling frequencies for subcarrier and for carrier to reduce computational cost of the model.

Next, the subcarrier BPSK modulation is done. The modulator is created with the block *Product.1*.

The harmonic subcarrier wave is created with block *Sine Wave.1*. The frequency of the subcarrier is user-specified. In our simulations it is $f_{sub} = 8$ kHz with $f_{vz\,sub} = 128$ kHz.

Before modulation on the carrier, the interpolation $f_{vz\,sub} \rightarrow f_{vz\,carr}$ is necessary. Interpolation factor is given by

$$I = f_{vz\,carr} / f_{vz\,sub}, \tag{3}$$

where $f_{vz\,carr}$ is sampling frequency for signal modulated on carrier. Interpolation lowpass filter is a part of block *FIR Interpolation*. The normalized cutoff frequency of this filter is given by

$$F_c = 1/I. \tag{4}$$

In our simulations we use $f_{vz\,carr} = 640$ kHz, $I = 5$, $F_c = 0.2$ and the 30-th order of interpolation filter.

Next, the carrier BPSK modulation is done. The multiplicative modulator is created with block *Product.2*.

The harmonic carrier wave is created with block *Sine Wave.2*. The frequency of the carrier is user-specified. In our simulations it is $f_{carr} = 80$ kHz. After the carrier modulation, the resulting signal modulation scheme is BPSK/BPSK.

The output bandpass filter of the transmitter is created with block *Digital Filter Design.1*. The center frequency of this filter f_0 is equal to the carrier frequency f_{carr}. The bandwidth of this filter, B_{3dB}, is not very critical and is selected to meet the relationship

$$B_{3dB}\{DFD.1\} >\approx 2.f_{sub} + R_b.(1+\alpha) \qquad (5)$$

where the symbol ">≈" means "slightly wider". After the output filtration, the transmission through the AWGN channel is done. Block *AWGN Channel* adds white Gaussian noise to the input signal. The user can specify the value of E_b/N_0 [dB]. Writing this parameter in the form "$E_b/N_0 + 3$ dB" is used because of the subsequent mixer impact. The mixer causes the frequency transposition. Consequently, in the frequency band of carrier Costas loop only the half of the signal power (-3dB) will be found.

One of our tasks is to study the impact of phase noise. The presented model contains a local oscillator with user-definable shape of phase noise. Blocks *Sine Wave.3*, *Sine Wave.4* and *Re-Im To Comp* generate the complex harmonic signal with frequency f_{LO}. In our simulations $f_{LO} = 120$ kHz with sampling frequency $f_{vz\ carr} = 640$ kHz. The block *Embedded MATLAB Function* "PN" is linked with source code of phase noise model for Matlab environment. This part of our communication system model is based on the phase noise model by Alex Bar [5], and it is described in detail in our paper [3]. User can define the shape of the phase noise in the way

$$L = [\ 0\quad ;\quad -a\ ;\quad -b\ ;\quad -c\ ...]\ \text{dBc/Hz}$$

$$@\ \Delta f = [\ 0\quad ;\quad 1\ ;\quad 10\ ;\quad 100\quad ...]\ \text{Hz}.$$

At the output of *Embedded MATLAB Function* "PN" block the real and imaginary component of the phase noise sample is available for every step of simulation. They are converted with the block *Re-Im To Comp.1* to complex form. Auxiliary block *Digital Clock* provides the control clock. In the block *Product*, the phase noise is superimposed on the complex harmonic signal with frequency f_{LO}. The output of this block is a complex harmonic signal with user-defined phase noise. This complex signal is with the block *Comp To Real* converted to the real harmonic phase noised signal with frequency f_{LO}.

The down-convertor is created with two blocks. The block *Product.3* is a multiplicative mixer and the block *Digital Filter Design.2* is a lowpass for filtering out undesired spectral components from frequency conversion. This lowpass is not necessary here, because subsequent carrier Costas loop has a narrow frequency band and it is not affected by unwanted spectral components out of its band.

Block *Gain.2* (*Gain.3*) is used to set the power of input signal for carrier (subcarrier) Costas loop. The Costas loop is, among others, designed for a specific input signal power. In our case the carrier (subcarrier) Costas loop is designed for $P\{Gain.2_{OUT}\} = 0.25$ W ($P\{Gain.3_{OUT}\} = 0.5$ W). The gain G of *Gain.2* (*Gain.3*) block is given by

$$G = \sqrt{\frac{P\{Gain._{OUT}\}}{P\{Gain._{IN}\}}}\ . \qquad (6)$$

Here, $P\{Gain._{IN}\}$ is power of signal at Costas loop input without the additive white Gaussian noise.

