


Article

Performance Evaluation of Power-Line Communication Systems for LIN-Bus Based Data Transmission

Martin Brandl *  and Karlheinz Kellner

Department of Integrated Sensor Systems, Danube University, 3500 Krems, Austria;
karl-heinz.kellner@donau-uni.ac.at

* Correspondence: martin.brandl@donau-uni.ac.at; Tel.: +43-2732-893-2790

Abstract: Powerline communication (PLC) is a versatile method that uses existing infrastructure such as power cables for data transmission. This makes PLC an alternative and cost-effective technology for the transmission of sensor and actuator data by making dual use of the power line and avoiding the need for other communication solutions; such as wireless radio frequency communication. A PLC modem using DSSS (direct sequence spread spectrum) for reliable LIN-bus based data transmission has been developed for automotive applications. Due to the almost complete system implementation in a low power microcontroller; the component cost could be radically reduced which is a necessary requirement for automotive applications. For performance evaluation the DSSS modem was compared to two commercial PLC systems. The DSSS and one of the commercial PLC systems were designed as a direct conversion receiver; the other commercial module uses a superheterodyne architecture. The performance of the systems was tested under the influence of narrowband interference and additive Gaussian noise added to the transmission channel. It was found that the performance of the DSSS modem against singleton interference is better than that of commercial PLC transceivers by at least the processing gain. The performance of the DSSS modem was at least 6 dB better than the other modules tested under the influence of the additive white Gaussian noise on the transmission channel at data rates of 19.2 kB/s.

Keywords: power line communication; local interconnect network (LIN); direct sequence spread spectrum; automotive sensor network



Citation: Brandl, M.; Kellner, K. Performance Evaluation of Power-Line Communication Systems for LIN-Bus Based Data Transmission. *Electronics* **2021**, *10*, 85. <https://doi.org/10.3390/electronics10010085>

Received: 30 November 2020

Accepted: 28 December 2020

Published: 4 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Power line communication (PLC) is a versatile method using existing infrastructure like power cables for data transmission. This makes PLC an alternative and cost-efficient technology by the dual use of the powerline to other communication solutions like wireless radio frequency communication. The PLC modems can be installed plug and play on the existing power infrastructure without the need for other costly installation procedures. Typically the rooms on each floor and beyond are linked by the same power line infrastructure which can be used easily for PLC. However, the PLC transmission channel is a very rough environment and need special attention to the design of the PLC [1–3]. PLC technology is a good alternative to wireless communication and is intended for use such as home automation, smart grid, city automation and cyber-physical systems as well as long-range communication. Another approaching field for PLC systems is given by the application in vehicular-, ship-, and aircraft- technology [4–14].

Especially in vehicular technology CAN (Controller Area Network, ISO 11898) and LIN (Local Interconnect Network, ISO 17987 [15,16]) buses are used for communication links between different components. The CAN bus offers data rates up to 1 Mbit/s (high speed CAN) and the hard wiring between the CAN transceivers allows a lossless bitwise arbitration method which makes CAN suitable as a real-time prioritized communication system. This procedure also provides a hierarchy of messages among each other. The message with

the lowest identifier may always be transmitted. For the transmission of time-critical messages, a high-priority identifier can be assigned in order to give the message priority during transmission. In vehicular technology LIN buses are typically used for low-speed and low priority communication between sub-systems. For cost-sensitive and low speed applications mainly the LIN bus is used in automotive applications. The LIN bus serves as a sub-bus for cost-effective networking of control devices in cars. LIN is used for simple sensor and actuator applications where the CAN bus was not an option for this cost-sensitive area, as the bandwidth and versatility are not required and the costs for networking with this bus are too high. Up to 16 bus subscribers can be networked in a LIN cluster, with a maximum data rate of 20 kbit/s. The LIN bus is typically used for non-critical safety components and sensors/actuators in cars where the application area includes comfort functions, such as window regulators, central locking, electric mirror adjustment, electric seat adjustment, rain sensor, light sensor, electric sunroof, or air conditioning control etc.

The LIN-Bus

With the demand for a cost-effective communication system for use in the sensor/actuator area, LIN found its way into the vehicle as a so-called “sub-bus”. LIN communication was very simple and does not require a communication controller or crystal oscillator for synchronization. The data transmission takes place via a single line (single wire). In order to meet the interference immunity requirements typical for motor vehicles, the transmission levels correspond to the battery voltage but with slew rates below $2\text{ V}/\mu\text{s}$ therefore limiting the symbol transmission rate to 20 kbit/s. The master-slave architecture makes LIN communication deterministic. A master controls the entire communication in a LIN cluster on the basis of a well-defined time table (Figure 1). One LIN frame consists of the two parts header and response. The header is always sent by the LIN master, while the response is sent by either one dedicated LIN slave or the LIN master itself. A LIN slave addressed by the message responds with a frame which can consist of up to eight data bytes. Due to the six-bit wide identifier up to 64 LIN messages can be specified. In contrast to CAN, LIN is a centrally controlled message distribution system. A LIN message always begins with the so-called “break-field”. This helps all LIN slaves to synchronize and is used to activate all attached LIN slaves to listen to the following parts of the header. The “break-field” command consists of one start bit and several dominant bits with a length of at least 11-bit times. Data integrity is checked using parity bits in the message header and a checksum transmitted in the message response. In this study we focused on the replacement of the LIN bus by PLC systems for vehicular applications. Especially for low speed and low priority bus systems like the LIN bus, PLC modems can replace the wired interconnections between the sub-systems.

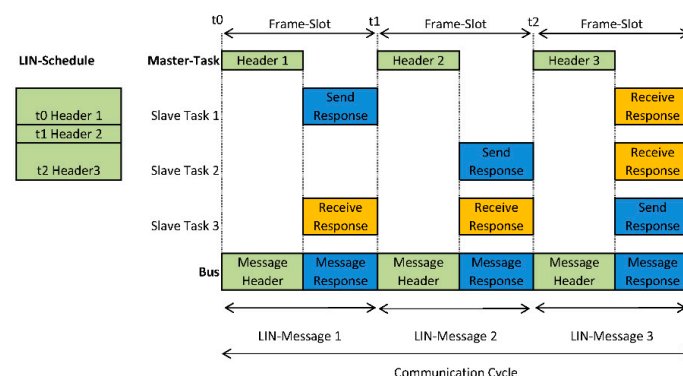


Figure 1. LIN schedule.

A further advantage is that the wiring of additional components or device upgrades is much easier, since only the power line and no additional LIN bus cables are required for

data communication. After power connection of the new component, it logs on via the PLC at the master unit and immediately becomes part of the LIN network.

2. Performance of PLC Systems for LIN Data Transmission

The goal of our development was to design a robust communication system for automotive applications and to radically reduce component costs to less than 1 Euro. For applications in the automotive industry, the cost of the components is of high importance and can be kept very low by almost complete system implementation in a low-cost microcontroller. Direct sequence spread spectrum (DSSS) is a promising technology for interference-resistant PLC data transmission, and because of its low computational power requirements, it can be easily implemented in a low-performance microcontroller. For evaluation of our cost optimized DSSS PLC system (Figure 2c), it was compared to two commercial PLC modules #1 and #2 (Figure 2a,b) which are dedicated for automotive use. The modules were analyzed regarding their sensitivity to single-tone interferers and additive white Gaussian noise (AWGN) on the transmission channel, as well as compared with theoretical results. Classical PLC modules which are used for low-voltage household networks (e.g., Homeplug-Standard IEEE-1901) were not used within this study because they are not qualified for automotive applications.

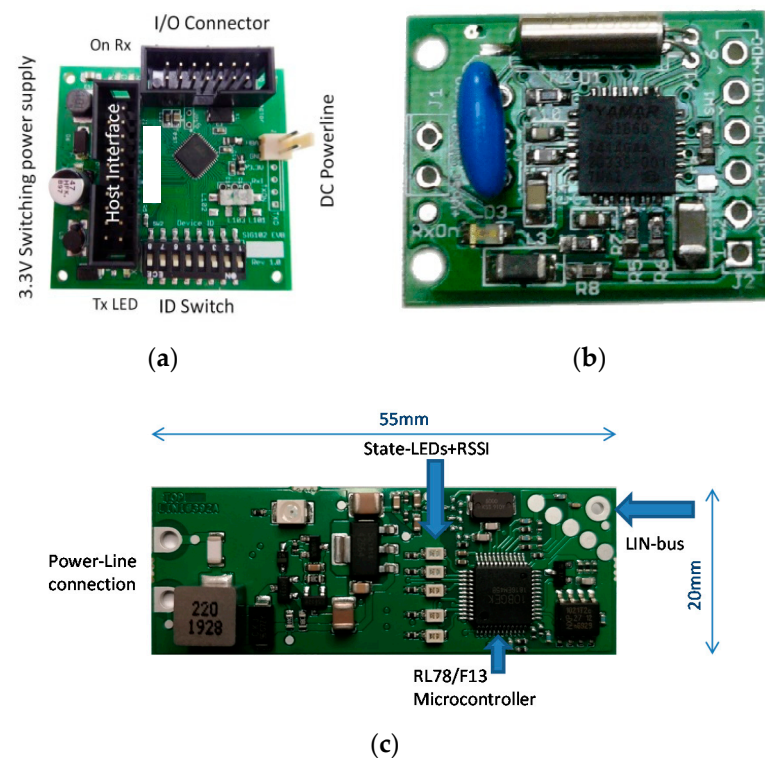


Figure 2. (a,b) Evaluation boards of the commercial PLC modules #1 and #2; (c) board of the self-designed DSSS modem.

