DISS. ETH NO. 22556

System Design and Channel Equalization for Evolved EDGE Physical Layer

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH
(Dr. sc. ETH Zurich)

presented by

STEFAN ALTORFER
MSc ETH
born on August 21, 1981
citizen of Glarus Nord, Switzerland

accepted on the recommendation of

Prof. Dr. Qiuting Huang, examiner
Prof. Dr. Andreas Burg, co-examiner

2015

Acknowledgments

My time at the IIS started already a long time ago with both a semester and a master's thesis which sparked my interest for wireless communications, digital VLSI design and FPGA prototyping. I then first deepened my understanding of both ASIC and FPGA design while working in industry before I took the opportunity and started my doctoral studies. I thereby enjoyed most the mix between algorithmic research, the design of VLSI architectures, and the collaboration with students who I hopefully have infected with my enthusiasm for hardware design.

I would like to thank my doctoral advisor Prof. Qiuting Huang for giving me the opportunity to work in such an interesting environment between academia and industry. The close project collaboration with ACP made it possible to get the developed ASICs manufactured and integrated into a cellular prototype system together with ACP's state-of-the-art RF transceivers. Furthermore, it has always been a great motivation for me to see that the research results will end up in an industrial product one day.

Further, I am deeply grateful to Prof. Andy Burg for co-examining my thesis. He contributed significantly to my enthusiasm for wireless communications and circuit design during my time at Celestrius and eventually managed to convince me to start my doctoral studies. He is certainly one of the most expert persons in the field of digital VLSI design for communications and we always highly appreciated his advice in the Evolved EDGE project.

I would also like to express my gratitude to Dr. Christian Benkeser who was the leader of the digital VLSI group of Prof. Huang for more than two years. It was very constructive to discuss simulation results,

hardware architectures and future plans with him and I always enjoyed his positive and motivating attitude.

Next, I would like to thank my fellow campaigners in the Evolved EDGE project, Harald Kröll and Benjamin Weber, for their significant contributions. The discussions with Harald about the most complex algorithms were always inspiring and his work on the interference suppression and on the embedded Linux in the PHY system very valuable. Benjamin made sure that our chip had a standard compliant channel decoder and helped a lot in the back-end design. Also many thanks go to the other digital guys in the analog group, Sandro Belfanti and Christoph Roth, for numerous fruitful discussions.

I am also grateful to Frank Gürkaynak and Beat Muheim from the DZ for setting up the excellent design flow and for their support with the back-end tools. Further, I would like to thank Hubert Käslin and Norbert Felber for their outstanding VLSI lecture series and for offering many students the unique opportunity of realizing an ASIC design project. Also many thanks go to the guys of ACP, especially to David Tschopp, for helping with the operation of the RF transceiver. Of the many students working under my supervision I would like to thank in particular Johannes Widmer who contributed significantly to the soft-output equalization and pre-filter VLSI architectures.

A big thank you also to all current and former colleagues from the IIS for contributing to this thesis in one way or the other. Especially the entertaining discussions and coffee breaks with my current office mates Pirmin, David and Michael were always a great motivation to come to the office every day. Moreover, I would like to thank Philipp Schönle and Tom Kleier for their help with the PCB design. I would also like to express my thanks to the IT support team of Christoph Wicki, Adam Feigin and Fredi Kunz and to the laboratory support of Hansjörg Gisler.

A special thank goes to my lunch jogging mates Markus and Kaspar. The common sporting activity during the lunch break was always so helpful to get a clear mind.

Finally, I wish to thank my parents and my wife Alexandra for their constant faith in me and for their enduring support.

Abstract

Evolved EDGE is an advancement of the cellular communication standard of the second generation (2G) with the goal of achieving higher spectral efficiency, better reliability and higher peak data rates. This progress ensures that 2G systems can continue to serve as a competitive fallback solution in places without 3G or 4G coverage. Furthermore, an enhancement of the GSM/EDGE standard consolidates its position as the most universal cellular solution. This is an important attribute since worldwide network coverage is a key requirement for many applications in the emerging field of machine-to-machine (M2M) communication. The significant increase in spectral efficiency of Evolved EDGE is realized by introducing modulation schemes with up to five bits per symbol. However, the high modulation order in combination with the severe inter-symbol interference inherent to 2G systems makes channel equalization and signal detection extremely complex. Equalizer solutions that can cope with this complexity while being small and power-efficient as required for M2M communication systems are most effectively realized by developing application specific integrated circuits (ASIC).

Optimal equalization algorithms like maximum-likelihood sequence estimation can only be implemented for legacy GSM modulation at reasonably low complexity. Equalization of Evolved EDGE signals requires reduced-complexity variants to achieve small and power-efficient designs. In this thesis, several of these algorithms are evaluated and optimized with regard to their suitability for an ASIC implementation. It is shown that reduced-state sequence estimation (RSSE) with only 16 trellis states in combination with minimum-phase channel shortening is sufficient to achieve close-to-optimal performance under

multipath fading as specified for 2G. The implemented channel shortening and RSSE solutions improve hardware efficiency of prior art by a factor 5 and 1.6, respectively. Moreover, a substantial performance gain is achieved by providing soft-values to the channel decoder, and the transmission reliability is improved by employing two receive antennas. A highly optimized hardware architecture that implements these two features is presented, boosting error-rate performance by up to 10 dB with hardware efficiency degradation of only a factor 2.5.

The feasibility of Evolved EDGE is demonstrated by presenting the first-ever baseband transceiver ASIC with a core area of 6 mm^2 in 130 nm CMOS. An integration of the ASIC into a complete physical layer (PHY) system allows assessing the sensitivity and interference performance of the whole processing chain, including RF transceiver. The measured sensitivity performance of the voice channel is among the best ever reported and the reference performance under fading conditions for the high modulation modes is exceeded by far. Furthermore, synchronization in time and frequency to a real-world base station is performed what demonstrates the completeness of the PHY system.

Zusammenfassung

Evolved EDGE ist eine Weiterentwicklung des Mobilfunkstandards der zweiten Generation (2G) mit dem Ziel einer besseren Ausnutzung des lizenzierten Spektrums, einer grösseren Zuverlässigkeit sowie einer höheren maximalen Datenrate. Dieser Fortschritt garantiert, dass das 2G System auch in Zukunft als konkurrenzfähige Backuplösung verwendet werden kann an Orten, die nicht durch 3G und 4G abgedeckt sind. Ausserdem festigt ein Ausbau des GSM/EDGE Standards seine Position als die weltweit meistverbreitetste Mobilfunklösung. Dies ist eine wichtige Eigenschaft, da weltweite Netzabdeckung eine wichtige Anforderung vieler Anwendungen im neu entstehenden Bereich der Machine-to-Machine (M2M) Kommunikation ist. Die merklich höhere Datenrate von Evolved EDGE wird erreicht indem bis zu fünf Bits pro Symbol übertragen werden. Allerdings macht diese hohe Modulationsordnung in Kombination mit der starken systemimmanenten Intersymbolinterferenz von 2G Systemen die Signaldetektion äusserst komplex. M2M Kommunikationssysteme erfordern kleine und energieeffiziente Empfänger, welche jedoch gleichzeitig die hohe Komplexität der Signaldetektion bewältigen müssen. Diese Anforderungen werden am effektivsten durch die Entwicklung von dedizierten integrierten Schaltungen (ASIC) erreicht.

Optimale Algorithmen für die Signaldetektion können nur für die binäre Übertragung des ursprünglichen GSM Systems mit vernünftiger Komplexität realisiert werden. Die Demodulation von Evolved EDGE Signalen erfordert Algorithmen mit reduzierter Komplexität um kleine und energieeffiziente Lösungen zu realisieren. In dieser Arbeit werden mehrere solche Algorithmen evaluiert und im Hinblick auf eine ASIC-Implementierung optimiert. Es stellt sich heraus, dass der

sogenannte Reduced-State Sequence Estimation (RSSE) Algorithmus mit nur 16 Trelliszuständen in Kombination mit minimalphasigem Kanalkürzungs-Filter genügt, um sehr nahe an die Leistung des optimalen Algorithmus zu gelangen. Die implementierten Architekturen des Kanalkürzungs-Filters und des RSSE-Detektors erreichen eine 5- respektive 1.6-mal bessere Hardware-Effizienz als der bisherige Stand der Technik. Darüber hinaus wird eine beträchtliche Leistungssteigerung erreicht, indem der Dekodierer zu jedem demodulierten Bit auch noch einen Wert über dessen Verlässlichkeit erhält, und die Zuverlässigkeit der Übertragung wird verbessert, wenn zwei Empfangsantennen eingesetzt werden. Eine äusserst optimierte Hardware Architektur, welche diese beiden Features beinhaltet, wird vorgestellt. Die Lösung verbessert die Fehlerrate um bis zu 10 dB und weist dabei eine nur 2.5-mal schlechtere Hardware-Effizienz auf.