The carrier demodulator is done with a modified Costas loop. Modification lies in the arm filters of Costas loop - blocks *Digital Filter Design.3* and *Digital Filter Design.4* - usage of second-order bandpass filters instead of lowpass filters. Centre frequencies of these bandpass filters are $f_0 = f_{sub}$ and their bandwidth meets the condition

$$B_{3dB}\{DFD.3\} = B_{3dB}\{DFD.4\} >\approx R_b.(1+\alpha)\ . \qquad (7)$$

The symbol ">≈" means "slightly wider".

Blocks *Product.4* and *Product.5* (*Product.7* and *Product.8*) are phase detectors. Multiplier *Product.6* (*Product.9*) is a phase error detector, followed by the "Carrier loop filter" ("Subcarrier loop filter") – the filter created with block *Digital Filter Design.5* (*Digital Filter Design.9*). This loop filter is a 1-st order lowpass and the design of its cutoff frequency $\omega_{cA\ carr}$ ($\omega_{cA\ sub}$) is quite complex. It is described in detail in Section 4.

The voltage controlled oscillator of the carrier (subcarrier) Costas loop is created with block *Discrete-Time VCO.1* (*Discrete-Time VCO.2*). The quiescent frequency of this oscillator is $f_{0\ r\ carr} \approx f_{IF} = f_{LO} - f_{carr}$ ($f_{0\ r\ sub} \approx f_{sub}$). In our simulations, we use $f_{0\ r\ carr} = 40$ kHz ($f_{0\ r\ sub} = 8$ kHz) with sampling frequency $f_{vz\ carr} = 640$ kHz ($f_{vz\ sub} = = 128$ kHz). The design of the input sensitivity $K_{VCO\ carr}$ ($K_{VCO\ sub}$) of this VCO is also a complex task. It is described in detail in Section 4.

Block *Delay.1* (*Delay.2*) represents the delay of $\pi/2$, so harmonic signals at inputs of blocks *Product.4* and *Product.5* (*Product.7* and *Product.8*) are in quadrature. The length of FIFO queue in samples N, for block *Delay.1* (*Delay.2*) is given as

$$N_{carr\ (sub)} = \frac{3}{4}.\frac{f_{vz\ carr\ (sub)}}{f_{0\ r\ carr\ (sub)}}\ . \qquad (8)$$

In our model $N_{carr\ (sub)} = 12$ samples.

After the carrier demodulation the decimation $f_{vz\ carr} \rightarrow f_{vz\ sub}$ is done. The decimation factor is given as $D = I$, where I is given by (3). In our case $D = 5$. The first step of the decimation process is filtering the signal with an antialiasing lowpass filter of the 14-th order in *Digital Filter Design.6*. The cutoff frequency of this filter has to be slightly less than $f_{vz\ sub} / 2$. In our model it is 63 kHz. The decimation is done with selecting every D-th sample by the block *Lookup Table.2*. The output signal of this block has sampling frequency $f_{vz\ sub}$, in our simulations 128 kHz.

The subcarrier demodulator is done with a common Costas loop. Blocks *Digital Filter Design.7* and *Digital*

Filter Design.8 are first-order lowpass filters. Single sided bandwidths of these filters meet the condition

$$B_{3dB}\{DFD.7\} = B_{3dB}\{DFD.8\} >\approx \frac{R_b.(1+\alpha)}{2}. \quad (9)$$

The symbol ">≈" means "slightly wider", equality in (9) namely leads to additional losses due to ISI. At the output of the subcarrier Costas loop, the signal is in the baseband.

In the next step, the demodulated signal is filtered with the block *Square root Raised Cosine Receive Filter.1*. It is a complementary filter to *Square root Raised Cosine Transmit Filter*. The roll-off factor α for the receive filter must be the same as in the case of the transmit filter. Undistorted output data from the transmit filter are delayed in the block *Delay.3* and then filtered by *Square root Raised Cosine Receive Filter.2*. The length of delay in block *Delay.3* is set to match the delay inside the communication system. So, the timing (phase delay) of symbols on outputs of *Square root Raised Cosine Receive Filter.1* and *Square root Raised Cosine Receive Filter.2* is set with *Delay.3* to be in good match. In the described model the optimal value of delay is 55 samples.