DSSS is a well-known and established data transmission method for robust transmission over distorted channels and is often used because of its easy implementation [17–24]. In spread spectrum technology, the signal to be transmitted is spread over a wide frequency range, with spreading being achieved by modulating the carrier frequency with a pseudostatistical spreading code. During spreading, the bits of the user data are spread with the spreading sequence.

This spreading results in a high-bit-rate code sequence, which is called a chip sequence to distinguish it from the bit sequence. The “chips” represent the smallest spread information units. The chip rate depends on the spreading factor which corresponds to the quotient of the chip rate to the bit rate. The wider the spread, the less susceptible the transmission is

to interference from other signals. At the same time, spreading provides a certain degree of protection against unauthorized eavesdropping.

All modules operate with carrier modulation, where the carrier frequency is adjustable within a certain range. The carrier frequency of the DSSS modem is free adjustable but typically set to 8 MHz and depend on the clock-oscillator crystal frequency ($f_{carrier} = f_{crystal}/4$). Both commercial modules can operate on different carrier frequencies in the range from 1.75 MHz to 13 MHz in coarse steps (module #1) and from 5 MHz to 30 MHz with a spacing of 100 kHz for module #2. All modems are able to transfer the LIN bus specific data rate of 19.2 kbit/s. In our DSSS system an $N = 7$ bit Barker code is used as spreading function which is modulo 2 added to the binary data sequence where N is the length of the sequence. Each data bit is spread by a full 7 bit Barker sequence and therefore the chip duration T_c is T_b/N of the bit duration T_b (Figure 3a). An overview about the proposed DSSS modem and first results were also given [25]. The used Barker code has an optimal autocorrelation function which means that the autocorrelation peak has an amplitude of N and the side-lobes are no larger than 1 (Figure 3b) [26]. An important performance parameter of spread spectrum systems is the processing gain PG which is defined as the ratio of the signal bandwidth B_s on the transmission channel after spreading in relation to the message signal bandwidth B_m defined by the symbol rate Equation (1).

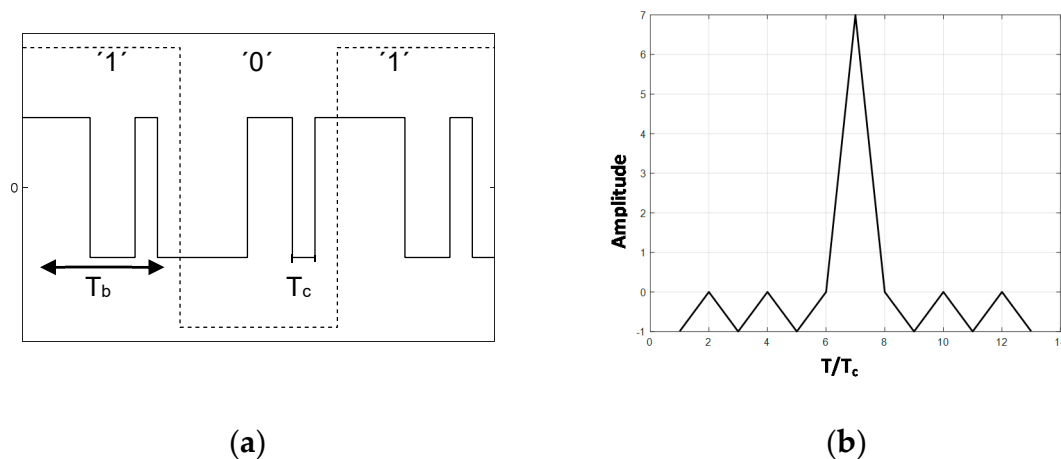


Figure 3. (a) Generation of a DS spread spectrum signal (dashed line: data bits, full line: spreading function modulo 2 added to the data sequence); (b) autocorrelation function of an $N = 7$ bit Barker code.

If BPSK modulation is used, the processing gain for the proposed barker code with length $N = 7$ is $PG = 7$ or 8.5 dB:

$$PG = \frac{B_s}{B_m} = \frac{T_b}{T_c} = \frac{R_c}{R_b} = N = 7 \rightarrow 8.5 \text{ dB} \quad (1)$$

A simplified block structure of the DSSS transceiver is given in Figure 4a. The LIN bus data stream with a data rate of $R_b = 19.2$ kB/s is modulo 2 added to the $N = 7$ Barker code. This means that the clock rate of the Barker code generator is $R_C = N \cdot R_b$. For transmission over the powerline the spread message signal is DE-BPSK (differential encoded binary phase-shift keying) [27] modulated on an 8 MHz carrier by a balanced modulator. Incoherent detection, the receiver obtains its demodulation frequency and phase references using a carrier synchronization loop.

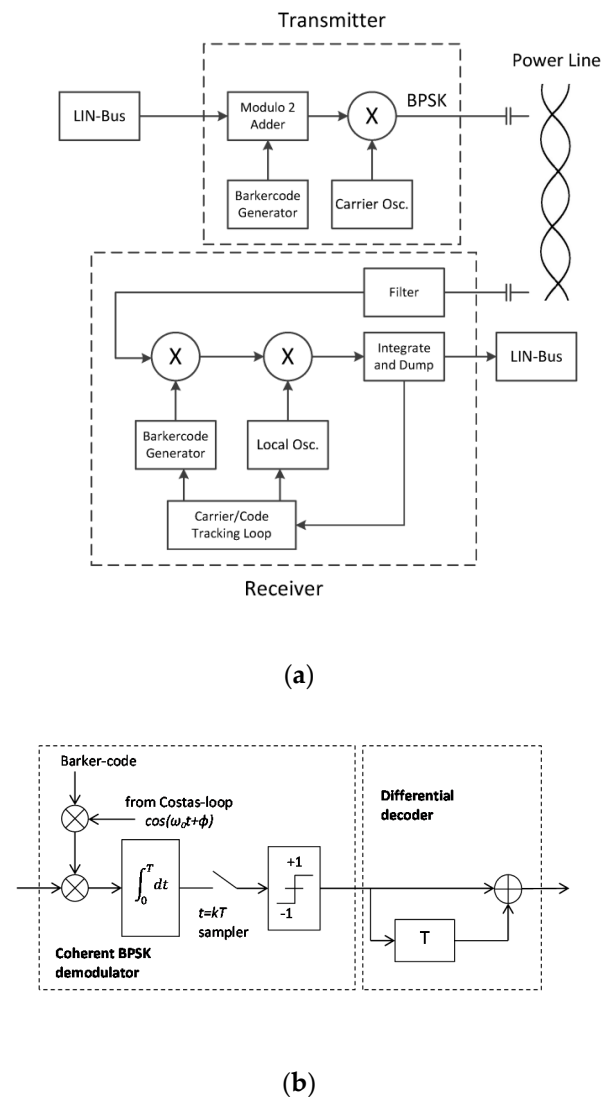


Figure 4. (a) Block diagram of the proposed DSSS transceiver; (b) detailed structure of the DE-BPSK receiver.