Die Machbarkeit von Evolved EDGE wird durch die Präsentation des allerersten Basisband-Transceiver ASICs demonstriert. Der Chip wurde mit einer 130 nm CMOS Technologie hergestellt und benötigt eine Siliziumfläche von 6 mm^2. Durch die Integration des ASICs in ein komplettes Kommunikationssystem kann die Qualität der gesamten Signalverarbeitungskette, inklusive der Hochfrequenzbauteile, gemessen werden. Die gemessene Empfindlichkeit im Sprachmodus gehört zu den besten, die je publiziert wurden und die vom Standard vorgegebene Referenzleistung unter Fading-Bedingungen wird bei weitem übertroffen. Ausserdem wird die Vollständigkeit des ganzen Systems unter Beweis gestellt indem eine Zeit- und Frequenz-Synchronisation zu einer realen Basistation hergestellt wird.

Contents

Chapter 1

Introduction

The combination of modern wireless communication standards with powerful and feature-rich smartphones has led to the situation that people are nowadays used to access the Internet from wherever they are. There is an ongoing trend that not only personal computers and smartphones are connected to the Internet, but more and more also machines and everyday objects communicate over the Internet to a host computer where sensor data is processed and control information is sent back. This trend has originated the term *Internet of Things* which is aimed at the prediction that in the future everything will be connected to the Internet. According to [1], there will be 50 billion wireless devices connected to the Internet by 2020 and the total number of Internet devices could reach 500 billion at that time.

1.1 The Advance of M2M Communication

The capability of machines to communicate in some way or the other is nothing new. Also connecting devices or sensors to a network has been possible since a long time. However, it was only after the availability of inexpensive electronics, ubiquitous network coverage, and cloud computing that it has become feasible to equip basically any device with a communication module. This is the key enabler for machine-to-machine (M2M) communication where devices are connected to the

Internet by either a wired or wireless technology and interact with each other, with a central computer, or with a human being.

The field of application for M2M communication is very diverse and some prominent examples are briefly introduced in the following.

Smart Grid: An electricity network where the power consumption of the end-user is reported in real time is referred to as smart grid. The knowledge of the present power consumption allows the electrical supply company to predict the future demand more accurately and the network can be managed more efficiently. For example, one million GPRS equipped smart meters have been deployed to date in Sweden to accurately monitor the monthly electricity consumption [2].

Logistics: The tracking and reporting of the position along with various other parameters that could be damaging for the transported goods such as temperature, humidity or abrupt acceleration is important in logistic applications. The operating range of this application could be limited to within a factory or could be worldwide in case of global shipping companies.

Telemetry: This application refers to the process of taking automated measurements at remote or inaccessible places and to send the recorded data mostly over a wireless link to a host computer where it is evaluated.

E-Health: The monitoring of blood levels (e.g., glucose level of diabetes patients) and the automatic forwarding to a central computer where they are monitored by a doctor or from where a warning message is automatically sent to the smartphone of the patient could significantly reduce the reaction time in critical situations. Alternative E-Health applications are alarm systems for elderly or handicapped people. Equipping these systems with the possibility of voice- or video-feedback would significantly reduce the probability of a false alarm.

The list of possible M2M applications could be much longer what emphasizes the fact that the market potential of M2M communication

systems is immense. Especially for mobile network operators, the M2M market could be a means to further increase the number of subscriptions with low traffic demands in times of a saturated cell phone market and increasing traffic per flat-rate subscription.

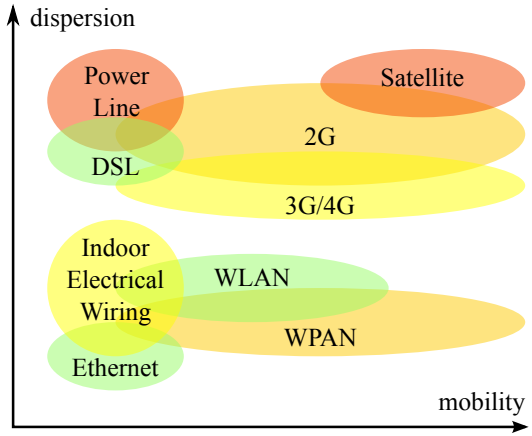


Figure 1.1: Potential M2M communication technologies grouped according to their suitability for mobile or stationary applications and whether the devices are concentrated at one location or dispersed. The colors indicate the maximal throughput of the technology (green=high, red=low).

As diverse the field of M2M applications is, as different are the demands on the technology used for the Internet connection. While the sensors of a smart grid are stationary, a tracked freight container must communicate its position from all over the world. Moreover, the throughput requirements are very different when comparing an alarm system capable of real-time video streaming and a smart meter which reports the power consumption once in a day or month. M2M applications can choose from a broad field of different technologies to fulfill their requirements in terms of throughput and mobility. Many of these technologies are shown in Figure 1.1 where they are grouped according to their capabilities regarding mobility and dispersion of the M2M devices. The mobility criterion refers to whether the device

is fixed in one location or constantly moving, whereas dispersion indicates whether all the devices of an M2M communication system are concentrated in one place or distributed over a large area. Furthermore, the color indicates the maximum achievable throughput of the corresponding technology. Green refers to technologies with a throughput of more than a Gbit/s, yellow refers to a throughput of several hundred Mbits/s, and orange is around one Mbit/s. Long-haul power line communication and satellite networks like Iridium offer typically a throughput even lower than that.

For many applications, a solution based on cellular communication is the perfect match. The advanced technology used in smartphones combined with small and robust SIM cards which can be directly soldered to the device, and the unlimited amount of IP addresses offered by IPv6 make it possible to equip basically any embedded system with a cellular connection to the Internet. Cellular networks provide both reasonably high data rates and good coverage - to a large extent also indoors. Even if the newly deployed fourth generation (4G) standard LTE (Long-Term Evolution) and also many 3G standards offer very high data rates, the well-established 2G EDGE (Enhanced Data rates for GSM Evolution) network has two main advantages when it comes to M2M solutions:

1. **Coverage:** GSM/EDGE is the only cellular standard which is available on all continents and which has ubiquitous coverage in many places, including rural areas (see Section 1.3). Remote telemetry sensors have often only 2G network coverage and logistics companies require devices that report their position from all over the world.

2. **Cost:** In EDGE systems, the frequency band is divided into 200 kHz slots which is very narrowband compared to 3G and 4G systems. The narrow bandwidth allows the design of inexpensive radio frequency (RF) transceivers what is key for an M2M solution to be deployed by the million. The broadband solution LTE requires much more sophisticated RF parts and is overkill for M2M applications that do not require high throughput.

A cost-effective solution for M2M devices is key to make EDGE the technology of choice even for more stationary applications where

also wired solutions are an option. The only way to achieve low cost in large quantities is the development of a 2G modem as an application specific integrated circuit (ASIC).

1.2 Cellular Network Coverage and the Need for Evolved EDGE

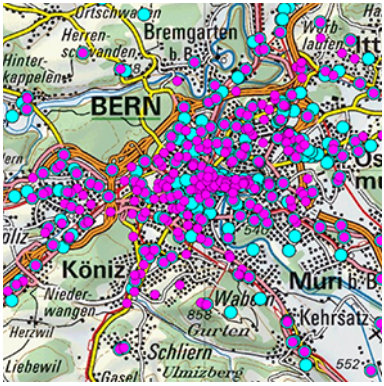
Since the launch of GSM in 1991, cellular communication systems have gradually evolved and provide ever higher data rates to the mobile user. The mobile phone industry has likewise evolved and has come up with smartphones which have become indispensable gadgets to many people. Applications running on these smartphones are often data hungry and use up all of the available bandwidth. In highly populated places, network providers invest in the latest technologies to cope with the enormous throughput requirements, but in more remote places and in less developed countries the well-established GSM/EDGE standard remains as a fallback solution. Whenever mobile users enter such areas and want to use their usual applications, they experience drastic performance degradation. A further development of the ubiquitous GSM/EDGE standard is therefore crucial to keep this performance gap within acceptable limits.

Such a development of the GSM/EDGE standard should not only increase the data rate of a single link by introducing higher throughput modes and reducing the end-to-end latency, but it should also enhance the overall system capacity by adding features to suppress interference from other users. Especially in urban areas, this interference is often the limiting factor and mobile stations with strong co-channel interference suppression capabilities would allow the network providers to use the available frequency band more efficiently.

For many M2M applications, a high peak throughput is only secondary, but a reliable link is of utmost importance. A connection to the Internet is required at all times and independently of external circumstances such as high mountains, rainy weather, closed doors, and so forth. In order to make GSM/EDGE a competitive alternative for M2M communication, it is important that the standard is extended with features that make the wireless link more reliable.