Next, the symbols are restored with *Sample and Hold.1* and *Sample and Hold.2* blocks. Timing for the symbol sampling is controlled by the block *Pulse Generator*. The control pulse has one-sample width. The period of control pulses corresponds to the symbol period, $T_b = 1/R_b$. Phase of control pulses define the instant time of sampling. It is set according to the eye diagram. After the symbol restoration, the sampling frequency matches the symbol rate.

In our model of the communication system the Costas loops are used. The Costas loop has 180 degrees phase ambiguity. This phenomenon can lead to randomly occurring inversion of received data. To eliminate this problem it is necessary to apply the differential coding principle as it is discussed in our paper [3].

The differential coding principle for received data is involved in the model with blocks *Relay.1*, *Windowed Integrator.1* and *Abs.1*. Block *Relay.1* works as a Schmitt trigger converting sampled data to levels ±1 with zero-level threshold. Thanks to this conversion, the two consecutive sample integration in the block *Windowed Integrator.1* leads to level -2, 0 or +2 in each step of simulation. After the calculation of the absolute value in the block *Abs.1* the output signal has level +2 or 0. Differential coding for transmitted data works on the same principle and is done with blocks *Relay.2*, *Windowed Integrator.2* and *Abs.2*.

It can be shown that only one single error bit in data transmission causes two errors after differential coding, and a burst with two or more error bits in data transmission (corresponding to data inversion due to Costas loop phase ambiguity) causes just two errors after differential coding.

The bit error rate for differentially coded data stream is calculated with block *Error Rate Calculation*. In this block, the user can define the computation delay. Typically, this delay is used to provide some time for Costas loops settling after the start of simulation.

The relationship between the differentially coded BPSK bit error rate, BER_1, and uncoded BPSK bit error rate, BER, is given by

$$BER = \frac{1 - \sqrt{1 - 2.BER_1}}{2}. \quad (10)$$

Block *Display* shows the actual number of transmitted bits, error bits and bit error rate. Block *Signal to Workspace* provides those variables in the vector "BER" which can be read from Matlab environment and be used for example to control the run of simulator.

3. Phase Difference Calculator

The model of communication system described in Section 2 allows BER calculation only. It is a basic indicator of quality of data transmission, but does not say anything specific about the function of individual components of the communication system.

Very important parts of the modeled communication system are Costas loops. The reliability of the Costas loop can be determined by (statistic) value of tracking phase error in the presence of disturbances. For this purpose it is necessary to measure the phase difference between the loop's VCO output signal and a useful input signal of the Costas loop. The block diagram of the developed phase difference calculator is presented in Fig. 2.

Input harmonic signals of the phase difference calculator are connected to ports *Signal1* and *Signal2*. In our case, *Signal1* is output signal from model block *Discrete-Time VCO.1* or *Discrete-Time VCO.2*, and *Signal2* is auxiliary harmonic signal with the same parameters as output signal of block *Sine Wave.1* or *Sine Wave.2*. Instant samples of *Signal1* and *Signal2* are transferred as **t1** and **t2** to *Embedded MATLAB Function* "fcn". For both these signals, their first derivations are also determined with blocks *Discrete Derivative.1* and *Discrete Derivative.2*. Results of derivations are transferred as **d1** and **d2** to *Embedded MATLAB Function* "fcn".

In the block *Embedded MATLAB Function* "fcn", the instant phase of both signals *Signal1* and *Signal2* is computed. First, with the instant sample and the result of the first derivation the square is determined. Then the phase of the signal is computed with the goniometric function arcsin. The results of computation are true phases in the range < 0 ; 2π) rad for both signals, *Signal1* and *Signal2*. Next the phase difference is determined in the default range < 0 ; 2π) rad or < 0 ; π) rad for subsequent computation of the phase variance. The value of the phase difference is then exported on port *Phase difference*.