Such synchronization circuits can lead to a phase ambiguity in the detected phase, which can lead to wrong decisions in the demodulated bits. For example, the Costas loop has a phase ambiguity of integer multiples of pi-radians at the locking points. As a result, carrier recovery may lock into pi-radians out of phase, resulting in a situation where all detected bits are completely inverted compared to the bits at perfect carrier synchronization. Phase ambiguity can be efficiently addressed by applying differential coding at the input of the BPSK modulator and by performing differential decoding at the output of the coherent demodulator at the receiver side (Figure 4b). Via capacitive coupling the modulated signal is fed onto the power line. In the receiver, the despreading is performed by synchronous multiplication of the received signal with a replica of the Barker code sequence as used in the transmitter. After despreading the bandpass signal is transferred into the baseband by synchronous demodulation. The carrier frequency synchronization and the time synchronization of the despreading code generator are done by carrier/code tracking loops. As noted previously, one of the advantages of spread spectrum is the ability to reject interferences which is very useful especially in a heavy electromagnetic polluted environment like automotive applications. If the interference is within the passband of the receiver input filter, the jammer will be spread by the Barker code sequence to the bandwidth B_s of the spreading function. Therefore the power of the interfering signal P_j is reduced by

an amount called the bandwidth expansion factor B_s/B_m which is equal to the processing gain PG [28]. The total power J_0 of the interference at the output of the receiver is:

$$J_0 = P_J \frac{B_s}{B_m} = \frac{P_J}{PG} \quad (2)$$

The reduction of interference power by the processing gain PG is one of the main advantages using spread spectrum transmission systems in interference polluted environments like automotive applications.

3. Test Setup and Results

The test setup for characterizing the robustness of the different transceivers against single-tone interference and AWGN on the transmission channel is shown in Figure 5.

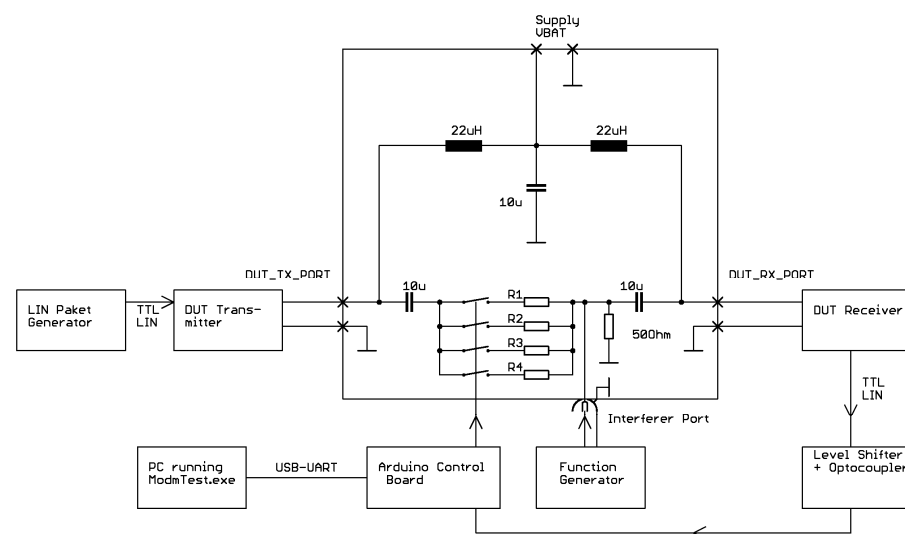
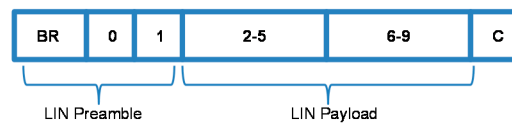


Figure 5. Test setup for all transceivers.

The transmission channel was realized by a switchable resistive damping circuit where different attenuation factors can be adjusted for varying the signal to interference ratio (SIR) and the signal to noise ratio (SNR) on the receiver input. For electrical decoupling and avoidance of crosstalk between the transmitter and the receiver separated power supplies and optocouplers are used. Predefined LIN-packets are delivered from a LIN-packet generator.

The transmitted LIN-packets are designed to enable an asynchronous data transmission and packet error detection by transmitting an incrementing counter value in the payload and a checksum byte (Figure 6) A LIN-packet error was detected if a counter error between two subsequent received packets or a CRC error occurs. LIN-packets were transmitted every 10 ms with data rate of $R_b = 19.2$ kB/s. The LIN-packet error rate (PER) was calculated from the number of faulty or missing packets output by the receiver to the number of packets sent.



BR:	Break signal (minimum 13-bit low)
Byte 0:	Sync message 0x55 h
Byte 1:	LIN Protected Identifier 0x08 h
Byte 2..5:	Data Byte 0..3 = 32-bit; packet counter
Byte 6..9:	Data Byte 4..7 = 32-bit; copy of the packet counter
C:	LIN checksum byte. For each transmitted byte (B0→C) one start of the byte and two end of the byte bits are used (LIN compliance)

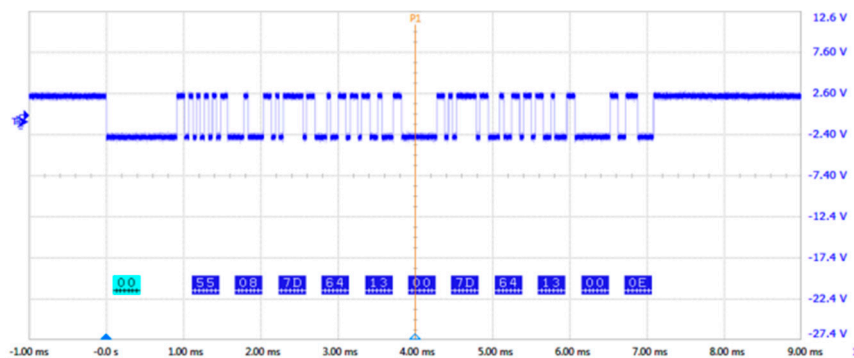


Figure 6. Configuration and waveform of the transmitted LIN-packets.

3.1. Sensitivity to Single-Tone Interference

The PLC data communication in industrial or automotive environments is frequently corrupted by interferences based on harmonics from, e.g., switched power supplies and other electronic components coupling high-frequency radiation into power lines. To investigate the influence of narrowband interference on the quality of LIN data transmission, a sinusoidal interference signal was superimposed on the PLC signal on the transmission channel. A sine wave generated by a function generator (33250A, Keysight, Santa Rosa, CA, USA) was coupled to the transmission line and swept over a frequency band $\Delta f = \pm 200$ kHz around the PLC carrier center frequency. The SIR was calculated from the RMS amplitude of the PLC signal and the RMS amplitude of the interferer on the transmission channel. Depending on the design of the PLC transceivers, the sensitivity to interference depends on the absolute amplitude of the received PLC signal which can be caused by amplifier saturation, amplifier slew rate, etc.

The SIR was also measured in dependency of the PLC signal attenuation over the transmission channel. All PLC modules show a threshold sensitivity in *PER* to the amplitude of the single-tone interferer where the *PER* rises in a very close interval from low values to 1 (Figure 7). The differences of the evaluated PLC modules in interference sensitivity are mainly based on the used filters for interference rejection and the spreading gain for DSSS systems. In the DSSS module a 2nd-order LC bandpass (BP) filter at the PLC center frequency is used in front of the receiver and a 10 kHz 2nd-order active lowpass (LP) filter builds a signal matched filter in front of the ADC [25]. Module #1 and module #2 PLC transceivers are using commercial high-Q ceramic BP-filters where module #1 is operating

with a 6.5 MHz \pm 80 kHz frontend BP-filter (#773–0065, Oscilent Corporation, Irvine, CA, USA) and module #2 with a 10.7 MHz \pm 165 kHz IF-filter (#762-0107-A20, Oscilent Corporation, Irvine, CA, USA) in the standard configuration, respectively. Module #2 is an evolution of module #1 and uses instead of a direct conversion receiver which needs for each adjusted PLC center frequency a different ceramic frontend filter, a superhet receiver with a fixed intermediate frequency (IF) at 10.7 MHz with the advantage that the PLC carrier frequency can be adjusted independently of the filter center frequency.

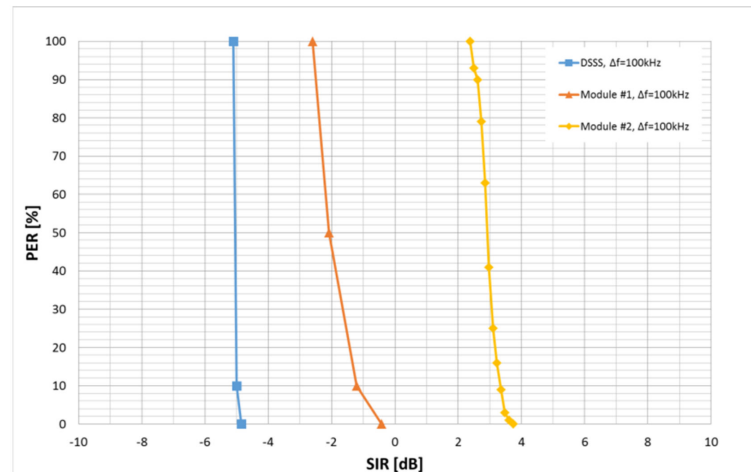


Figure 7. Sensitivity of the packet error rate (*PER*) to the *SIR*, which shows a distinct threshold effect for all different transceivers.