Another important factor for GSM/EDGE to become the M2M technology of choice is its universal availability. Figure 1.2 illustrates that even in Switzerland, some rural areas are only covered by 2G cells and 3G antennas are concentrated in urban areas. In the case of Switzerland, it is likely that these gaps are filled by investments in new 3G or 4G infrastructure, but the installation of new antennas is costly and takes time. In other parts of the world, ubiquitous 3G/4G coverage is economically not reasonable and an upgrade of the existing 2G infrastructure is much faster and cost-effective.

Due to the high price of licensed frequency bands, it is likely that some of the 2G radio spectrum is going to be reused by LTE networks and some cellular providers are even considering to decommission their 2G network altogether. In order to remain an attractive alternative to 3G/4G solutions and to avoid the loss of its ubiquitousness, it is very important that the GSM/EDGE standard evolves.



(a) An urban area with many 3G antennas.

(b) In rural areas, 2G antennas are often more frequent.

Figure 1.2: GSM/EDGE (blue) and 3G (red) antenna locations in an urban and a rural area [3].

1.3 Features of Evolved EDGE

In 2008, the 2.75G standard Evolved EDGE (E-EDGE) has been released with the goal of increasing the network capacity of 2G systems [4]. The standard is based on the same time- and frequency-division multiple access scheme as GSM/EDGE and can hence be realized with the same antenna and RF infrastructure. The bulk of the throughput increase comes at the expense of an increased receiver complexity. As shown in Figure 1.3 the gap to a moderately advanced 3G phone (HSDPA category 6) in terms of peak achievable downlink data rate is reduced from a factor 15 to a factor of 3 with E-EDGE. The increased data rate, the higher reliability, and the improved coverage offered by E-EDGE result from several new features which are briefly summarized in the following.

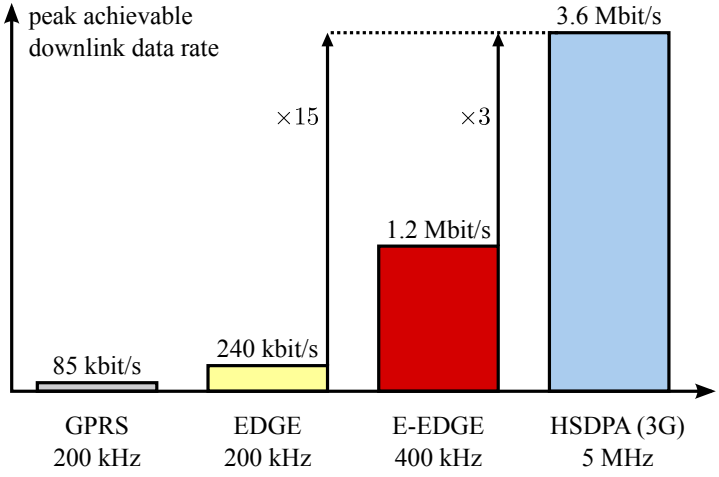


Figure 1.3: Maximal throughput of 2G and 3G systems

Downlink Dual Carrier: This feature mitigates the main drawback of GSM/EDGE: the narrow bandwidth of only 200 kHz. It gives the base station the possibility to assign two 200 kHz carriers to the same cell phone and thereby to double the downlink data rate. As

many cell phones already have two RF chains to support the MIMO communication of the newest LTE standard, the second carrier could be received without additional hardware cost. However, even if the throughput is doubled, the spectral efficiency is not improved as also the occupied bandwidth is doubled.

Latency Reduction: Increasing the downlink data rate alone does not automatically result in a reduced download time. In TCP/IP-based communication, every data packet needs to be acknowledged and faulty packets are retransmitted. Especially under bad channel conditions, many retransmissions are required and the over-full data buffers are becoming the limiting factor. The download time can be improved by reducing the end-to-end latency of the system. E-EDGE foresees two new mechanisms to address this problem: On the one hand, the transmission time is reduced (RTTI) by distributing a data block over only two instead of four frames. On the other hand, the mechanism of acknowledging packets is improved, leading to a faster retransmission of erroneous packets.

High-Order Modulation: The spectral efficiency is improved by introducing high-order modulation schemes. While the binary Gaussian minimum shift keying (GMSK) modulation is used in the original GSM system, the throughput has already been tripled by the introduction of 8-ary phase-shift keying (8-PSK) in EDGE (cf. Figure 1.3). E-EDGE foresees even higher modulation orders in combination with quadrature amplitude modulation (QAM). Employing 16-QAM and 32-QAM modulation allows a transmission of up to 5 bits per symbol within the same bandwidth. Obviously, the higher throughput comes at the cost of increased receiver complexity and a sufficiently high signal-to-noise ratio (SNR) is required.

Turbo Codes: The transmission of signals with high modulation order is susceptible to bit-errors. Hence, the high-order modulation schemes are protected with a more powerful forward error correction scheme. As opposed to EDGE, where convolutional channel coding is applied, the new schemes employ Turbo codes. A Turbo code is a parallel concatenation of two convolutional codes and an iterative

decoding process is required in the receiver. Transmission schemes with Turbo codes have the potential to closely approach the Shannon capacity.

Receive Diversity: Employing two receive antennas to gain two independent observations of the same transmitted signal does not bring any throughput gain at first sight. However, the overall network capacity can be significantly improved, as the so-called RX-diversity makes the transmission more robust against co-channel interference. Furthermore, the receiver can benefit from an SNR improvement originating from the array gain and the probability of a successful transmission is increased as the probability that both wireless channels are in a deep fade is smaller compared to the case with only one antenna. This significantly improves the system reliability and minimizes the risk of a link disruption.

Some of the features like latency reduction and downlink dual-carrier are straightforward to implement, but the complexity of channel equalization and signal detection grows exponentially with the number of transmitted bits per symbol and with the number of symbols affected by inter-symbol interference (ISI). The narrow bandwidth of 2G systems has its advantages for frequency allocation and for the design of RF transceivers, but the bandwidth being below the sampling rate results in system-inherent ISI. In combination with the multipath wireless channel, the ISI can extend to over 7 symbols. Finding equalizer solutions that can successfully detect 32-QAM modulated signals under severe ISI with a complexity that enables an implementation of dedicated hardware suitable for small and battery-operated devices is a formidable research challenge.

1.4 Contributions

This thesis aims at the development of high-performance channel equalization solutions for single-carrier communication systems with high modulation orders. To this end, the most promising algorithms are optimized with regard to their suitability for hardware implementation, resulting in VLSI circuits with an outstanding hardware

efficiency when compared to prior art. Based on these new equalization circuits, the first ever physical layer (PHY) transceiver ASIC for E-EDGE is presented. Furthermore, the ASIC is embedded into a complete PHY system allowing for a demonstration of its capabilities under real-world conditions and in real time.

Algorithm Evaluation and Optimization

It has been shown in [5] that a linear channel shortening pre-filter in combination with reduced-state sequence estimation (RSSE) [6] is a viable solution for the equalization of 8-PSK signals in an EDGE system. A solution for the higher modulation orders of E-EDGE is presented by Benkeser in [7]. However, the full flexibility of RSSE is neither exploited in his work nor anywhere else in the literature.

- In this thesis, the optimal RSSE configuration for 16-QAM and 32-QAM transmission over standardized GSM channel models is derived and it is shown that 16 states are sufficient to achieve close-to-optimal performance.

- In order to address the channel shortening part of the equalization process, a novel minimum-phase pre-filter algorithm based on homomorphic filtering in the cepstrum domain and direct computation of the filter coefficients is presented. The algorithm is optimized such that hardware-costly complex-valued divisions and exponentials are avoided at the cost of a transformation from Cartesian to polar number representation, what can be efficiently realized in hardware.

- Calculating soft-outputs and exploiting RX-diversity both significantly improves the block-error rate performance at the cost of more complex signal processing in the receiver. In [8], a low-complexity version of the optimal MAP detection algorithm is presented for a binary modulation scheme. In this thesis, it is shown that this approach is also suitable for higher modulation orders in combination with RSSE at the cost of surprisingly low performance degradation. Furthermore, the soft-output RSSE is extended with the possibility to combine the two received data streams in an optimal way.

- As an alternative to the reduced-state equalization, an approach based on the M-algorithm (MA) [9] is developed for soft-output equalization. To this end, the full trellis is considered, but only a limited number of paths are retained at each stage. It is shown that a slightly better performance can be achieved by the soft-output MA compared to the soft-output RSSE with even less trellis states.

VLSI Circuits for Channel Equalization

In order to demonstrate the hardware suitability of the optimized algorithms, they are implemented in VLSI.

- The architecture of the novel pre-filter algorithm is optimized for low silicon area leading to an improvement of 5× in terms of area-throughput ratio compared to prior-art.

