Proc. Industrial Embedded Systems, p. 97 - 102, Montpellier, France, 2008

A Digital Receiver Architecture for RFID Readers

Christoph Angerer
Institute of Communications and Radio Frequency Engineering
Vienna University of Technology
Gusshausstrasse 25/389, 1040 Wien, Austria
Email: cangerer@nt.tuwien.ac.at

***Abstract*— This paper presents a digital receiver architecture for an RFID reader. The main challenge in RFID reader design is the detection of the backscattered signals from the tags, which can be severely complicated due to the largely varying scale of possible receive powers. Furthermore noise, which power depends on the environment can degrade the detection performance. The detection of the signals of the tags is additionally impeded by the very strong self interference at the reader with the carrier it needs to send in order to supply the tags with energy. To fight these various disturbances, a new RFID receiver algorithm is proposed, that sets its decision threshold adaptively, depending on the strength of the input signal, the noise power at the receiver and the extent of the carrier interference. This is the first algorithm for signal detection in RFID, setting its threshold accordingly to the environmental conditions, and thus leading to near optimum performance. Details of the implementation of the digital receiver architecture on an FPGA are introduced. Bit error ratio measurements have been carried out to rate the receivers performance, which have never been shown before for RFID receivers. Presented measurement results substantiate the performance of the suggested algorithm.**

I. Introduction

Radio Frequency Identification (RFID) is an identification technique such as the barcode or magnetic strip technology. It is nowadays already used in many applications such as automated library systems, public transport, access control and security, product tracking or logistics, and has not yet exploited its potential to full extent. RFID systems operate in different frequency bands, from very low frequencies (around 130 kHz) over high frequencies (HF) range at 13.56 MHz up to ultra high frequencies (UHF, 868 - 950 MHz).

In RFID technology one reader or interrogator establishes the communication link with typically several tags, which can in general be separated into two categories: active and passive tags. In contrast to active tags, passive ones do not carry their own power supply. They obtain the energy needed for retrieving, processing and transmitting information from the electromagnetic field emitted by the interrogator and respond to its requests by a backscatter modulation of the carrier [1]. This implies, that the reader has to provide the energy in form of a continuous carrier (CC) signal, even when no information is being transferred to the tag, including those time periods in which the tag responds to the requests ($T \rightarrow R$). Figure 1 demonstrates this situation on an example of a basic RFID communication: The reader always transmits a continuous carrier (CC), except during the times of interrogation ($R \rightarrow T$), where this carrier is modulated. After a certain idle period (time t_1), the tags respond to the reader ($T \rightarrow R$) by backscattering the carrier.

This work has been funded by the Christian Doppler Laboratory for Design Methodology of Signal Processing Algorithms

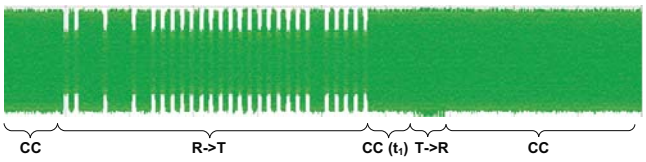

Fig. 1. Basic RFID communication.

However, this results in a self interference due to the superposition of the transmitted powering field with the tag to reader backscatter modulation, that features a typically many magnitudes lower signal strength. Furthermore, the receive power at the RFID reader heavily depends on the tag to reader distance, and hence can vary significantly. Especially in industrial environments noise severely degrades the detection performance at the receiver. Thus, one of the predominant challenges in RFID reader design is a sophisticated receiver technique to reliably detect the very small reflected tag response among the huge self interference and noise.

This work focuses on the aforementioned problems. A new receiver algorithm is introduced and its implementation on an FPGA is presented. The proposed system has been set up on a rapid prototyping system [2] and has been verified with measurements with state-of-the-art tags over varying distances. Several standards are supported by this rapid prototyping testbed and measurements for evaluating the receive algorithm have been carried out according to the draft of the EPC global HF standard [3].