In Figure 8, the *SIR* threshold level, where no further data transmission was possible (*PER* = 1), of the different PLC modules against single-tone interference is depicted. The single-tone interferer was a sine wave generated by a function generator at different frequency offsets to the transmission center frequency $\Delta f = f_{carrier} - f_{interferer}$ and a *SIR* adjusted by different channel attenuations of the PLC signal. Theoretically, the robustness of a DSSS system against single-tone interference must be by the process-gain *PG* better than that of a conventional receiver.

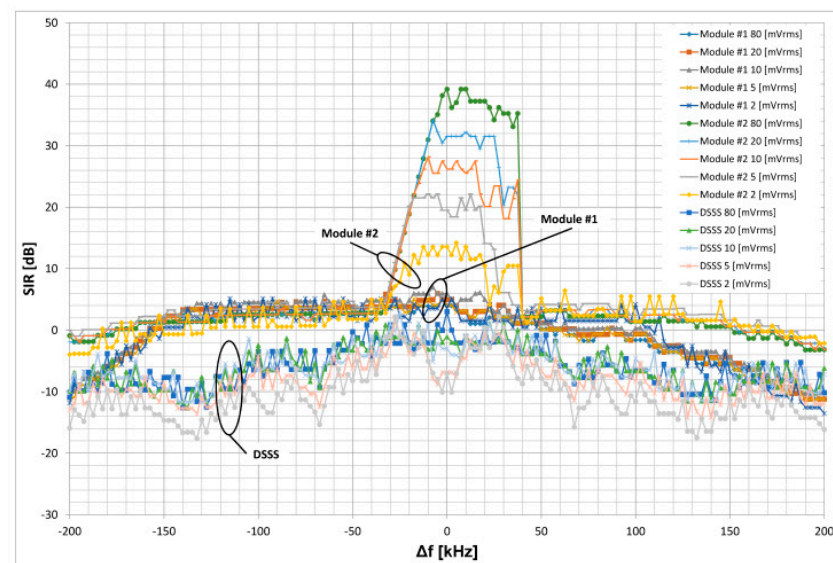


Figure 8. Sensitivity to single tone interference with a frequency offset Δf to the PLC carrier frequency.

As expected, the DSSS module had an on average 7 dB better *SIR* performance ($SIR_{average} \pm 80 \text{ kHz} = -4 \text{ dB}$) within the signal bandwidth $f_c \pm B_s/2 = f_c \pm 70 \text{ kHz}$.

The DSSS process gain is as shown in Figure 8 in dependency of the received signal amplitude because of the nonlinear characteristic of the pre-amplifier where for increasing signal amplitudes the gain decreases [25]. Therefore, the best performance was found for low signal amplitudes (DSSS 2 mVrms and DSSS 5 mVrms). PLC module #1 show within the receiver bandwidth $f_c \pm 80$ kHz given by the frontend BP-filter an about constant mean sensitivity of $SIR_{average} \pm 80$ kHz = +3 dB against single-tone interference. Module #2 is designed as a superhet receiver with fix intermediate frequency of 10.7 MHz and an IF-filter bandwidth of ± 165 kHz.

Module #2 shows a strong sensitivity against single-tone interferer located in a range of about ± 30 kHz around the carrier frequency and is independent on the IF-filter properties. We assume that the carrier tracking of the internal carrier recovery loop is probably shifted to the single-tone interferer and not to the carrier; therefore additional bit errors occur and the properties of the loop filter are mirrored in Figure 8. In general, the sensitivity to single-tone interference follows the characteristics of the BP-filter used and is highest in the passband (Figure 9). Transmission methods having a process gain like DSSS can significant reduce the sensitivity to interferences. The low order and quality of the input filters of the DSSS system compared to module #1 and module #2, which uses ceramic filters of high quality, shows clear disadvantages against interference reduction for signals that are several hundred kHz away from the carrier. This fact should be improved in the next evolution of the DSSS system.

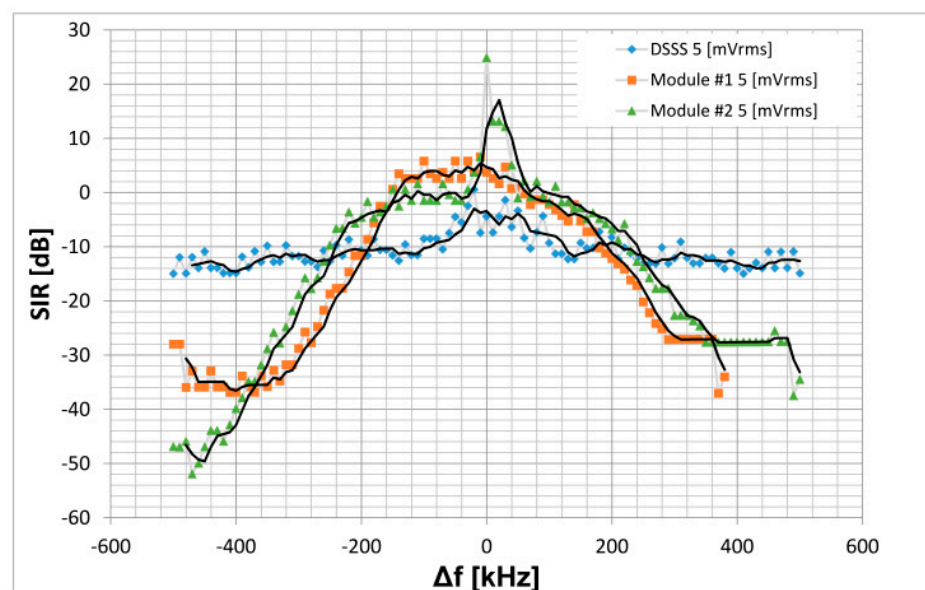


Figure 9. Wideband measurement of the SIR sensitivity.

3.2. Sensitivity to AWGN on the Transmission Channel

The effect on bit, byte and packet errors under influence of additive white Gaussian noise (AWGN) on the transmission channel was investigated for all transceiver modules. The noise source for all measurements was a function generator (33250A, Keysight, Santa Rosa, CA, USA) with a noise bandwidth of 50 MHz. The transmission channel was modeled by a resistive damping network and the noise was supplied to the test setup as shown in Figure 5. The numbers of corrupted bits, bytes, and LIN data packets were counted during transmission and the corresponding error rates were calculated. In general the SNR on the transmission channel (SNR_c) can be calculated from the effective values of the modulation signal u_{signal_rms} and the noise signal u_{n_rms} at the input of the receiver which were measured with a high-speed sampling oscilloscope:

$$SNR_c = \frac{P_s}{P_n} = \frac{u_{signal_rms}^2}{u_{n_rms}^2} = \frac{P_s}{N_0 B_n} \quad (3)$$

The noise power was calculated by:

$$P_n = u_{n_rms}^2 = N_0 B_n \quad (4)$$

where N_0 is the single-sided spectral noise power density and B_n is the noise bandwidth of the generator.

The noise bandwidth B_n of the function generator used is 50 MHz according to the data sheet which was also validated by measurements. A simplified performance comparison of different transceiver modules under AWGN conditions can be made based on the SNR on the transmission channel (SNR_c) and on the resulting data error rates P_e which are depicted in Figure 10. However, this comparison does not take into account the different system designs with regard to the possible data transmission rates and the necessary filter/noise bandwidths.

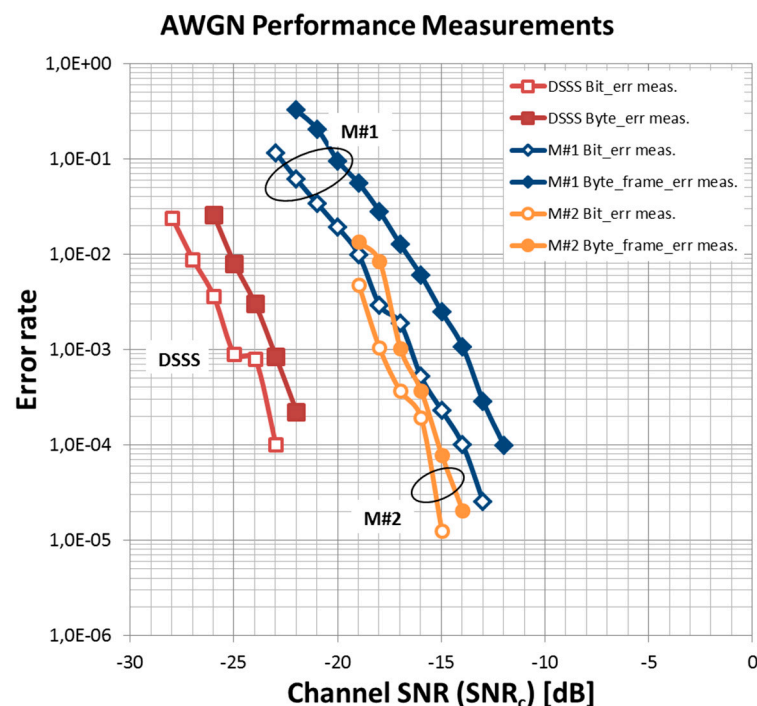


Figure 10. Performance of different transceiver systems under AWGN conditions. On the DSSS system the bit and byte error rate was measured, on module #1 and #2 the bit and byte-frame (preamble+byte) error rate.