- The computationally most intensive part in trellis-based equalization is the branch metric computation. The number of operations can be highly reduced by pre-computation and storage of partial reference symbols. In this thesis, the approach of [7] is applied to the more general case of RSSE and the tradeoff between computational and storage complexity is analyzed. In combination with the optimal RSSE configuration for 32-QAM, the pre-computation concepts leads to a 1.6 times more efficient VLSI solution which even shows a slightly better bit-error rate performance when compared to the only previously published E-EDGE equalizer [10].

- VLSI circuits for the two soft-output equalization candidates are presented, both of them supporting RX-diversity. A comparison of the two solutions shows a clear advantage of the soft-output RSSE in terms of hardware efficiency.

PHY System

The efficient VLSI solutions of pre-filter and soft-output RSSE build the core of the first-ever Evolved EDGE transceiver ASIC. With a silicon area of 6 mm^2 in a 130 nm CMOS technology it shows a much

smaller footprint compared to other 2G modem solutions which are mainly based on processors [11]. The ASIC requires only 39 mW to receive the highest throughput mode of E-EDGE with enabled RX-diversity. Together with a state-of-the-art RF transceiver the ASIC builds a complete PHY system which is successfully running in a real time testbed. Synchronization to a real-world base-station demonstrates the completeness of the system. Furthermore, the sensitivity performance measurements confirm the high algorithmic quality by outperforming prior-art and by exceeding the 3GPP reference performance by far.

1.5 Thesis Outline

The remainder of this thesis is organized as follows. In Chapter 2, the baseband system model is introduced and the optimal equalization and signal detection algorithms for ISI-affected systems are reviewed and reduced-complexity variants capable of handling the high modulation orders of E-EDGE are presented. The chapter concludes by highlighting the benefits of soft-output equalization and RX-diversity.

Chapter 3 is dedicated to the VLSI implementations of E-EDGE equalization algorithms. To this end, the algorithms of Chapter 2 are adjusted to the case of E-EDGE and optimized with focus on their suitability for an implementation in hardware. VLSI solutions for hard- and soft-output equalizers and for channel shortening are presented and compared among each other and to prior art.

In Chapter 4, the E-EDGE transceiver ASIC is presented in detail and it is shown how it is integrated into a complete PHY system. The acquired block-error rate measurements confirm the accuracy of the simulation framework used during the algorithm evaluation and demonstrate the good sensitivity performance.

Finally, Chapter 5 concludes the thesis and provides an outlook on the next steps towards a single-chip E-EDGE modem with minimal area as required for applications in M2M communication.

Chapter 2

Equalization Algorithms for Evolved EDGE

Data transmission in cellular networks is a challenge since the impairments by the wireless channel are numerous and vary quickly over time. Furthermore, the usage of licensed frequency bands is expensive and a high spectral efficiency is of paramount importance. In GSM/EDGE systems, the frequency band is divided into 200 kHz slots but in favor of a higher throughput, the symbol rate has been chosen as high as 271 kHz what means that the system suffers inherently from ISI. The optimal equalization algorithm for ISI affected signals is maximum likelihood sequence estimation which is explained in Section 2.2.

To further improve the spectral efficiency, E-EDGE foresees higher-order modulation schemes which increase the number of transmitted bits in the same bandwidth. However, the strong ISI remains unchanged what makes channel equalization and symbol detection extremely challenging. This challenge is addressed by employing reduced-state sequence estimation which is introduced in Section 2.3 and which is well suited for an implementation in small and power efficient dedicated hardware as required for mobile devices. The key for a good performance is the smart reduction of the trellis in combination with pre-filtering as suggested in [6, 12] and explained in Section 2.4.

The performance is significantly improved by providing reliability information, so-called soft-outputs, to the channel decoder. Two different trellis-based soft-output equalization algorithms with reduced complexity are introduced in Sections 2.5 and 2.6. In Section 2.7, it is shown how the RX-diversity feature can be exploited to achieve a superior performance. All equalization algorithms require information about the wireless channel and hence Section 2.8 is about the estimation of channel state information.

2.1 E-EDGE System Model

Throughout this chapter, the equivalent baseband E-EDGE transmission system depicted in Figure 2.1 is considered. All parameters are defined in a generic form and the specific values for E-EDGE are summarized in Table 2.1. The system consists of three main parts: transmitter, multipath channel and receiver. They are all described in the following.

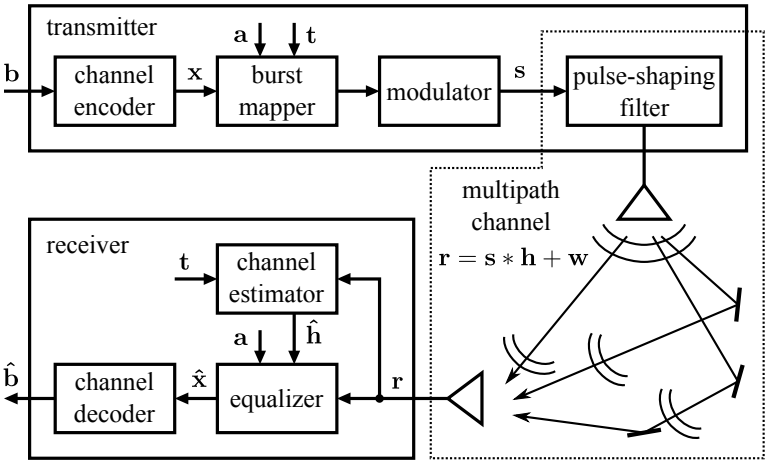


Figure 2.1: Baseband model of an E-EDGE cellular system transmitting over a wireless multipath channel.

2.1.1 Transmitter

A block of data, denoted by the binary-valued B-dimensional vector $\mathbf{b}$, is extended by redundancy information to allow for forward error correction in the receiver. The actual encoding process adds a fixed percentage of redundant bits, but practically any coding rate $R \leq 1$ can be achieved by removing bits in a predefined manner. This process is referred to as puncturing. Forward error correction is only effective if the erroneous bits do not occur in series. Therefore, the encoded bits are interleaved in such a way that neighboring bits after encoding are sent over the channel well apart from each other. Hence, the probability that all these bits are incorrectly transmitted is reduced, since the channel conditions typically vary a lot over the interleaving interval. The channel encoder transforms the input block $\mathbf{b}$ into a binary-valued vector $\mathbf{x}$ of length B/R.

The encoded block $\mathbf{x}$ is then mapped to Z different bursts as described in the GSM/EDGE specification [13]. Usually, block rectangular interleaving is employed and $\mathbf{x}$ is mapped to $Z = 4$ bursts. Each burst contains two parts of $N = 58$ information symbols and the number of bits per symbol is denoted by $Q \in \{1, 3, 4, 5\}$. Hence, each burst contains $2NQ$ encoded bits denoted by $x_{q,k} \in \{+1, -1\}$ with $q = 1, \ldots, Q$ being the bit index within the symbol and $k = 1, \ldots, 2N$ being the symbol index within the burst. Between the two parts of information symbols, a training sequence $\mathbf{t}$ is inserted and each burst starts and ends with predefined tail symbols $\mathbf{a}$ as shown in Figure 2.2. The mapping of encoded bits $x_{q,k'}$ to information symbols s_k of the first burst is as follows:

$$s_k = \begin{cases} [x_{1,k-3} \ldots x_{Q,k-3}] & \text{for } k = 4 \ldots 61, \\ [x_{1,k-29} \ldots x_{Q,k-29}] & \text{for } k = 88 \ldots 145. \end{cases}$$

The remaining encoded bits are mapped to symbols of the other three bursts accordingly.

Subsequently, the modulator maps all symbols to different points of a complex-valued symbol alphabet denoted by the set $\mathcal{O}$ with cardinality $|\mathcal{O}| = 2^Q$ at a rate of $f_{\text{sym}} = 271\,\text{kHz}$. As already mentioned in Section 1.3, the original modulation of the GSM system is GMSK. For this modulation scheme, the output of the channel encoder is first differentially encoded and the modulating data is then passed

3 tail sym.	58 information symbols	26 training sym.	58 information symbols	3 tail sym.
a_1	$[x_{1,1} \dots x_{Q,1}]$	t_1	$[x_{1,59} \dots x_{Q,59}]$	a_1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
a_3	$[x_{1,58} \dots x_{Q,58}]$	t_{26}	$[x_{1,116} \dots x_{Q,116}]$	a_3

Figure 2.2: Mapping of information bits, training sequence and tail symbols to GSM burst as specified in [13].

through a filter with Gaussian impulse response to limit the signal bandwidth to 200 kHz before it is used to modulate the frequency. For $Q = 3$, 8-ary phase-shift keying (8-PSK) is used and for the high modulation orders with $Q = 4$ and $Q = 5$, quadrature amplitude modulation is employed (16-QAM/32-QAM). The modulated PSK and QAM symbols are filtered with a pulse-shaping filter which is specified such that the spectral shape of the transmitted signal is similar to that of GMSK. Since this filtering step is a linear process, the pulse-shaping filter is seen as a part of the channel and the vector $\mathbf{s}$ at the modulator output is considered as the transmitted signal. The modulation process is described in more detail in the 3GPP specification [14].