II. Related Work

Only quite limited amount of work has been carried out on signal detection on digital RFID receivers and their implementations. C. Mutti et al. [4] compared different detection algorithms that exploit the correlation properties of a range of encodings with memory, namely differential biphase, Manchester and Miller encodings. Their work is of theoretical nature and did hence not consider any issues arising from a real time implementation, and neither they did present any

results based on measurements. Y. Liu et al. [5] proposed a receiver architecture based on twelve correlators with multiply accumulate operations to estimate the precise symbol timings and frequency offsets. The implementation of the proposed correlator banks however consumes a huge amount of hardware resources due to their lengths of 51 stages on average, which exceeds the capacity of average FPGAs. However, they show good results regarding the frequency offset tolerance of their receiver, but they did not address the issue of signal detection. Choi et al. [6] present a VHDL and PSPICE model, focusing on multi tag recognition and anticollision protocols. They also present some measurement results, but do not focus on detection of the receive signal of the reader. Additional publications rather focus on physical limitations and on simulation models of RFID systems. N. Roy et al. [7] introduce their RFID reader architecture built out of off-the-shelf components, including both digital and analogue parts. However they do not comment on any digital signal processing algorithms to detect the signals. Furthermore, there exist several publications on limitations of RFID systems derived from calculations, assuming the required parameters for their models. Mayordomo et al. [8] present an RFID system analysis based on theoretical models and highlight those parameters causing a performance limitation, like oscillator noise etc. In their model a predefined decision threshold for the receive signal detection is assumed to be on the stronger path of either the inphase (I) or the quadrature (Q). That however is not a feasible solution in real scenarios, where receive power may vary strongly, depending on the reader to tag distance, and where we want to utilise the entire signal power available in both the I and Q components at the receiver. Finally, many RFID reader designers rather focus on processing in the analogue domain, and just input the demodulated baseband signal to a pin of a microprocessor, adjusting the signal levels using an automatic gain control circuit [6].

The rest of the paper is organised as follows: The next section gives an overview of the entire architecture of the reader, Section IV presents the receive algorithm and its implementation, Section V includes the verification and measurements while the last section concludes the paper.

III. Reader Architecture

This section first briefly introduces the physical setup of the RFID testbed and then presents an overview of the proposed RFID reader architecture. The implementation is very flexible and does not depend on a particular standard, but has been verified following the draft of the EPC global HF standard [3]. Extensions towards other standards can be easily accomplished by the use of a highly automated design, flow [2] as already demonstrated with the implementation of the ISO 15693 standard [9]. In the following section the testbed setup is described in detail.

A. RFID Testbed Setup

The underlying hardware for the reader implementation consists of a rapid prototyping board which is depicted in Figure 2. The main reconfigurable components are a fixed-point DSP from Texas Instruments (TMS320C6416), used for protocol stack processing and a Xilinx Virtex II FPGA, fulfilling the signal processing tasks. Additionally, it exhibits two digital to analogue converters (16 bits) and two analogue

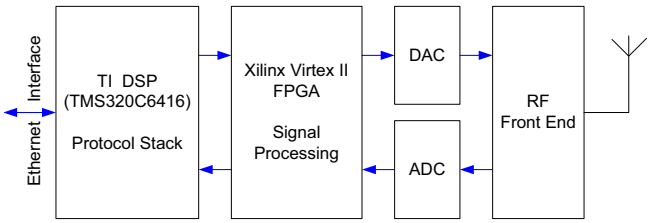

Fig. 2. Block diagram of the rapid prototyping board.

to digital converters (14 bits). The DSP is clocked with 600 MHz while the FPGA, DACs, and ADCs are clocked by a single oscillator of 40 MHz. An ethernet interface connects the DSP to a PC thus allowing for an external communication to an application running on a PC. The RF frontend exhibits a carrier suppression module.

B. FPGA Architecture

In the following we will only focus on the signal processing part implemented on the FPGA. Figure 3 depicts a basic block diagram of its architecture. The design is decomposed into transmit and receive paths.