The DSSS transceiver is designed for a maximum data rate of 19.2 kbit/s where the commercial modules are designed for maximum data rates up to 115 kbit/s and, therefore, their receiving filter bandwidths. For data rates below the maximum possible data rate, the filter bandwidths of the receivers are not matched to the transmitted data rate. Compared to an ideally matched filter, there is an excess bandwidth which leads to a deterioration of the SNR at the receiver's decision unit. The SNR per bit (E_b/N_0) at the receiver's sampler can be approximated by:

$$\frac{E_b}{N_0} = SNR_c \cdot \frac{B_n}{R_m} \cdot \frac{B_m}{B_{n_filter}} \cdot a_f \quad (5)$$

where R_m is the symbol rate at the noise limiting filter. The filter used depends on the system design and can be for the baseband case a LP-filter or for the passband case a BP-filter. The message bandwidth B_m is defined by the modulation format and the data rate. The noise bandwidth of the receiver filter used is B_{n_filter} and a_f is a factor that depends on the power loss of the message signal due to signal shaping by the receiver filter. The specific

system parameters are given in Table 1. As shown in Figure 4, the noise bandwidth of the DSSS system is given by the integration filter which is realized by a 2nd-order active Sallen-Key low pass filter with a corner frequency of 8.8 kHz which is about $R_b/2$. The noise bandwidth of the LP filter was found to be about 19.2 kHz. The low pass filter shapes the almost rectangular modulation signal into an approximately sinusoidal signal where the signal power changes by the factor a_f .

Table 1. System parameters.

System Parameters ($R_b=19.2$ kB/s)		DSSS	M#1	M#2
Message rate	$R_m = R_b$	19.2 k		
	$R_m = R_c = 3R_b$		57.6 k	57.6 k
Message bandwidth	$B_m = R_b/2$	9.6 kHz		
	$B_m = R_c = 3R_b$		57.6 k	57.6 k
Noise bandwidth	B_{n_filter}	19.2 kHz	160 kHz	330 kHz
Filter shape factor	a_f	-1.7 dB	0 dB	0 dB

The noise power limitation of transceiver module #1 is done by a 160 kHz BP-filter at the receiver’s frontend. A possible receiver structure for the used SDPSK modulation format is given in Figure 11. As described in chapter II module #1 and #2 transmit a data bit by 3 chips with a chip time $T_c = T_b/3$. Therefore, the symbol rate R_m on the channel is $3R_b$ resulting in a passband message bandwidth B_m of $3R_b$. Since a high-quality ceramic BP-filter is used (#773-0065, Oscilent Corporation, Irvine, CA, USA) the filter noise bandwidth of module #1 can be approximated with the filter bandwidth $B_{n_filter} = B_{BP} = 160$ kHz. The signal shaping by the BP-filter can be neglected with respect to the message bandwidth due to the high excess bandwidth and, therefore, $a_f = 0$ dB.

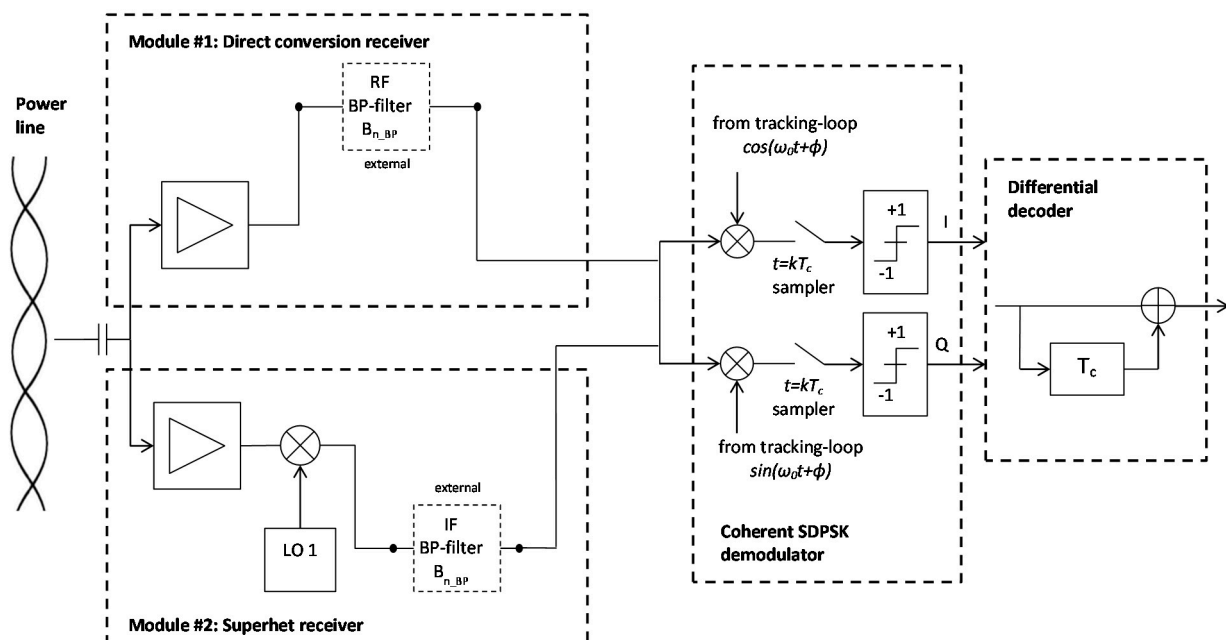


Figure 11. Possible receiver structure of module #1 (direct conversion receiver) and module #2 (superhet receiver) with SDPSK demodulation.

3.3. Calculation of the Bit Error Rate (P_e)

In all modules investigated, synchronous detection of the differential encoded symbols is done by a suppressed carrier tracking loop which is restoring the phase ambiguity (Figures 4 and 11). For coherent detection of differentially encoded signals, the assumption is that the channel phase characteristic is constant during the $N-1$ th and the n th transmission intervals which is equivalent to requiring that the loop does not lose synchronism for at least two symbol periods. A general solution of the error probability P_e for coherent detection of differentially encoded binary PSK signals [29] is given by:

$$P_e = \operatorname{erfc} \left(\sqrt{\frac{d_{min}^2}{4N_0}} \right) \left[1 - \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{d_{min}^2}{4N_0}} \right) \right]. \quad (6)$$

In the DSSS module, coherent demodulation of the differentially encoded phase-modulated signal (DE-BPSK) is used. Under AWGN conditions the probability of a bit error for DE-BPSK is given by

$$P_{e \text{ DE-BPSK}} = \operatorname{erfc} \left(\sqrt{\frac{E_b}{N_0}} \right) \left[1 - \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{E_b}{N_0}} \right) \right] \quad (7)$$

where the Euclidian distance for antipodal signal constellations like in DE-BPSK (Figure 12) $d_{min} = 2\sqrt{E_b}$ was substituted in Equation (6). The modulation format used in module #1 and #2 is SDPSK. In SDPSK the present symbol is either $+\pi/2$ or $-\pi/2$ in phase from the previous symbol, depending on whether an input data is '1' or '0'. For symbol reconstruction 4 decision areas are necessary (Figure 12). In DE-BPSK the Euclidian distance d_{min} between 2 antipodal signal points is $2\sqrt{E_b}$ despite to SDPSK where the distance is $d_{min} = \sqrt{2E_b}$ because of the 4 decision areas necessary. Since coherent demodulation with differential decoding is used, Equation (6) is also valid for SDPSK under the assumption that $d_{min} = \sqrt{2E_b}$ and therefore the bit error rate is:

$$P_{e \text{ SDPSK}} = \operatorname{erfc} \left(\sqrt{\frac{E_b}{2N_0}} \right) \left[1 - \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{E_b}{2N_0}} \right) \right] \quad (8)$$

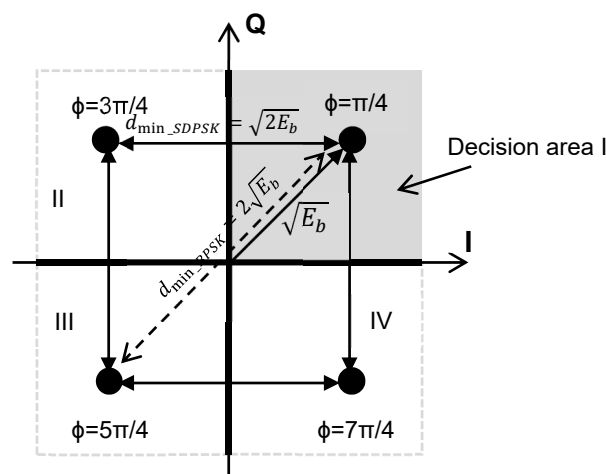


Figure 12. Symbol constellation diagram of SDPSK with 4 decision areas for symbol reconstruction.