The value of the actual phase difference is then poised to block *Relay* and here compared with π. The output value

of block *Relay* is 0 for the phase difference in the range $< 0 ; \pi\text{-eps} >$ and 1 for the phase difference in the range $< \pi\text{+eps} ; 2\pi >$. Here, *eps* is hardware-dependent value of Matlab numerical accuracy (eps ≈ 2.22e-16 in our case). If the phase slip occurs (i.e. the phase difference just exceeds the border value π between stable points at $\pi/2$ and $3\pi/2$), then the output of block *Relay* is changed from 0 to 1 or vice versa. In the subsequent block, *Logical Operator XOR*, the actual value of *Relay* output is compared with the value from the previous step of simulation, which is stored in the block *Unit Delay1*. If the phase slip occurs, there is the one simulation step long value 1 at the output of block *Logical Operator XOR*.
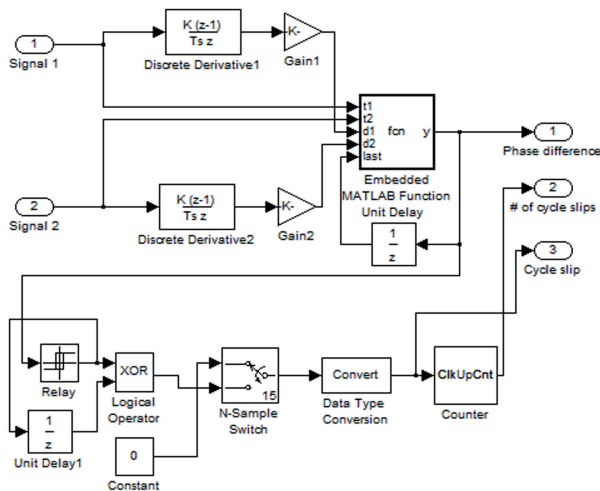
Fig. 2.   The block diagram of the phase difference calculator.

The block *N-Sample Switch* is used for elimination of undesired counting at the start of simulation. At the output of this block there is 0 for *N* first steps of the simulation. After *N* steps of simulation, *N* is user-definable (usually after the initial loop settling), the block *N-Sample Switch* starts to transfer the output value of block *Logical Operator XOR*. In the block *Data Type Conversion* the boolean to real data type is converted. If the phase slip occurs, there is the one simulation step long impulse at the output of block *Data Type Conversion*. This impulse is then exported on the port *Cycle slip*.

Block *Counter* counts impulses on the output of block *Data Type Conversion*. The output of block *Counter* is exported on port *# of cycle slips*.

4.   Design of Costas Loops

The design of the Costas loop means primarily the determination of parameters for crucial blocks of Costas loop - loop filter *Digital Filter Design.5 (...9)* and loop VCO *Discrete-Time VCO.1 (...2)*. The design can be carried out with regard to dynamic properties of the resulting loop. It generally means that for the desired time of loop settling t_{set} and the desired damping factor ξ we search for

an unknown loop filter cut-off frequency ω_{cA} and VCO's input sensitivity K_{VCO}.

The universal process of the design of the Costas loop can be done as follows. Firstly, with the goniometric approach as described in [6] or [4] in more detail, the phase detector response should be determined. From this response, the Costas loop gain A in the steady state can be obtained. In our case, with the gain of phase detectors $k_m = 1/2$, it is

$$A\{\text{sub}\} = \frac{1}{4}.k_m \quad \text{for subcarrier loop,}$$
$$A\{\text{carr}\} = \frac{1}{8}.k_m \quad \text{for carrier loop.}$$
(11,12)

Next, with the equivalent model of the loop, the transmission function of the loop $H_{sm}(s)$ can be determined, as described for example in [7]. The result is

$$H_{sm}(s) = \frac{2.A.H_{LP}(s).H_{VCO}(s)}{1 + 2.A.H_{LP}(s).H_{VCO}(s)}$$
(13)

where $H_{LP}(s)$ is the transfer function of the loop filter and $H_{VCO}(s)$ is the transfer function of the loop's VCO. Equation (13) can be converted from $s = j.2.\pi.f$ to $z = exp(j.2.\pi.f.T_{vz})$ plane [8] as

$$H_{sm\,BUT}(z) = \frac{K_N.z + K_N}{z^2 + (K_N - a - 1).z + (K_N + a)}$$
(14)

where $K_N = 2.A.K_{LP}.K_{VCO}.T_{VZ}$. Here T_{vz} is the sampling period and a and K_{LP} is given from the bilinear transformation of 1-st order lowpass Butterworth filter as

$$a = \frac{2 - \omega_{cA}.T_{vz}}{2 + \omega_{cA}.T_{vz}} \quad , \quad K_{LP} = \frac{1 - a}{2}.$$
(15,16)