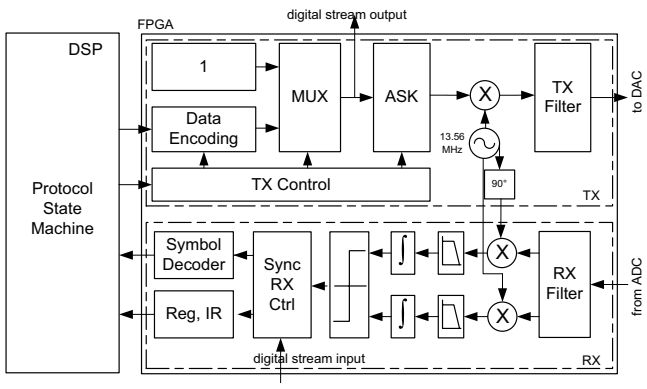

Fig. 3. Block diagram of the FPGA architecture.

The transmitter consists of the DSP to FPGA interface, the data encoder, and a multiplexer switching to continuous carrier mode after the transmission of a sequence. A digital output stream is generated, then modulated with amplitude shift keying (ASK), up-converted to 13.56 MHz and a bandpass filter is applied for pulse shaping.

On the receiver the signal with carrier suppression is sampled at the ADC and the noise bandwidth is reduced by a bandpass filter. The bandwidth is set to 5 MHz, corresponding to 2.5 MHz on each of the two sidebands. This lets pass the third harmonics of the modulating signal, which is 847 kHz,

according to the draft of the EPC global standard [3]. Then the receive signal is down-converted using an I/Q demodulator with low pass filters, followed by subsequent integration.

This integrator is implemented as a moving average block, integrating the samples over a half period of the link frequency (which corresponds to the data rate). Note, that this is essentially a correlation with a rectangular pulse of exactly the half period of the link frequency and hence corresponds to the matched filter, assuming that the tag modulates with rectangular pulses. The implementation of this block is rather inexpensive, since it only requires one adder, one subtracter and a memory of the size of the pulse length, avoiding any multipliers. The bitwidths for the signals within these single units is constrained to 16 bits for the measurements presented later on.

The slicer with the decision algorithm is discussed in the next section in more detail. The following synchronisation unit extracts the bits of the digital stream after the slicer. As the system is highly oversampled (40 MHz clock frequency at a maximum data rate of 847 kHz), this is achieved by choosing the closest multiple of the link frequency periods that a level is hold. For instance, the draft of the EPC global standard can operate at maximum link frequency periods of 847 kHz, corresponding to 24 samples per half period at 40 MHz sampling frequency. If a level is hold for 30 samples for example, 1 half period is decided, if the level is hold for 37 samples, 2 half periods are decided. Finally the FPGA to DSP interface handles the signaling and data transfer to the DSP.

IV. Slicer

The slicer has to detect the receive signal in the presence of the strong interference caused by the energy supplying carrier and the noise at the receiver. Furthermore, the decision threshold needs to be set adaptively, according to the receive power at the input of the slicer, which is likely to vary to a large degree due to the tag to reader distance and fading. Finally the algorithm has to detect if there is any signal at all, which is not the case if there are no tags within the read range of the interrogator.

To illustrate the described situation, the samples during the idle time t_1 and the tag response time ($T \rightarrow R$, Figure 1) at the input of the slicer (baseband), have been captured during a real time measurement in the FPGA. These samples have been imported into Matlab (Figure 4). The first subfigure shows the inphase (I) and quadrature (Q) signal at the input of the slicer. Note, that an offset of 5000 is added to the the quadrature component for clarity in this illustration to avoid overlapping of the two signals in the plot.

There are two symbols to estimate: at first, the value of the idle channel (no backscattering, symbol $S0$) and the second symbol $S1$ at the maximum deviation from symbol $S0$ (tag is backscattering energy). Each RFID standard dictates distinct timing intervals between a request command issued by the reader and the following response from the tag (see Figure 1). This time t_1 is used by the algorithm to estimate both the