2.1.2 Channel

E-EDGE is completely based on the GSM infrastructure and hence the same channel models are employed. Figure 2.3 shows the power-delay profiles of the *typical urban* (TU) and *hilly terrain* (HT) channel models which are predominantly used in this thesis. While the delay spread of the TU channel is not much more than a symbol duration of 3.7 μs, the latest echoes of the HT channel arrive with a delay of more than 5 symbols. Moreover, the pulse-shaping filter in the transmitter further broadens the overall channel impulse response (CIR) to a total length of $L = 8$ symbols as illustrated in Figure 2.3. Both channels are modeled by a Rayleigh fading process. Throughout this thesis ideal frequency hopping is assumed, meaning that the channel realizations of subsequent bursts fade independently of each other.

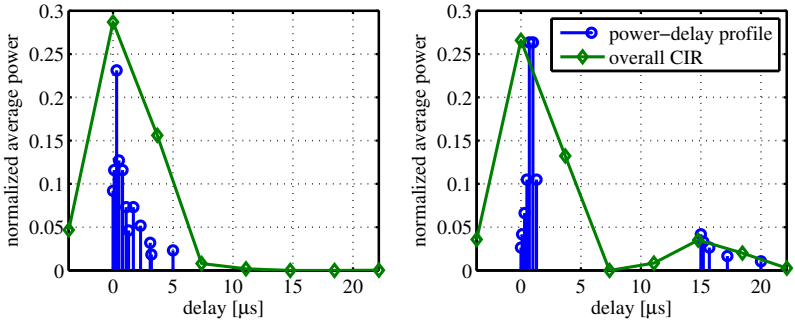


Figure 2.3: Power-delay profile and overall CIR at symbol rate including pulse-shaping filter of typical urban (left) and hilly terrain (right) GSM channels as specified in [15].

The overall multipath channel in the baseband is given by the L-dimensional complex-valued vector $\mathbf{h}$ and the input-output relationship of the channel is defined as

$$r_k = \sum_{m=0}^{L-1} s_{k-m} h_m + w_k \,, \tag{2.1}$$

where the sum represents the convolution of $\mathbf{h}$ with the transmitted signal $\mathbf{s}$ at time index k and w_k is an element of an N-dimensional circularly symmetric complex Gaussian distributed noise vector $\mathbf{w}$ with variance σ^2, i.e., $\mathbf{w} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I}_N)$. As easily seen by the summation in (2.1), the channel introduces inter-symbol interference (ISI) over $L-1$ symbols.

2.1.3 Receiver

In order to obtain an estimate $\hat{\mathbf{b}}$ of the original input sequence, the following three blocks are required: channel estimator, equalizer and channel decoder.

The channel estimator delivers an estimate $\hat{\mathbf{h}}$ of the CIR by comparing the received midamble with the transmitted version $\mathbf{t}$ which is known to the receiver.

The received information symbols are passed to the equalizer where the effect of the channel is undone and an estimate of the encoded bit sequence $\mathbf{x}$ is provided to the channel decoder. The known tail symbols $\mathbf{a}$ can be used to initialize the estimation process. The equalizer can either deliver hard-outputs, quantized to ± 1 or calculate soft-outputs which contain reliability information about every estimated bit.

The result of the equalizer is then first reordered according to the predefined interleaving scheme before neutral information is reinserted at the positions of the punctured bits. Finally, the redundant information is exploited to obtain an estimate $\hat{\mathbf{b}}$ of the original input block.

Maximal code-block size	$B_{\max} = 658$
Coding rate	$0.33 \leq R \leq 1$
Bursts per code block	$Z = 4$
Bits per symbol	$Q \in \{1, 3, 4, 5\}$
Information symbols per half-burst	$N = 58$
Length of CIR at symbol rate	$L = 8$
Symbol rate [kHz]	$f_{\text{sym}} = 271$

Table 2.1: System model parameters for E-EDGE.

2.1.4 Performance Measures

The performances of the receiver algorithms are measured by different means. On the one hand, the uncoded bit-error rate (UBER) refers to the quality of uncoded transmission and is given by the number of bit errors in $\hat{\mathbf{x}}$ divided by the length of $\mathbf{x}$. On the other hand, the quality of the complete system is determined by the block-error rate (BLER) which is given by the rate of not successfully decoded blocks $\hat{\mathbf{b}}$. Note that an E-EDGE data block can consist of several individually encoded parts plus a header part and a block error occurs if a bit of any of these parts is wrong.

2.2 Maximum Likelihood Sequence Estimation

Channel equalization refers to the process of undoing the effect of the channel and to deliver an estimate of the transmitted symbol sequence, denoted by $\hat{\mathbf{s}}$. As there is a one-to-one mapping between $\mathbf{x}$ and $\mathbf{s}$, the estimated bit sequence $\hat{\mathbf{x}}$ follows immediately. The estimation strategy which maximizes the probability that the estimated symbol sequence $\hat{\mathbf{s}}$ is equal to the transmitted version $\mathbf{s}'$ is known as maximum likelihood sequence estimation (MLSE) and the estimate is given by

$$\hat{\mathbf{s}} = \underset{\mathbf{s}\in\mathcal{O}^N}{\arg\max} \left\{ \mathrm{p}(\mathbf{r} \,|\, \mathbf{s}' = \mathbf{s}, \mathbf{h}) \right\} . \tag{2.2}$$

Since the information symbols of a burst are divided into two parts by the training sequence (cf. Figure 2.2), the estimation is done separately for both parts, where each part consists of N symbols. Due to the circularly-symmetric Gaussian distribution of the noise vector $\mathbf{w}$, the conditional probability density function of the received signal of one half-burst is given by [16]

$$\mathrm{p}(\mathbf{r} \,|\, \mathbf{s}' = \mathbf{s}, \mathbf{h}) = \frac{1}{(\pi\sigma^2)^N} \exp\left(\frac{-\|\mathbf{r} - \mathbf{s} * \mathbf{h}\|^2}{\sigma^2} \right) , \tag{2.3}$$

where $\|\mathbf{d}\|$ denotes the ℓ^2-norm of vector $\mathbf{d}$. Since $\log(x)$ is monotonically increasing in x, the estimate (2.2) is equivalent to

$$\hat{\mathbf{s}} = \underset{\mathbf{s}\in\mathcal{O}^N}{\arg\min} \|\mathbf{r} - \mathbf{s} * \mathbf{h}\|^2 . \tag{2.4}$$

An illustration of the intersymbol interference channel (2.1) is depicted in Figure 2.4. Every received sample r_k is a linear combination of the last L transmitted symbols plus noise w_k. Hence, the channel can be seen as a state machine with the transmitted symbols s_k as input, the noiseless received samples $\tilde{r}_k$ as output, and $|\mathcal{O}|^{L-1}$ different states, defined by the $L-1$ symbols in the channel memory. Unrolling the state machine in time results in a trellis diagram as shown in Figure 2.5. States are denoted by $c \in \mathcal{C}_k$, where the set $\mathcal{C}_k$ contains all trellis states at time k. Branches are given by state tuples (c', c), where $c' \in \mathcal{C}_{k-1}$ is the predecessor state of $c \in \mathcal{C}_k$.

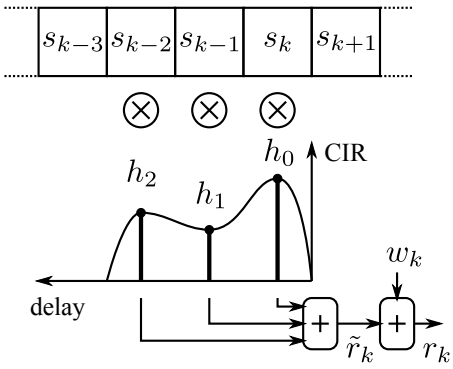


Figure 2.4: Illustration of the baseband multipath channel for $L = 3$.

The problem of (2.4) can be efficiently solved with the Viterbi algorithm [17, 18]. The squared vector norm in (2.4) is thereby iteratively calculated by traversing through the trellis and adding up the branch metrics (BM)

$$\Gamma_k(c', c) = \left| r_k - \sum_{m=0}^{L-1} s_{k-m} h_m \right|^2 , \tag{2.5}$$

with the symbols $s_{k-(L-1)} \dots s_{k-1}$ defined by state c', and symbol s_k defined by the branch (c', c). Whenever different paths merge into a state c, the best path is selected based on a path metric which is the sum of the BM of the incoming branch $\Gamma_k(c', c)$ and the state metric of the predecessor state $A_{k-1}(c')$. The new state metric is given as the minimal metric of all paths merging into the state c

$$A_k(c) = \min_{c' \in \chi(c)} A_{k-1}(c') + \Gamma_k(c', c) , \tag{2.6}$$

where $\chi(c)$ is the set of all predecessor states of c. The branch associated with the smallest metric is denoted as the winning branch and once the end of the trellis is reached, the symbols corresponding to the ML solution are found by tracing back through the trellis along the winning branches.