Here, ω_{cA} is the angular cut-off frequency of the loop filter. Now, it is possible to compare characteristic polynomial of (14) and characteristic polynomial of regular dynamic system of second order to each other. And next, with the method of the impulse invariance to transform characteristic polynomial of regular dynamic system to z plane, in order to derive relationships between a, K_{LP} and T_{vz}, ξ, ω_n (or t_{set}) as described in detail in [9]. The result formulas are

$$a = -\frac{1}{2} + \exp(-\xi.\omega_n.T_{vz}). \quad ...$$
$$... \quad \cos(\omega_n.T_{vz}.\sqrt{1 - \xi^2}) + \frac{1}{2}.\exp(-2.\xi.\omega_n.T_{vz})$$
(17)

and

$$K_N = \exp(-2.\xi.\omega_n.T_{vz})$$
(18)

where the natural frequency of the loop is given by

$$\omega_n = \frac{4}{\xi.t_{set}}.$$
(19)

The Costas loop design is done as follows: For the desired values of t_{set} and ξ, the value of ω_n using (19) is computed. Next, with the help of (17) and (18), the values of a and K_N are determined for T_{vz} used in the modeled loop. The gain of the loop filter, K_{LP}, is then given by (16), and the angular cut-off frequency of the loop filter, ω_{cA} [rad/s], can be computed using (15). Finally, sensitivity of the VCO, K_{VCO} [rad/s/V], can be determined from the relation $K_N = 2.A.K_{LP}.K_{VCO}.T_{VZ}$, where the subcarrier or carrier Costas loop gain in the steady state A is defined by (11,12).

5. Results

In Fig. 3 and 4, the simulation results for the communication system model described in Section 2 are presented. In these figures, BER is the uncoded bit error rate, as given by (10). Simulation is done for the receiver's local oscillator phase noise $(1/f)$ PN = -40 dBc@100 Hz. The damping factor for both - carrier and subcarrier - Costas loops in those simulations is $\xi_{sub} = \xi_{carr} = 0.7$, and the value of loop settling time t_{set} for the certain loop is changed.

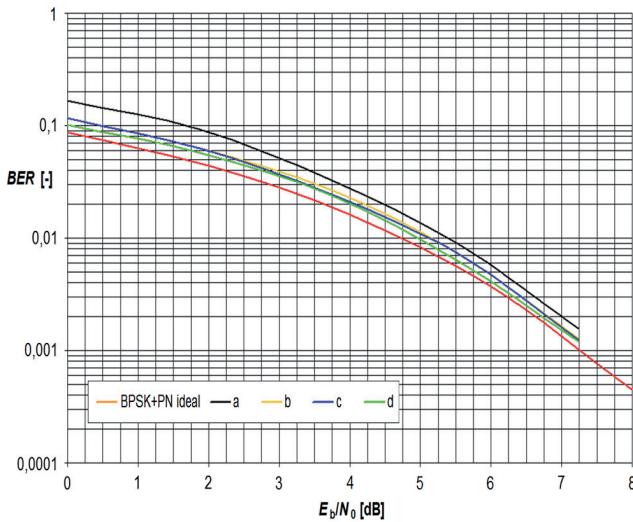**Fig. 3.** Comparison of the theoretical $BER = f(E_b/N_0)$ for phase noise $(1/f)$ PN = -40 dBc@100 Hz with:
a - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 0.05$ s, $t_{set\ carr} = 0.005$ s,
b - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 0.05$ s, $t_{set\ carr} = 0.05$ s,
c - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 0.05$ s, $t_{set\ carr} = 0.5$ s,
d - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 0.05$ s, $t_{set\ carr} = 5$ s.