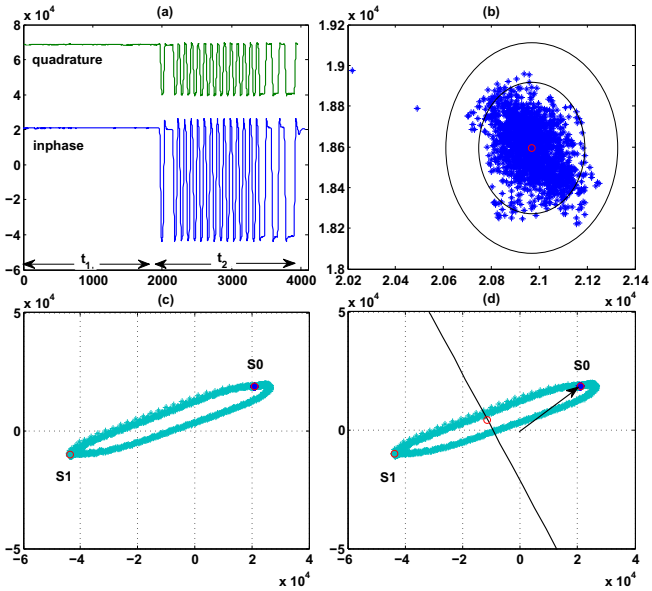

Fig. 4. Detection Algorithm, high SNR.

noise power at the channel and the symbol $S0$, corresponding to the variance and the mean of the signal measured over time t_1, respectively. As the tag does not backscatter any signal during time interval t_1, the symbol $S0$ exactly corresponds to the carrier interference at the receiver. Figure 4b shows the samples during t_1 in a plot of the signal in the I/Q plane. Note the offset from the origin, which again is caused by the carrier interference.

After time t_1 a specific region is defined, outside which we look for the second symbol $S1$. This region is proportional to the estimated noise in each component, for instance $5 * \hat{\sigma}_I$ and $5 * \hat{\sigma}_Q$. Here $\hat{\sigma}_I$ and $\hat{\sigma}_Q$ denote the estimated standard deviations of the noise of each component. In case of a circularly symmetric white gaussian noise process, the two noise components σ_I and σ_Q are equal: $\sigma_I = \sigma_Q = \sigma$. The circles in Figure 4b correspond to the regions outside 2.5σ and 4σ. If a received sample lies outside this region, it is assumed, that there is modulation on the channel. During the following period of the link frequency, symbol $S1$ is determined at the largest deviation from symbol $S0$. Searching outside of this region prevents the decision threshold to be set too low and avoids a decision to be found within the noise. In the presence of low noise, this region is very tight, otherwise only regions with large variations from the symbol $S0$ are scanned. If there are no tags around, the signal only moves outside this region with a probability depending on its size. Assuming gaussian noise, the probability that one sample lies outside the region of 5σ is less than 10^{-6}. Figure 4b also shows the first two samples that belong to the first received symbols (in the left upper corner), which lie outside the regions, where only noise is expected.

Within one period of the link frequency the samples move on a trajectory in the I/Q plane as shown in the plot in

Figure 4c. Due to the high oversampling rate there are several samples within one such period (in turquoise colour). Note the samples of time interval t_1 (in blue), which are located in a very tight region around $S0$. Since now both symbols $S0$ and $S1$ have been estimated, the threshold is set as the straight line separating the two symbols in the middle (see Figure 4d). The decision is taken depending on the half-plane in which the following receive samples are located (during time period t_2 as denoted in Figure 4a). Moreover, the arrow in Figure 4d shows the interference component of the carrier.

The samples in Figure 4 correspond to a high SNR scenario. In case of a system operating in an environment with higher noise or at bigger distances, the plots look very differently, as shown in Figure 5. Note the different position of the symbols in this constellation due to a different carrier interference and receive signal power.

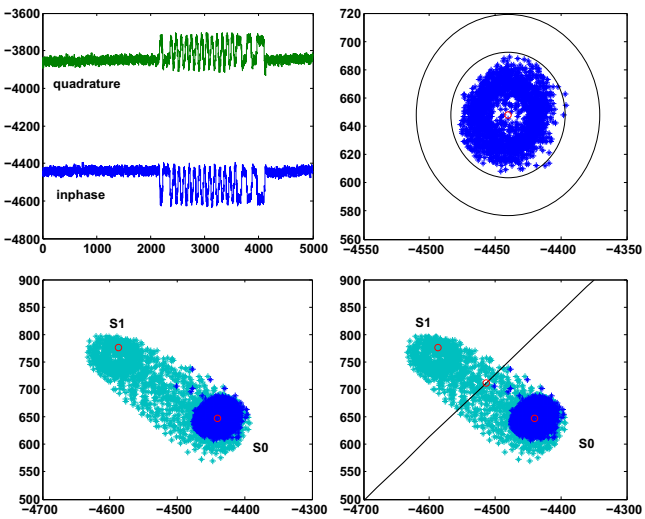

Fig. 5. Detection Algorithm, low SNR.