A reasonable measure for the algorithmic complexity is the number of branch metric calculations (2.5). Solving the ML problem by

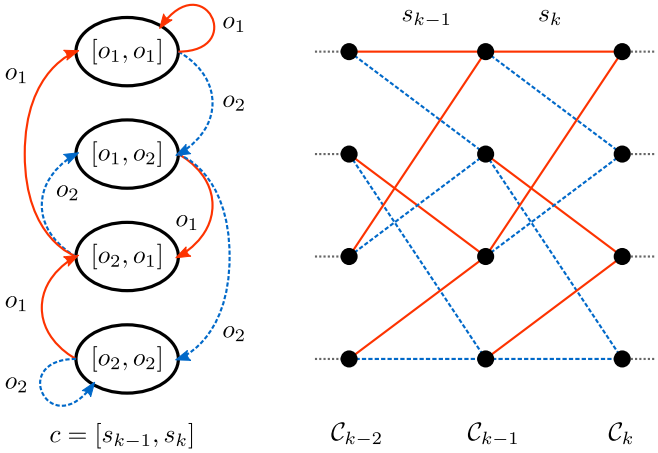


Figure 2.5: The multipath channel with $L = 3$ as state machine. With a symbol alphabet $\mathcal{O} = \{o_1, o_2\}$, the state machine and the resulting trellis have $|\mathcal{O}|^{L-1} = 4$ states.

performing an exhaustive search over all possible symbol sequences would require $|\mathcal{O}|^N = 2^{QN}$ BM calculations, where N is the sequence length and Q is the number of bits per symbol. By using the Viterbi algorithm, the complexity is reduced to $N|\mathcal{O}|^L = N2^{QL}$ calculations, where L is the CIR length. For GSM receivers ($Q = 1$, $L = 8$ and $N = 58$), the complexity of MLSE is still feasible, but due to the exponential growth in Q, MLSE is prohibitively complex for the high order modulation schemes of E-EDGE.

2.3 Reduced-State Sequence Estimation

In order to keep the equalization complexity within realistic boundaries, Eyuboglu et al. suggested in [6] an algorithm which combines the optimal MLSE with decision-feedback equalization. Compared to MLSE, reduced-state sequence estimation (RSSE) reduces the number of trellis states by dividing the symbol alphabet into subsets and defining the trellis on these subsets. The division of the $|\mathcal{O}|$ symbols

into J_m subsets is optimally chosen such that the Euclidean distance of symbols within the same subset is maximal (e.g. through Ungerboeck partitioning [19]). The number of subsets J_m with $1 \leq m \leq L-1$ can be different for every symbol s_{k-m} that would define the MLSE trellis state, but $|\mathcal{O}|$ must always be a multiple of J_m and the condition $J_1 \geq J_2 \ldots \geq J_{L-1}$ must hold. Furthermore, a reduction in the number of subsets must result from merging several subsets. The subsets are denoted by $\mathcal{P}^i_{k-m}$ where $i = 1 \ldots J_m$ is the subset index, k is the index of the trellis section, and $m = 1 \ldots L-1$ corresponds to the position of the symbol in the channel memory. Naturally, the number of symbols in each subset is given by $|\mathcal{P}^i_{k-m}| = |\mathcal{O}|/J_m$ and the number of trellis states is given by

$$K = \prod_{m=1}^{L-1} J_m \,. \tag{2.7}$$

Whenever a winning path is selected, a decision between symbols of the same subset is done immediately, but the selection among subsets is postponed. Dividing the symbol alphabet into more subsets results on the one hand in a larger Euclidean distance between the symbols of the same subset and hence in a reduced probability for wrong decisions. On the other hand, the number of trellis states is increased. Each state keeps a list of the decisions within a subset and together with the subset indices given by the state itself all symbols required in the branch metric computation (2.5) are uniquely defined. RSSE can be seen as a sort of decision feedback equalization where the decisions are more reliable at the cost of trellis processing.

An Example

Figure 2.6 shows a fragment of an RSSE trellis for 16-QAM ($Q = 4$) with $L = 4$, $J_1 = 4$, $J_2 = 2$ and $J_3 = 2$. The example shows all predecessor states of a specific state $c \in \mathcal{C}_k$ defined by the subsets $\left[\mathcal{P}^1_{k-2}, \mathcal{P}^1_{k-1}, \mathcal{P}^3_k\right]$. The symbols belonging to a specific subset are encircled in the small I/Q diagram of the complete 16-QAM symbol alphabet. Since $J_1 \neq |\mathcal{O}|$, a total of four parallel branches connect every predecessor state with c. During the selection process in (2.6), not only the best predecessor state is selected, but also the best symbol

from the subset $\mathcal{P}_k^3$ is immediately determined. This immediate decision for the most recent symbol is kept in a list together with decisions for older symbols which are inherited from the winning predecessor state. In the example in Figure 2.6, the selected symbols within a subset are marked with a red cross. The complexity reduction of RSSE comes from the fact that all combinations for symbols with a circle but without a cross are not considered anymore.

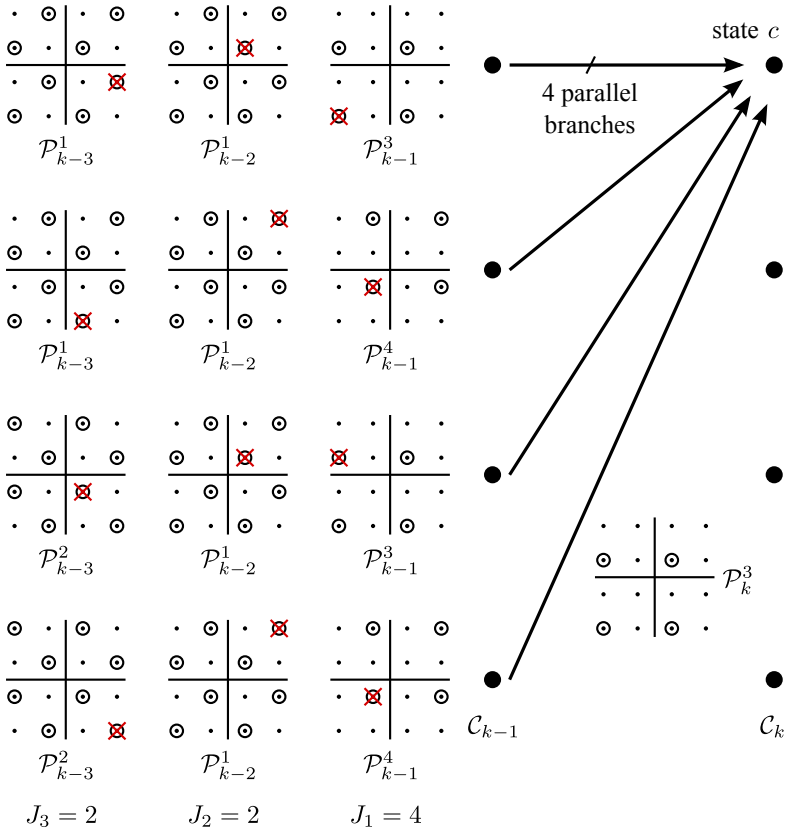


Figure 2.6: Fragment of an example trellis for RSSE with $Q = 4$ (16-QAM) and $L = 4$.

RSSE Special Cases

By choosing the J_m in a special way, RSSE turns into algorithms with different names.

- If $J_m = |\mathcal{O}|$ for all m, then this is equivalent to MLSE.
- Choosing $J_m = 1$ for all m results in a trellis with just one state and the algorithm is equivalent to decision feedback equalization.
- If $J_m = |\mathcal{O}|$ for $1 \leq m \leq D$ and $J_m = 1$ for $D < m \leq L-1$, the algorithm is called delayed decision feedback sequence estimation (DDFSE) and has been suggested by Duel-Hallen and Heegard in [12].

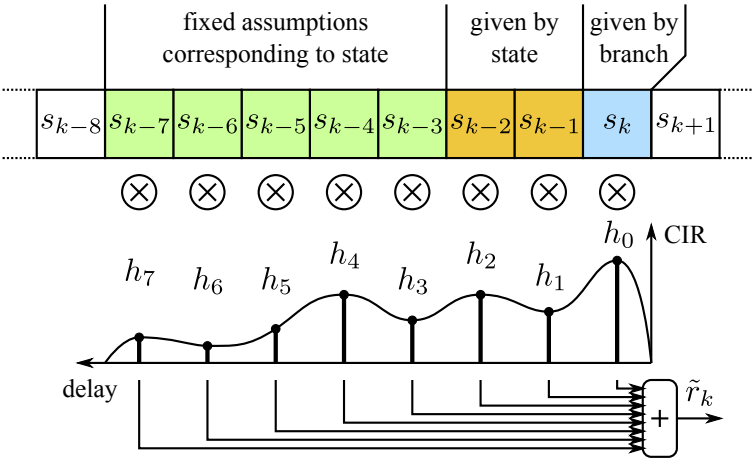


Figure 2.7: Reference symbol computation for DDFSE with $D = 2$ and channel length $L = 8$.