Thanks to the phase difference calculator described in Section 3, it is possible to distinguish the errors due to cycle slips. When CS is the number of cycle slips, and E is the count of all errors for differentially coded data stream, then the relative value of the error due to phase slips is

$$\frac{BER_1\{csl\}}{BER_1} = \frac{CS}{\dfrac{E - CS}{2} + CS}. \qquad (20)$$

The dependence of this parameter for the subcarrier loop with various settling time $t_{set\ sub}$ is presented in Fig. 5.
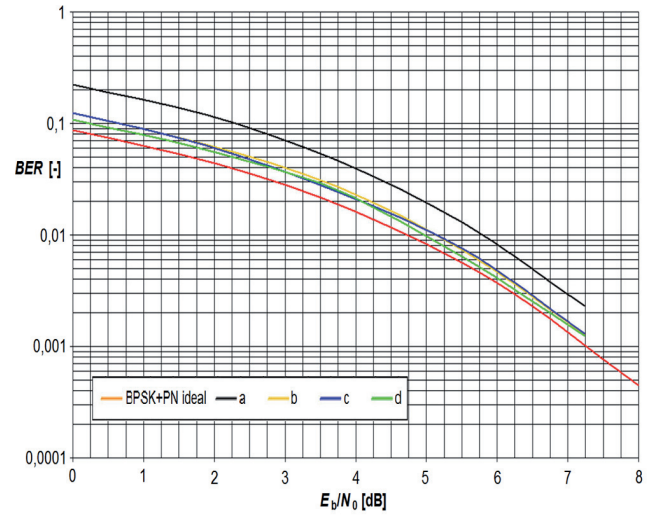

Fig. 4. Comparison of the theoretical $BER = f(E_b/N_0)$ for phase noise $(1/f)$ PN = -40 dBc@100 Hz with:
a - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 0.005$ s, $t_{set\ carr} = 0.05$ s,
b - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 0.05$ s, $t_{set\ carr} = 0.05$ s,
c - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 0.5$ s, $t_{set\ carr} = 0.05$ s,
d - $\xi_{sub,carr} = 0.7$, $t_{set\ sub} = 5$ s, $t_{set\ carr} = 0.05$ s.

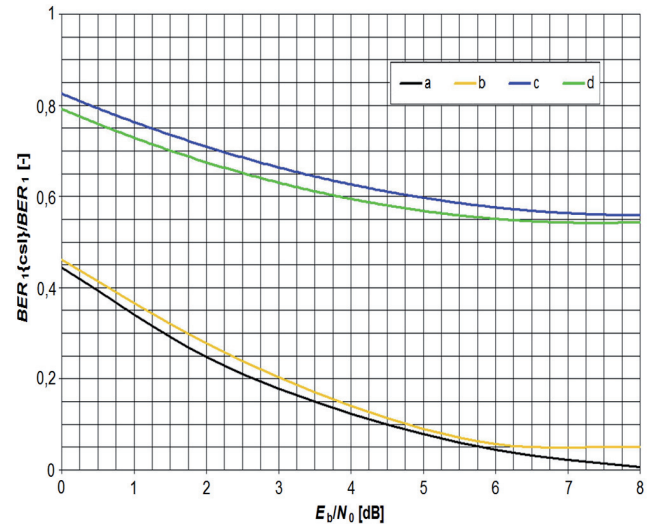**Fig. 5.** Dependence of $BER_1\{csl\}/BER_1$ on E_b/N_0 for AWGN only and AWGN with $(1/f)$ PN = -40 dBc@100 Hz:
a - $\xi_{sub} = 0.7$, $t_{set\ sub} = 0.005$ s – AWGN only,
b - $\xi_{sub} = 0.7$, $t_{set\ sub} = 0.005$ s – AWGN + PN,
c - $\xi_{sub} = 0.5$, $t_{set\ sub} = 0.005$ s – AWGN only,
d - $\xi_{sub} = 0.5$, $t_{set\ sub} = 0.005$ s – AWGN + PN.

The instant simulation time by the cycle slip event can be stored. After that, the statistical evaluation of the data set can be done. In Fig. 6, the dependence of average time between consecutive cycle slips on E_b/N_0 is presented. Simulations were performed for subcarrier loop with vari-

ous values of damping factor ξ_{sub} and for scenario with and without local oscillator phase noise $(1/f)$ PN = -40 dBc@100 Hz.
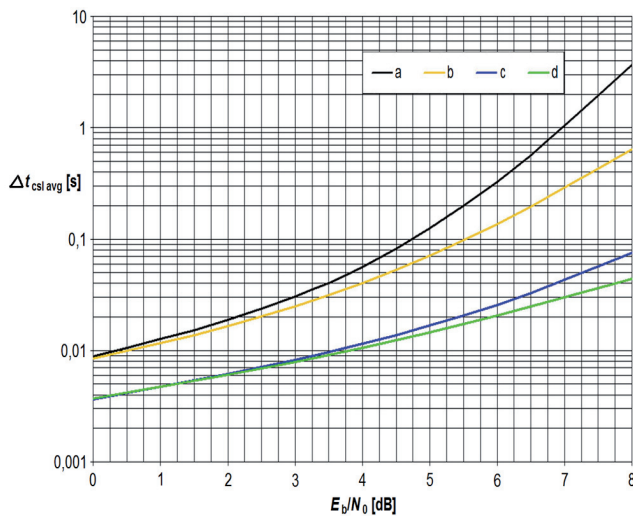