Note that this algorithm solely decides on the levels of the signal, not utilising the correlation properties of the different encoding schemes in RFID systems, such as FM0, Miller or Manchester encoding.

This algorithm is the first that addresses all of the three aforementioned difficulties. Firstly, it is robust against the noise disturbances, as it defines the search space for the receive signals only outside a defined region depending on the estimated noise power. Secondly, it is invulnerable against the carrier interference, and thirdly it adaptively adjusts the decision threshold according to the power of the received signal. If there are no tags in the read range, the threshold is set above the noise floor, ensuring not to detect any false signals. Yet, a carrier suppression in the analogue domain is still advantageous, in order to utilise the maximum dynamic range of the ADC for the resolution of the response signal of the tag.

The algorithm has been implemented on the FPGA using a finite state machine (see Figure 6) for switching to the different steps of the algorithm. It requires two multipliers, which are

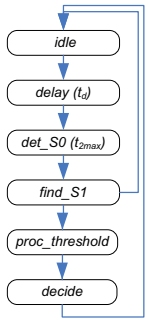

Fig. 6. Detection Algorithm - FSM.

utilised in various stages of the algorithm: first for determining the noise variance, later for computing the distance of samples from the estimated symbol $S0$, then searching for symbol $S1$, subsequently for for calculating the threshold and eventually for taking the decisions.

The state transition from *idle* to *delay* is enabled after a command has been sent to the tag. The algorithm remains in the *delay* state for time t_d, computed as total response delay between reader and tag commands t_1 minus the next smallest power of two value t_{2max}. That is then used in state *det_S0* to estimate symbol $S0$ ($t_{2max} = t_1 - t_d$). Hence, the number of cycles remaining in this state is a power of two, which facilitates the divisions for calculating means and variances of symbol $S0$, corresponding to the carrier interference and the noise power respectively. In the draft of the EPC global HF standard the total delay between reader and tag sequences t_1 is $73\mu s$, hence the statistics of symbol $S0$ are estimated during 2048 cycles, corresponding to $t_{2max} = 51.2\mu s$ at 40 MHz clock frequency and the remaining $21.8\mu s$ (t_d) is waited in state *delay*. The following state is *find_S1*, which searches for samples outside the region in which only noise is expected. Then the most distant sample in the IQ plane within one period of the link frequency is determined to be symbol $S1$. If no samples are detected in this region of interest, it is assumed that there are no tags present in the interrogator's read range and switch back to the *idle* state. Otherwise the FSM switches to the *proc_threshold* to process its threshold and thereafter to the *decide* state. If no signal is above the threshold for a certain time in the *decide* state then, the FSM switches back to the *idle* state.

V. Measurement Results

In order to evaluate the performance of the receiver structure, the bit error ratio (BER) of the system was measured, which has never been shown before for RFID systems. Thus, it was not possible to compare the performance of the interrogator with different implementations, however the detection performance was rated against the theoretical bounds of additive white gaussian noise channels, showing the optimality of the receiver.

For measuring the BER, the setup shown in Figure 7 was used. Both, the SNR at the receiver as well as the probability

of bit errors had to be estimated.

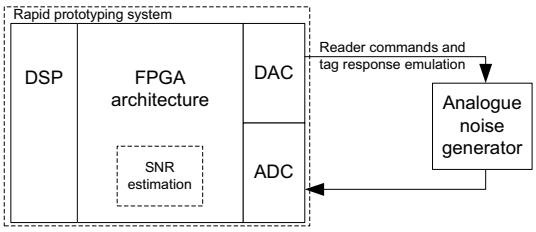

Fig. 7. setup for measuring the BER.