The DDFSE algorithm is a bit simpler than RSSE as no subset partitioning is required and it is well suited to illustrate the complexity

reduction compared to MLSE. Figure 2.7 illustrates the computation of a reference symbol

$$\tilde{r}_k = \sum_{m=0}^{L-1} s_{k-m} h_m \tag{2.8}$$

which is used in the branch metric computation (2.5). For the most recent $D+1$ symbols, all possible combinations are evaluated, resulting in a total of $|\mathcal{O}|^{D+1}$ BM computations per trellis stage and in $|\mathcal{O}|^{D}$ trellis states. Furthermore, every trellis state keeps a list of the most probable symbols s_{k-D-1} to s_{k-L+1} which are called survivor symbols and which are used to complete the sum in the computation of $\tilde{r}_k$. By choosing $D < L-1$, the number of MLSE branch metric computations is reduced from $N2^{QL}$ to $N2^{Q(D+1)}$ in the case of DDFSE.

However, DDFSE has the drawback that, for high modulation orders, the number of trellis states can be adjusted only very coarsely and choosing $D = 1$ is often the only feasible option. With RSSE the complexity can be adjusted much more fine-grained and often a better performance is achieved with less trellis states (see Section 3.2.2).

2.4 RSSE in Combination with Pre-Filter

The complexity reduction of RSSE comes at the price of decreased error-rate performance. The best performance is achieved if the CIR has strong first taps, since the survivor symbols are determined based on trellis processing on the most recent symbols only. If the first channel taps h_m ($0 \leq m \leq D$ in case of DDFSE) are strong, then the more important symbols s_{k-m}, for which all possible combinations are tested, get more weight in the branch metric computation (2.5). Alternatively, if the first channel taps are weak, then the effect of trellis processing vanishes, since the symbols which are defined by the trellis state are weighted with a small h_m and do not contribute much to the branch metric.

Normally, the first taps of the CIR are not the strongest, especially because the pulse shaping filter is considered as part of the overall channel. In this case, a linear channel shortening filter can be employed to transform the overall CIR in such a way that the first

taps are the strongest. However, the filter should have an all-pass characteristic to prevent the white noise contained in the received samples from becoming correlated.

Of all causal and stable filter realizations $\mathbf{h}$ with the same magnitude response $|\mathbf{g}|$, the minimum-phase filter $\mathbf{h}^{\min}$ has minimum group delay, meaning that $\sum_{m=p}^{L-1} |h_m^{\min}|^2$ is minimized for all $p \in [0, L-1]$. As a consequence, the energy of $\mathbf{h}^{\min}$ is concentrated near the start of the impulse response.

A linear pre-filter that transforms the CIR $\mathbf{h}$ to its minimum-phase equivalent $\mathbf{h}^{\min}$ provides exactly the desired behavior: The strong channel taps are moved to the front of the CIR without changing the magnitude response of $\mathbf{h}$. One approach to compute the coefficients of such a pre-filter is linear prediction (LP) [20]. It has been shown in [21] that the LP approach is suitable for integration in dedicated hardware. However, in Section 3.1.1, an even more efficient approach based on homomorphic filtering is presented.

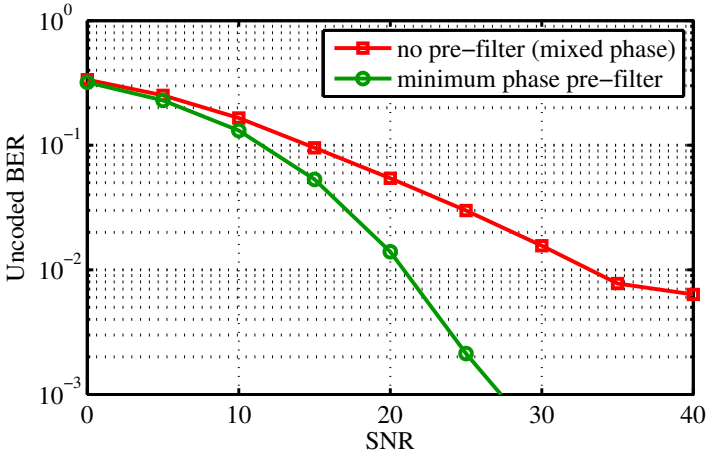


Figure 2.8: Uncoded BER over an HT channel for 8-PSK modulation and an 8-state DDFSE equalizer ($D = 1$).

The positive effect of transforming the channel into its minimum-phase equivalent can be seen in Figure 2.8, where the UBER of an

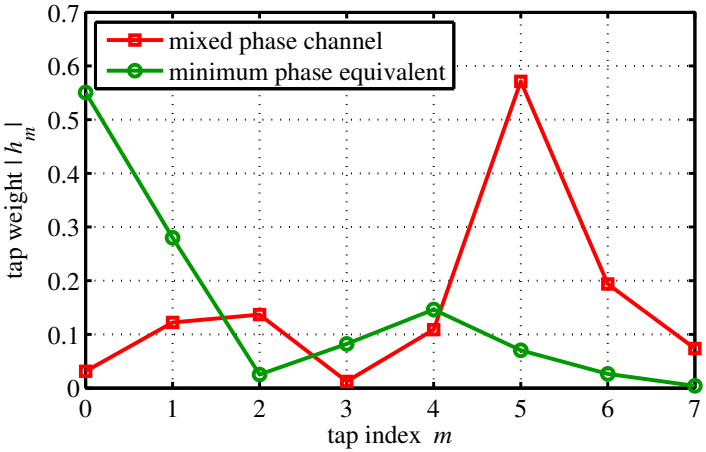


Figure 2.9: An HT channel realization and its minimum-phase equivalent after pre-filtering.

8-PSK transmission over an HT channel is shown for both a mixed-phase and a minimum-phase channel with equal magnitude response. Figure 2.9 illustrates how an HT channel realization is transformed to its minimum-phase equivalent. In the illustrated example, the pre-filtering is particularly beneficial as the first few channel taps of the original CIR are very weak.

2.5 Optimal Soft-Output Equalization

The block error-rate performance can be significantly improved if the equalizer delivers reliability information along with the de-mapped bits to the channel decoder. The decoder can then exploit this a priori information about the encoded bits $\mathbf{x}$ to improve the estimate of the original code block $\mathbf{b}$. The reliability information is typically expressed in terms of log-likelihood ratios (LLRs). As the name suggests, an LLR is the natural logarithm of the ratio of the probabilities of a bit being $+1$ or -1.

$$L(x_{q,k}|\mathbf{r}) = \log\left(\frac{P[x_{q,k}=+1|\mathbf{r}]}{P[x_{q,k}=-1|\mathbf{r}]}\right) \tag{2.9}$$

The LLRs are conditioned on the received vector $\mathbf{r}$ and are hence called *a-posteriori* LLRs.

An algorithm that computes these LLRs in an efficient way has been introduced in 1974 by Bahl et al. [22] and is known either as BCJR or as maximum a-posteriori (MAP) algorithm. The algorithm is similar to MLSE (cf. Section 2.2) as it is based on the same trellis, but is more complex as it involves a forward and a backward recursion through the trellis.

The bit probabilities in (2.9) can be expressed as a sum of branch probabilities

$$P[x_{q,k}=\pm 1|\mathbf{r}] = \sum_{(c',c)\in\mathcal{B}_{x_{q,k}}^{\pm 1}} P[C_{k-1}=c', C_k=c, \mathbf{r}]\,, \tag{2.10}$$

where $\mathcal{B}_{x_{q,k}}^{\pm 1}$ is the set of all branches between the trellis states c' and c with $x_{q,k}=\pm 1$. Furthermore, the branch probabilities can be rewritten as

$$\begin{aligned} &P[C_{k-1}=c', C_k=c, \mathbf{r}] = \\ &\underbrace{P[C_{k-1}=c', \mathbf{r}_{(1,k-1)}]}_{\alpha_{k-1}(c')}\,\underbrace{P[C_k=c, r_k|C_{k-1}=c']}_{\gamma_k(c',c)}\,\underbrace{P[\mathbf{r}_{(k+1,N)}|C_k=c]}_{\beta_k(c)}\,, \end{aligned} \tag{2.11}$$

where $\mathbf{r}_{(a,b)}$ stands for the part of the vector $\mathbf{r}$ from element a to b.