The SNR per bit (E_b/N_0) at the receiver's decision unit is calculated for all modules with Equation (5) and using Table 1.

3.4. Calculation of the Byte Error Rate (P_{e_byte})

If the bit transmission errors occur randomly and independently with probability P_e the binomial frequency functions give the probability of errors in an n -bit word:

$$P(i, n) = \binom{n}{i} P_e^i (1 - P_e)^{n-i}. \tag{9}$$

The probability of at least one error in an n -bit word without using forward error correction (FEC) is, therefore:

$$P_{e_word} = 1 - P(0, n) = 1 - (1 - P_e)^n. \tag{10}$$

The DSSS modem transmits in the UART-mode single data bytes where each data byte is led by a start-bit and followed by two stop bits. A transmitted data byte, therefore, consists of a total of 11 bits, whereby each bit failure leads to a byte error. The probability of a byte error by given bit error probability $P_{e_DE-BPSK}$ is:

$$P_{e_byte_DSSS} = 1 - (1 - P_{e_DE-BPSK})^{11} \tag{11}$$

and is shown in the following figures.

A different situation is given at module #1 and #2. The data transmission principle is based on symmetric differential phase-shift keying (SDPSK) [30]. Symmetric DPSK is a quadrature version of differential phase-shift keying (DPSK) where the relative phase shift between successive symbols is either $+90^\circ$ or -90° degrees depending on whether a logic '1' or '0' is to be sent. Each data byte transmitted by modules #1 and #2 is started with a preamble consisting of 5 chips with a chip time duration $T_c = T_b/3$. A data bit is transmitted by three consecutive phase shifts of $+90^\circ$ for a '1' data bit and three consecutive phase shifts of -90° for a '0' data bit. Each phase shift has a time duration of $T_b/3$ (Figure 13). The preamble which is sent in front of each data byte consists of five chips with phase changes $[+, -, +, -, -] \cdot 90^\circ$ and total time duration $5 \cdot T_c$.

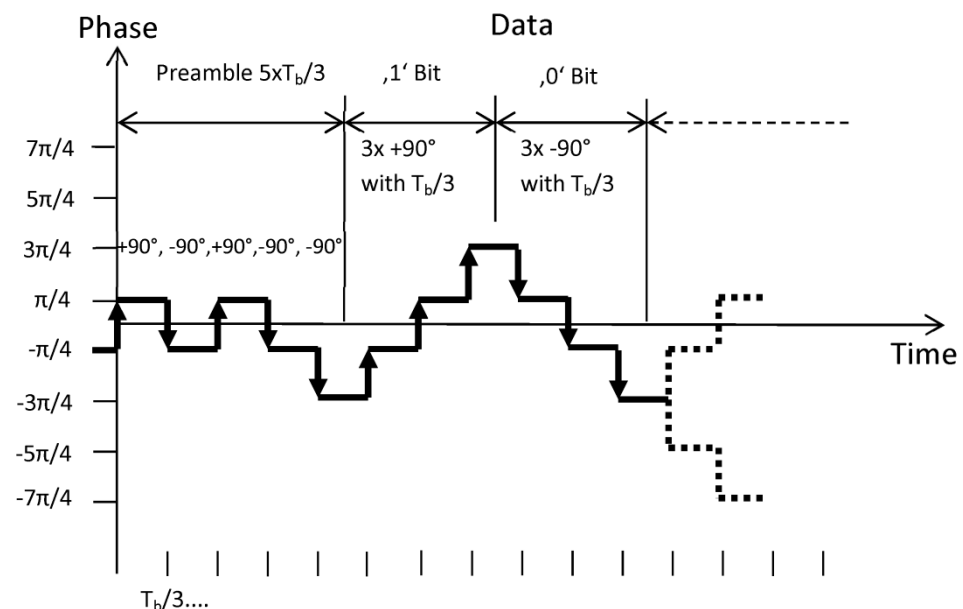


Figure 13. Example of a byte transmission of module #1 and #2 consisting of SDPSK modulated preamble and data bits.

Each transmitted data byte is preceded by a preamble of five chips with chip time duration $T_c = T_b/3$. For this word (preamble) consisting of five chips the error probability can be calculated by:

$$P_{e_pre_M} = 1 - \left(1 - P_{e_SDPSK_chip}\right)^5 \tag{12}$$

where $P_{e_SDPSK_chip}$ is calculated from Equation (8) substituting the E_b/N_0 by the SNR per chip $E_c/N_0 = E_b/3N_0$.

Subsequent to the preamble, a data byte with 11 bits where each bit consists of three chips with chip time duration $T_b/3$ (Figure 13) is transmitted. For a data bit consisting of three chips, a triple repetition code was used where a '1' bit is represented by 111 chips and a '0' bit is represented by 000 chips. With this encoding method a lower error probability is expected. Based on the majority-rule decoding of the triple repetition code, a single chip error can be corrected. That means and at least two of the three chips per bit must be correctly transmitted. Only double and triple errors result in a decoding error which can be formulated by:

$$P_{e_bit_M} = P(2,3) + P(3,3) = 3P_{e_SDPSK_chip}^2 - 2P_{e_SDPSK_chip}^3 \tag{13}$$

The probability of a data byte error where the byte consist equal to the DSSS modem of a leading start-bit, eight data bits, and two stop bit is then:

$$P_{e_byte_M} = 1 - (1 - P_{e_bit_M})^{11} \tag{14}$$

As previously described a transmitted frame for a data byte by module #1 and #2 consists in total of a five-chip preamble and eight data bits with an additional one start and two stop bits. Thus, the total error probability of the transmitted frame is a combination of the preamble error probability $P_{e_pre_M}$ and the data bit error probability $P_{e_byte_M}$. Since a triple repetition code is used for the data bits, the probability of an error in the preamble is much greater than that of a data bit error. However, a single chip error in the preamble leads to a preamble error and thus to a loss of the full data byte. Both error probabilities are depicted in Figure 14 for comparison. From theoretical calculations, it was found that the error probability of a data frame consisting of a five-chip preamble and a data byte depends mainly on the error probability of the preamble.

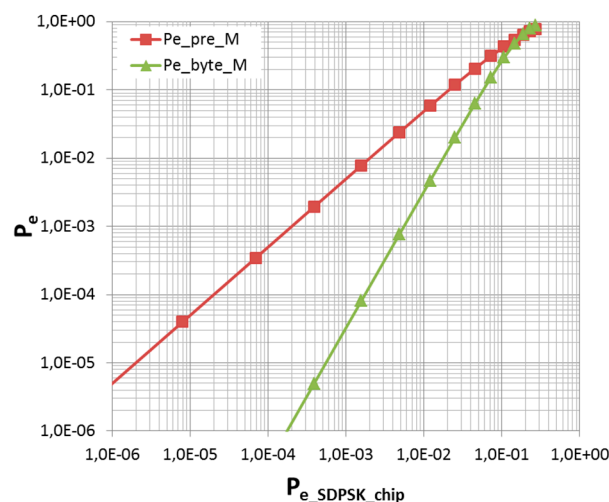


Figure 14. Theoretical error probability of module M#1 and M#2 for the transmission of a preamble ($P_{e_pre_M}$) or a data byte ($P_{e_byte_M}$).

For a rough but reasonable performance estimate, the probability of a byte frame decoding error is given by the approximation:

$$P_{e_byte_frame} = \frac{N_{pre}}{N_{frame}} P_{e_pre_M} + \frac{N_{byte}}{N_{frame}} P_{e_byte_M} \quad (15)$$

In Equation (15), the number of decisions required to decode a preamble is given by N_{pre} ($N_{pre} = 5$), the number of decisions required to decode a data byte is N_{byte} ($N_{byte} = 11$) and N_{frame} is the total number of decisions required to decode a data byte frame, which is the sum of N_{pre} and N_{byte} .