For the estimation of the SNR the estimated total noise power $\widehat{\sigma}_n^2 = \widehat{\sigma}_I^2 + \widehat{\sigma}_Q^2$ available at the receiver is used and the mean receive power of one receive sequence at the input of the slicer on the FPGA during the time interval t_2 is estimated as:

$$\widehat{P}_{RX} = 1/N \sum_{i=0}^{N-1} ((I_{RX}(i) - \widehat{I}_{S0})^2 + (Q_{RX}(i) - \widehat{Q}_{S0})^2 - \widehat{\sigma}_n^2),$$

where $I_{RX}(i)$ and $Q_{RX}(i)$ are the inphase and quadrature component of the sample i at the input of the slicer, $\widehat{I}_{S0}$ and $\widehat{Q}_{S0}$ denote the inphase and quadrature component of the estimated symbol $S0$ respectively and N is the number of samples the receive sequence lasts. This estimation assumes that the noise power σ_n^2 and the signal power are uncorrelated. The estimated SNR at the receiver then is $\hat{SNR} = \widehat{P}_{RX}/\widehat{\sigma}_n^2$.

At each SNR point 10^5 realisations are averaged to calculate the bit error ratio. For the SNR calculation the mean power and the mean noise are processed over all these realisations and then the SNR is computed.

In order to receive a defined test sequence for processing the bit error ratio, the transmitter first sends an interrogation command and thereafter emulates the tag response (Figure 7). The selected tag sequence is a fixed 16 bit random number command as a response to the query command transmitted by the reader. Different SNR scenarios have been realised by adding additional noise before the ADC of the board by means of an analogue noise generator. The emulation of tag responses was furthermore required, in order to generate approximately the same SNR realisation for all runs at each measurement point. This is not possible using commercial tags, as the reflected power may strongly vary with different realisations, even if there is no fading on the channel, as this is the case with the achieved readout distances.

Furthermore, the estimated SNR was measured depending on the distance between reader and tag, with a prototype of a tag following the draft of the EPC global standard in the HF frequency domain. This however did not result in any bit errors up to the maximum readout distance, as at the maximum distance at which the tag still receives enough power to process, hardly any bit errors occur. This shows that this system is limited by the receive power of the tag.

Figures 8 and 9 demonstrate this situation: Figure 8 shows the measured relation between the SNR estimate and the

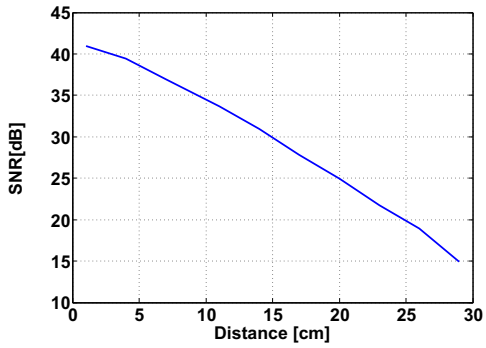

Fig. 8. measured SNR - distance relation.

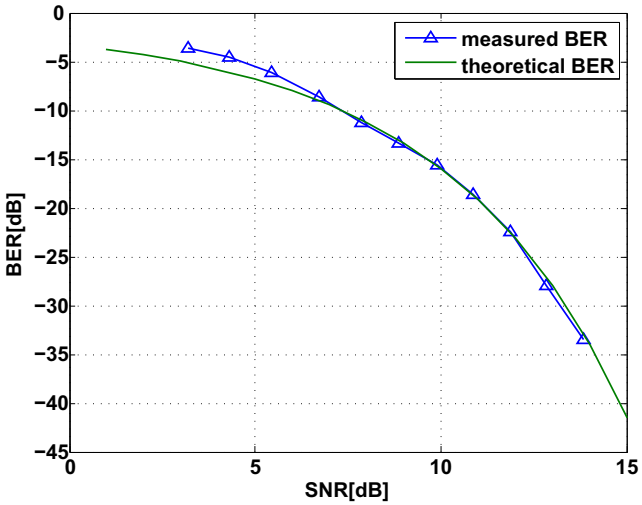

Fig. 9. Measured bit error ratio.

distance. The tag backscatters up to a distance of about $30\,cm$, which is about the diameter of the readers antenna coil, where we still have about $15\,dB$ SNR at the input of the receiver. The blue curve in Figure 9 shows the measured bit error ratio at the receiver depending on the estimated SNR. At $15\,dB$ the bit error ratio is around $-40\,dB$.