Fig. 6. Dependence of average time between cycle slips on E_b/N_0 for AWGN only and AWGN with $(1/f)$ PN = -40 dBc@100 Hz:
a - $\xi_{sub} = 0.4$, $t_{set\ sub} = 0.01$ s – AWGN only,
b - $\xi_{sub} = 0.4$, $t_{set\ sub} = 0.01$ s – AWGN + PN,
c - $\xi_{sub} = 0.3$, $t_{set\ sub} = 0.01$ s - AWGN only,
d - $\xi_{sub} = 0.3$, $t_{set\ sub} = 0.01$ s – AWGN + PN.

6. Conclusion

In this paper the model of low rate telemetry communication system for Matlab – Simulink is described. The model can be very easily modified and extended. Thanks to Matlab environment there is a variety of tools for processing results. In this paper, only a few of achievable results is presented.

Acknowledgements

References

[1] KINMAN, P.W. 34-m and 70-m telemetry reception. *DSMS Telecommunications Link Design Handbook,* 2003, [Online] Cited 2012-03-02. Available at: http://deepspace.jpl.nasa.gov/dsndocs/810-005/207/207A.pdf

[2] LINDSEY, W. C., SIMON, M. K. *Telecommunication Systems Engineering*. New York: Dover Publications, Inc., 1973

[3] ŠPAČEK, J., KASAL, M. The low rate telemetry transmission simulator. *Radioengineering,* 2007, vol. 16, no. 4, p. 24 - 32.

[4] ŠPAČEK, J. *Technology of Receiving of Very Weak Signals from Interplanetary Spacecrafts*. Doctoral thesis, Brno University of Technology, 2011. (In Czech) .

[5] BAR, A. *Phase Noise - Model for MATLAB*. [Online] Cited 2011-12-28. Available at: http://www.mathworks.de/matlabcentral/fileexchange/8844-phase-noise

[6] FEIGIN, J. *Practical Costas Loop Design*. [Online] Cited 2011-12-28. Available at: http://images.rfdesign.com/files/4/0102Feigin20.pdf

[7] SÝKORA, J. *Digital Radio Communication II*. Lecture notes, Czech Technical University in Prague, Prague, 1995. (In Czech).

[8] ŠEBESTA, J. *Digital Satellite Detection of PSK Signals Associated with the Phase Estimation and Symbol Timing*. Doctoral thesis, Brno University of Technology, 2005. (In Czech).

[9] LI, W., MEINERS, J. *Introduction to Phase-Locked Loop System Modeling*. [Online] Cited 2011-12-28. Available at: http://www.ti.com/lit/an/slyt169/slyt169.pdf

About Authors ...

Jiří ŠPAČEK was born in Zlín, Czech Republic, in 1981. He received his master's degree in Electrical Engineering from the Brno University of Technology (BUT) in 2002. At present, he works for the Czech Telecommunication Office at manned Monitoring Station Karlovice as a measurement technician. He is also still a distance PhD student at the Dept. of Radio Electronics, BUT. His research is focused on the technology for extremely weak deep space probes signals reception.

Miroslav KASAL (born in 1947 in Litomyšl, Czech Republic) graduated in Communication Engineering from the Faculty of Electrical Engineering, BUT, in 1970. In 1984 he obtained his PhD degree in metering engineering. He was the head of the NMR Department and Electronics Laboratory of the Institute of Scientific Instruments, Academy of Science of the Czech Republic (1991 - 2002). Since 2002 he has been with the Dept. of Radio Engineering, BUT, as a professor. Dr. Kasal is a senior member of the IEEE. In 2004 he received the Award of the Rector of the Brno University of Technology and the SIEMENS Prize for research. In 2007 prof. Kasal received the Prize for research of the Minister of Education of the Czech Republic.