3.5. LIN-Packet Error Rate (PER)

The format of a transmitted LIN-packet is given in Figure 6. A LIN-packet error (PER) was counted if the LIN preamble or one or more bytes of the transmitted LIN-packet payload were corrupted. The theoretical LIN-packet error rate was calculated under the assumption that in a corrupted LIN-packet at least one preamble chip or data bit is incorrect. The preamble of a LIN-packet consists of the 'break-field' (13 bit low) and the bytes B0 and B1 of 11 bit each. The 'break-field' is not part of the error analysis. The LIN preamble is followed by 8 payload data bytes and a terminating CRC byte where each byte contains an additional 1 start and 2 stop bits. Therefore, the total number of bits of a transmitted LIN-packet is $n = 121$ bits (Figure 6). The probability of at least one error in a transmitted LIN-packet of $n = 11$ bytes \cdot 11 bits = 121 bits transmitted with the DSSS system and, thus, a LIN-packet error can be calculated by:

$$PER_{DSSS} = 1 - (1 - P_{e_DE-BPSK})^{121} \quad (16)$$

The commercial modules #1 and #2 transmit each of the 11 LIN-packet bytes with an additional preamble of 5 chips as discussed in chapter III.B.2. Due to the error correction capability by the triple repetition coding of the data bits, a LIN-packet error is therefore mainly caused by chip errors in the 11 data byte preambles with each consisting of five chips. The PER based on chip errors in the preambles of one LIN-packet is given by:

$$PER_{preamble} = 1 - \left(1 - P_{e_SDPSK_chip}\right)^{5 \cdot 11} \quad (17)$$

and the PER based on bit errors is given by:

$$PER_{bit} = 1 - (1 - P_{e_bit_M})^{11 \cdot 11}. \quad (18)$$

For module #1 and #2 the overall probability of a LIN-packet error caused by an error in the preamble or a data bit error is:

$$PER_M = \frac{N_{LIN_pre}}{N_{LIN}} PER_{preamble} + \frac{N_{LIN_bit}}{N_{LIN}} PER_{bit} \quad (19)$$

where $N_{LIN_pre} = 11 \times 5 = 55$ indicates the number of decisions necessary to decode all preambles, $N_{LIN_bit} = 11 \times 11 = 121$ indicates the number of decisions necessary to decode all bits and N_{LIN} is the total number of decision necessary to decode a LIN-packet ($N_{LIN} = N_{LIN_pre} + N_{LIN_bit} = 176$).

In Figure 15a–c a comparison of the theoretical and the measured bit error rates (BER), byte error rates (Byte_err) and packet error rates (PER) for all transceiver modules are given. All measurements were repeated three times and the mean values are depicted. The AWGN performance of the DSSS transceiver Figure 15a fits excellent with theories given in Equations (8), (11), and (16) using the system parameters given in Table 1. Only for low error rates do the measurements show a degradation in performance with respect to the theory. This might be caused by jitter in the carrier recovery loop.

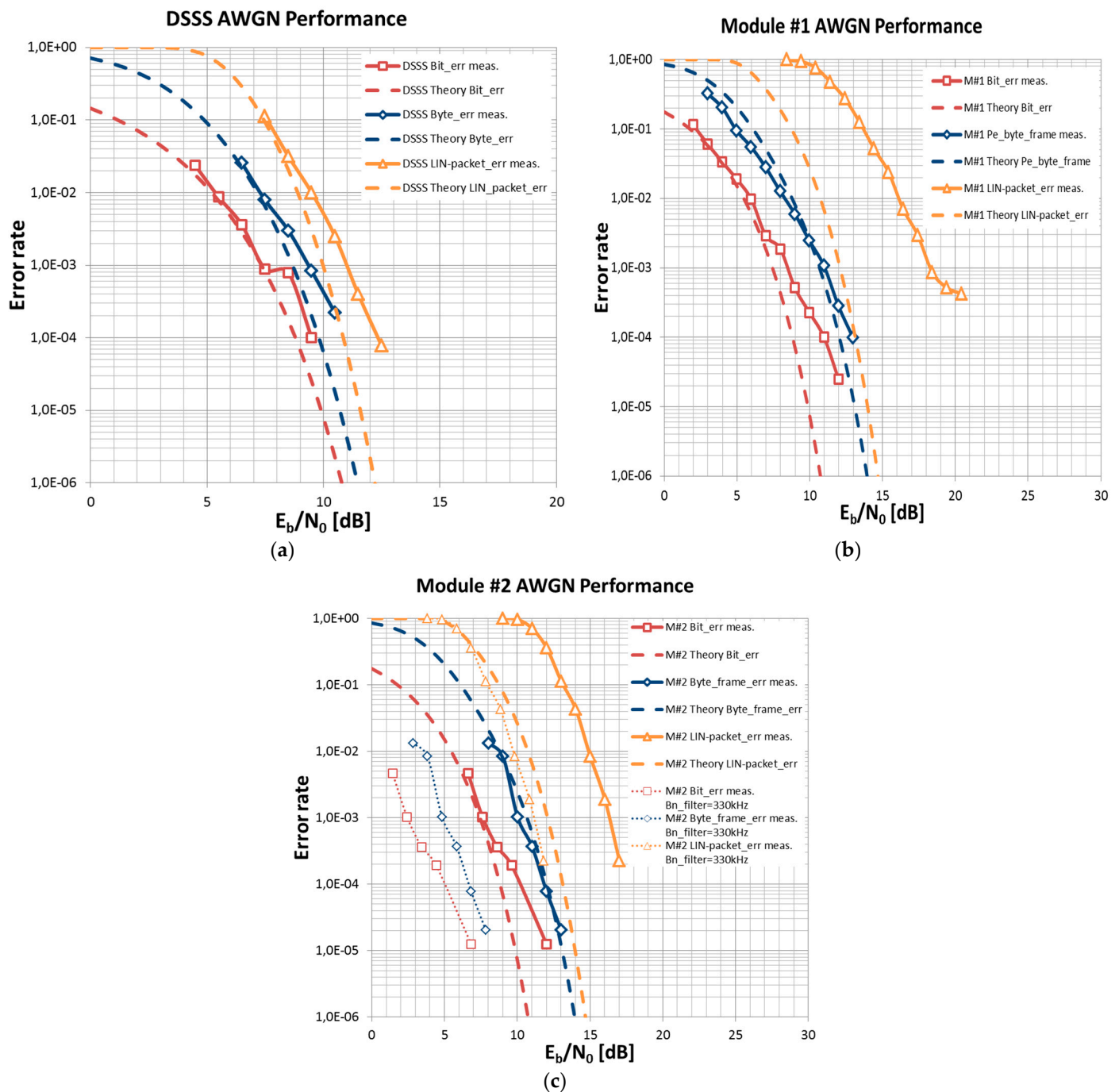


Figure 15. Mean values of the measured bit-, byte- and packet error rates under AWGN conditions in comparison to the theoretical values of the (a) DSSS system, (b) module #1, and (c) module #2.

The AWGN performance of the commercial modules #1 and #2 is given in Figure 15b,c. A direct conversion receiver is used in module #1 and a superhet receiver in module #2. The performance of the bit-error rate and the byte-frame error rate of module #1 show a good fit to the theoretical calculations which are given in Equations (13), (15), and (19) using the system parameters of Table 1. From the measurements, we found that the noise filter bandwidth of module #1 was determined by the bandwidth of the ceramic frontend filter (Table 1) and fit excellently to the theoretical predictions. However, in module #2 the obtained measurement results do not match to the noise bandwidth of the IF filter (Table 1, dotted lines in Figure 15c). Based on the measurement results, we assume that module #2 contains an internal digital filter with a filter noise bandwidth of about 100 kHz

in addition to the IF filter. This assumption also supports the minimum channel spacing of 100 kHz mentioned in the datasheet. Based on the assumption of an approximate filter noise bandwidth of $B_{n_filter} = 100$ kHz, the measurements obtained fit exactly to the theoretical prediction shown in Figure 15c. This confirms that the system design of the commercial modules has been largely optimized. The significant deviation of the LIN-packet error rate from the theoretical prediction for both modules could not be clearly explained. Especially for module #1 it could be found that, at a low SNR, a valid preamble was erroneously detected in transmission breaks and, thus, the following noise was erroneously decoded as a data byte leading to an increased error rate. If these de-coding errors occur during LIN-packet transmission, LIN-packets are erroneously decoded and the packet error rate can increase considerably.