The green curve in Figure 9 is the theoretical bit error ratio for the system modeled as an ASK system with the two equally probable symbols without any encoding, corresponding to the signal levels 0 and S_1 and additive white gaussian noise. The mean receive power in such a system then is: $P_{RX} = 0.5 * (0^2 + S_1^2) = S_1^2/2$, and the bit error ratio of such a system is $Q((S_1 - 0)/2\sigma)$, with σ denoting the standard deviation of the noise component in direction of the second symbol. As the noise process is assumed to be a circularly symmetric process, this component σ is $\sigma_n/\sqrt{2}$, where σ_n^2 is the total noise power. This leads to $BER = Q(\sqrt{SNR}/2)$, with $Q(\cdot)$ denoting the Q-function.

The measured curve in Figure 9 almost exactly matches the theoretical one at high SNR and deviates about $1\,dB$ at low SNR. The reason for this deviation at low SNR is the malfunction of the currently implemented synchronisation

algorithm in the low SNR region. Future work will include investigation of more sophisticated synchronisation structures. Furthermore, the data are decoded on the DSP currently, but will be moved to the FPGA and the effects of using the correlation properties of the different encodings will be investigated.

VI. Conclusion

The main challenge for RFID receivers is the detection of the receive signal under typically adversarial circumstances. The most crucial obstacles, being the strong self interference of the carrier, the wildly varying receive power of the tag response, and the ever present noise in industrial environments, have been tackled in this work. The proposed digital receiver can cope with all of this three mentioned sources of disturbance simultaneously, by setting its decision threshold in the I/Q plane adaptively to the estimated receive and noise power, and estimating the self interference of the carrier. The implementation of the proposed algorithm on a rapid prototyping board has been shown and extensive measurements have been undertaken. The measurements obtained for bit error ratio demonstrates the strength and near optimum performance of the receiver and substantiates the correct behaviour of the algorithm.

Acknowledgment

Our industrial partner Infineon Technologies has to be thanked for enabling this work and supporting us with many advices and helpful discussions. They also provided us with a selection of tag prototypes to enable our measurements. Moreover, we would like to thank Austrian Research Centers for support regarding the *SmartSim* rapid prototyping board.

References

[1] K. Finkenzeller, *RFID Handbook*. John Wiley and Sons LTD, 2004, second edition, ISBN 0-470-84402-7.

[2] C. Angerer, B. Knerr, M. Holzer, A. Adalan, and M.Rupp, "Flexible Simulation and Prototyping for RFID Designs," in *Procceedings of the First International EURASIP Workshop on RFID Technology*, September 2007.

[3] EPCGlobal, "EPC Global HF Air Interface Version 2, Document Version 0.1," November 2006.

[4] C. Mutti and A. Wittneben, "Robust Signal Detection in Passive RFID Systems," in *Proceedings of the First International EURASIP Workshop on RFID Technology, p.39 - 42*, September 2007.

[5] C. H. Y. Liu and H.Min, "Digital Correlation Demodulator Design for RFID Reader," in *Wireless Communications and Networking Conference, WCNC 2007, p.1664 - 1668*, March 2007.

[6] N. Choi, H. Lee, S. Lee, and S. Kim, "Design of a 13.56 MHz RFID System," in *Proceedings of the 8th International Conference on Advanced Communication Technology, Vol. 1, p.840 - 843*, October 2006.

[7] N. Roy, A. Trivedi, and J. Wong, "Designing an FPGA-Based RFID Reader," *XCell Journal*, pp. 26–29, Second Quater 2006.

[8] I. Mayordomo, R. Berenguer, D. Valderas, J. de No, I. Guruceaga, and I. Gutierrez, "A Passive RFID System Design and Analysis focused on Reader Performance," in *Conference on Design of Circuits and Integrated Systems, DCIS 2007*, November 2007.

[9] ISO / IEC, "ISO / IEC 15693, Identification Cards - Contactless Integrated Circuit Cards - Vicinity Cards," January 2000.