4. Conclusions

The performance of a self-designed direct sequence spread spectrum modem for PLC in automotive applications was investigated. The DSSS modem was compared with two commercial PLC modems which are qualified for automotive applications based on a direct conversion receiver (module #1) and a superhet receiver (module #2). It was shown that the performance of the DSSS modem against single-tone interference was as expected about the processing gain PG of 8 dB better than the commercial modules. At data rates of 19.2 kB/s used for LIN packet transmission, the performance of the DSSS module was better than the other modules by about 6 dB in bit error rate under AWGN conditions. It was clearly shown that the better performance of the DSSS system is due to the different designs of the transceiver modules regarding their maximum transferable data rate and was proven by theoretical analysis. If this aspect is taken into account, the performance is comparable in terms of bit error and byte error rate. Only in the transmission of LIN packets does the performance of the commercial modules appear to be lower than that of the DSSS system. The main advantage of the DSSS system is the significantly better interference suppression and the very low-cost design of the entire module with component costs of about 1 Euro.

Author Contributions: Conceptualization, M.B. and K.K.; software, K.K.; validation, M.B. and K.K. formal analysis, M.B.; writing—original draft preparation, M.B.; project administration, M.B.; funding acquisition, M.B. All authors have read and agreed to the published version of the manuscript.

Funding: Open Access Funding by the University for Continuing Education Krems.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No available data.

Acknowledgments: This work is co-funded by the government of Lower Austria and the European Regional Development Fund (ERDF). Project ID: WST3-F-5030664/004-2017.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ferreira, H.C.; Lampe, L.; Newbury, J.; Swart, T.G. *Power Line Communications: Theory and Applications for Narrowband and Broadband Communications over Power Lines*; John Wiley & Sons: Hoboken, NJ, USA, 2011.
2. Lampe, L.; Tonello, A.M.; Swart, T.G. *Power Line Communications: Principles, Standards and Applications from Multimedia to Smart Grid*; John Wiley & Sons: Hoboken, NJ, USA, 2016.
3. Barmada, S.; Raugi, M.; Tucci, M.; Maryanka, Y.; Amrani, O. PLC systems for electric vehicles and Smart Grid applications. In Proceedings of the 2013 IEEE 17th International Symposium on Power Line Communications and Its Applications, Johannesburg, South Africa, 24–27 March 2013; pp. 23–28. [[CrossRef](#)]
4. Pittolo, A.; de Piante, M.; Versolatto, F.; Tonello, A.M. In-Vehicle Power Line Communication: Differences and Similarities among the In-Car and the In-Ship Scenarios. *IEEE Veh. Technol. Mag.* **2016**, *11*, 43–51. [[CrossRef](#)]

5. Mohammadi, M.; Lampe, L.; Lok, M.; Mirabbasi, S.; Mirvakili, M.; Rosales, R.; van Veen, P. Measurement study and transmission for in-vehicle power line communication. In Proceedings of the 2009 IEEE International Symposium on Power Line Communications and Its Applications, Dresden, Germany, 29 March–1 April 2009; pp. 73–78.
6. Lienard, M.; Carrion, M.O.; Degardin, V.; Degauque, P. Modeling and analysis of in-vehicle power line communication channels. *IEEE Trans. Veh. Technol.* **2008**, *57*, 670–679. [[CrossRef](#)]
7. Navet, N.; Song, Y.; Simonot-Lion, F.; Wilwert, C. Trends in automotive communication systems. *Proc. IEEE* **2005**, *93*, 1204–1222. [[CrossRef](#)]
8. Huck, T.; Schirmer, J.; Hogenmüller, T.; Dostert, K. Tutorial about the implementation of a vehicular high speed communication system. In Proceedings of the 2005 International Symposium on Power Line Communications and Its Applications, ISPLC 2005, Vancouver, BC, Canada, 6–8 April 2005; Volume 2005, pp. 162–166.
9. Schiffer, A. Statistical channel and noise modeling of vehicular DC-lines for data communication. *IEEE Veh. Technol. Conf.* **2000**, *1*, 158–162.
10. Nolte, T.; Hansson, H.; Bello, L.L. Automotive communications—Past, current and future. In Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation, ETFA, Catania, Italy, 19–22 September 2005; pp. 985–992.
11. Maryanka, Y. Wiring reduction by battery power line communication. In Proceedings of the IEEE Seminar on Passenger Car Electrical Architecture (Ref. No. 2000/088), Birmingham, UK, 21 June 2000; Volume 8, pp. 1–4. [[CrossRef](#)]
12. da Rocha Farias, L.; Monteiro, L.F.; Leme, M.O.; Stevan, S.L. Empirical analysis of the communication in industrial environment based on G3-power line communication and influences from electrical grid. *Electronics* **2018**, *7*, 194. [[CrossRef](#)]
13. Zhu, Q.; Chen, Z.; He, X. Resource allocation for relay-based OFDMA power line communication system. *Electronics* **2019**, *8*, 125. [[CrossRef](#)]
14. Ahiadormey, R.K.; Anokye, P.; Lee, K.J. Cooperative Non-Orthogonal Multiple Access over Log-Normal Power Line Communication Channels. *Electronics* **2019**, *8*, 1254. [[CrossRef](#)]
15. ISO 17987. *Road Vehicles-Local Interconnect Network (LIN)*; ISO: Geneva, Switzerland, 2016.
16. LIN Specification Package, Revision 2.2A, LIN Consortium. 2010. Available online: https://www.cs-group.de/wp-content/uploads/2016/11/LIN_Specification_Package_2.2A.pdf (accessed on 28 November 2019).
17. Ziemer, R.E.; Peterson, R.L. *Digital Communications and Spread Spectrum Systems*; Macmillan: New York, NY, USA, 1985.
18. Kavehrad, M.; Bodeep, G. Design and experimental results for a direct-sequence spread-spectrum radio using differential phase-shift keying modulation for indoor, wireless communications. *IEEE J. Sel. Areas Commun.* **1987**, *5*, 815–823. [[CrossRef](#)]
19. Kavehrad, M.; Ramamurthi, B. Direct-sequence spread spectrum with DPSK modulation and diversity for indoor wireless communications. *IEEE Trans. Commun.* **1987**, *35*, 224–236. [[CrossRef](#)]
20. Chung, B.Y.; Chien, C.; Samueli, H.; Jain, R. Performance analysis of an all-digital BPSK direct-sequence spread-spectrum IF receiver architecture. *IEEE J. Sel. Areas Commun.* **1993**, *11*, 1096–1107. [[CrossRef](#)]
21. Song, M.K.; Sakong, S.C.; Tchah, K.H. Realization of power line modem using a direct sequence spread spectrum technique. *IEEE Trans. Consum. Electron.* **1993**, *39*, 646–652. [[CrossRef](#)]
22. Wen, N.D.; Hui, Z.B.; Dong, L.; Bo, Z.Q.; Klimek, A. Application of Spectrum-spread Technology in Power Line Communication. In Proceedings of the 45th International Universities Power Engineering Conference UPEC2010, Cardiff, Wales, UK, 31 August–3 September 2010; pp. 1–4.
23. Goiser, A.M.J. Direct-sequence spread-spectrum modem for data communications over power-lines. In Proceedings of the MIL-COM 97, Monterey, CA, USA, 3–5 November 1997; Volume 2, pp. 1052–1056.
24. Chen, Z.; Feng, X. A novel design of Direct Sequence Spread Spectrum receiver based on software radio. In Proceedings of the International Conference on Wireless Communications and Signal Processing, Nanjing, China, 13–15 November 2009; pp. 1–4. [[CrossRef](#)]
25. Brandl, M.; Kellner, K.; Posnicek, T.; Hochwarter, D. Performance Evaluation of a DQPSK and a DSSS PLC-Modem for Vehicular Applications. In Proceedings of the IEEE International Symposium on Power Line Communications and its Applications (ISPLC), Praha, Czech Republic, 3–5 April 2019; pp. 1–5.
26. Barker, R.H. Group synchronizing of binary digital systems. In *Communication Theory*; Academic Press: New York, NY, USA, 1953; pp. 273–287.
27. Lindsey, W.C.; Simon, M.K. *Telecommunication Systems Engineering*; Courier Corporation: North Chelmsford, MA, USA, 1991.
28. Proakis, J.G.; Salehi, M. *Communication Systems Engineering*, 2nd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
29. Lindsey, W.; Simon, M. On the detection of differentially encoded polyphase signals. *IEEE Trans. Commun.* **1972**, *20*, 1121–1128. [[CrossRef](#)]
30. Winters, J. Differential detection with intersymbol interference and frequency uncertainty. *IEEE Trans. Commun.* **1984**, *32*, 25–33. [[CrossRef](#)]