The term $\gamma_k(c',c)$ corresponds to the probability that the trellis is in state c and the observed signal is r_k, given the trellis was in state c' at time $k-1$. Since the received signal is disturbed by white Gaussian noise, this probability is given by

$$\begin{aligned} \gamma_k(c',c) &= P[C_k=c, r_k|C_{k-1}=c'] \\ &= \frac{1}{\pi\sigma^2}\exp\left(\frac{-|r_k-\tilde{r}_k|^2}{\sigma^2}\right)\,, \end{aligned} \tag{2.12}$$

where $\tilde{r}_k$ is the reference signal of (2.8), resulting from the symbol sequence given by the states c' and c. The factor before the exponential can be left out as it would be reduced from the fraction (2.9) anyway.

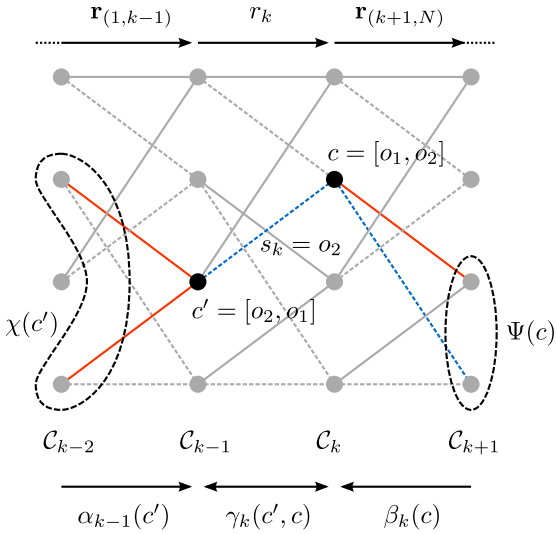


Figure 2.10: Section of BCJR-trellis for $L = 3$ and $\mathcal{O} = \{o_1, o_2\}$ with forward state metrics α_{k-1}, backward state metrics β_k, as well as branch metrics γ_k.

The term $\alpha_{k-1}(c')$ is the probability that the trellis is in state c' and the received sequence up to this point is $\mathbf{r}_{(1,k-1)}$. The main feature of the BCJR algorithm is the fact that this probability can be calculated iteratively during the forward recursion according to

$$\alpha_k(c) = \sum_{c' \in \chi(c)} \alpha_{k-1}(c') \gamma_k(c', c)\,, \tag{2.13}$$

where $\chi(c)$ is the set of all predecessor states that connect to c.

Similarly, the β_k values are calculated during the backward recursion according to

$$\beta_{k-1}(c') = \sum_{c \in \Psi(c')} \beta_k(c) \gamma_k(c', c)\,, \tag{2.14}$$

with $\Psi(c')$ being the set of all successor states that connect to c'. The $\beta_k(c)$ values represent the probability that the future received

sequence from $k+1$ until the end of the burst will be $\mathbf{r}_{(k+1,N)}$ given the trellis is in state c at time k.

Figure 2.10 illustrates the just now introduced terms for a trellis with $L = 3$ and a binary symbol alphabet $\mathcal{O} = \{o_1, o_2\}$ (see also Figure 2.5 where the same example is used for MLSE).

2.6 Soft-Output Algorithms with Reduced Complexity

Due to its high computational complexity, the MAP algorithm is seldom implemented in its original form. Many alternatives exist to reduce the algorithmic complexity at only little performance loss. Some of them are discussed in the following.

2.6.1 Max-Log Approximation

The complexity of the MAP algorithm can be drastically reduced by doing the forward and backward recursion in the logarithmic domain and by applying the Max-Log approximation

$$\log\left(\sum_i \exp(d_i)\right) \approx \max_i(d_i)\,. \tag{2.15}$$

To this end, the branch metric in the (negative) logarithmic domain is defined as

$$\Gamma_k(c', c) = -\log(\gamma_k(c', c)) = \frac{|r_k - \tilde{r}_k|^2}{\sigma^2}\,, \tag{2.16}$$

making the computation much simpler, as no exponential function is involved anymore.

The forward state metric is defined accordingly as

$$\begin{aligned} A_k(c) &= -\log(\alpha_k(c)) \\ &= -\log\left(\sum_{c' \in \chi(c)} \alpha_{k-1}(c')\gamma_k(c', c)\right) \end{aligned}$$

$$= \quad -\log\left(\sum_{c'\in\chi(c)} \exp(-A_{k-1}(c') - \Gamma_k(c',c))\right) \quad (2.17)$$

and is approximated according to (2.15) by

$$\begin{aligned} A_k(c) &\approx -\max_{c'\in\chi(c)} -A_{k-1}(c') - \Gamma_k(c',c) \\ &= \min_{c'\in\chi(c)} A_{k-1}(c') + \Gamma_k(c',c)\,. \end{aligned} \quad (2.18)$$

Equivalently, the backward state metric is given as

$$B_{k-1}(c') \approx \min_{c\in\Psi(c')} B_k(c) + \Gamma_k(c',c)\,. \quad (2.19)$$

Finally, the LLRs are approximately given by

$$\begin{aligned} L(x_{q,k}|\mathbf{r}) &= \log\left(\frac{\sum_{(c',c)\in\mathcal{B}^{+1}_{x_{q,k}}} \alpha_{k-1}(c')\gamma_k(c',c)\beta_k(c)}{\sum_{(c',c)\in\mathcal{B}^{-1}_{x_{q,k}}} \alpha_{k-1}(c')\gamma_k(c',c)\beta_k(c)}\right) \\ &\approx \min_{(c',c)\in\mathcal{B}^{-1}_{x_{q,k}}} A_{k-1}(c') + \Gamma_k(c',c) + B_k(c) \\ &- \min_{(c',c)\in\mathcal{B}^{+1}_{x_{q,k}}} A_{k-1}(c') + \Gamma_k(c',c) + B_k(c)\,. \end{aligned} \quad (2.20)$$

The forward recursion of the Max-Log MAP algorithm[1] is equivalent to the MLSE and the reliability information comes at the cost of the backward recursion and the final LLR computation.

2.6.2 Reduced-State BCJR

Even with the max-log approximation, the BCJR algorithm is still far too complex for high modulation orders as its complexity grows exponentially with the number of bits per symbol (cf. Section 2.2). As for DDFSE and RSSE, the complexity can be reduced by merging trellis states by means of subset partitioning of the symbol alphabet. During the forward recursion, only the best symbol within a subset survives. For the backward recursion, several possibilities for the survivor selection exist:

[1] Actually, the algorithm should be called Min-Log MAP in this context as the branch metrics are defined in the negative logarithmic domain, but the name stems from channel decoding applications where it is used as Max-Log MAP.

- New survivors are selected during the backward recursion, independently of the forward recursion.
- The survivors that have been found during the forward recursion are used for the backward recursion.
- Instead of storing the survivors of the forward recursion, the branch metrics Γ_k are stored and reused during the backward recursion.

Especially for minimum-phase channels, the preliminary decisions during the forward recursion are quite reliable and it makes sense to reuse the survivor map in the backward recursion. Furthermore, the branch metric computation involves many costly multiplications. Hence, storing and reusing the Γ_k values reduces the computational complexity a lot. This type of algorithm is denoted by RS Max-Log MAP in the BLER performance comparison in Figure 2.12.

As suggested in [8], the backward recursion can be omitted altogether. Since the algorithm is then very similar to the RSSE algorithm, it is denoted as Soft-Output RSSE (SO-RSSE). Surprisingly, the performance degradation is small for a reduced-state trellis and a minimum phase channel, but the complexity reduction is immense. By omitting the backward recursion, the LLR computation (2.9) is slightly changed and the LLRs are computed with a decision delay of D symbols [8], where $D < L$ is the number of symbols which compose a trellis state (cf. DDFSE in Section 2.3).

$$\begin{aligned} L(x_{q,k-D}|\mathbf{r}) \quad &\approx \quad \min_{(c',c)\in\mathcal{B}^{-1}_{x_{q,k-D}}} A_{k-1}(c') + \Gamma_k(c',c) \\ &- \quad \min_{(c',c)\in\mathcal{B}^{+1}_{x_{q,k-D}}} A_{k-1}(c') + \Gamma_k(c',c) \end{aligned} \tag{2.21}$$

Figure 2.11 exemplifies the LLR computation for a trellis with two bits per symbol ($Q = 2$) and a decision delay of one symbol ($D = 1$) by illustrating the sets of branches that contribute to the $+1$ group $\mathcal{B}^{+1}_{x_{q,k-D}}$ in (2.21). For the LLR computation of the first bit ($q = 1$, left part of Figure 2.11), all paths with $x_{1,k-1} = +1$ in the binary representation of symbol s_{k-1} form the set of branches $\mathcal{B}^{+1}_{x_{1,k-1}}$ which is used to group the metrics at trellis stage k. The set $\mathcal{B}^{+1}_{x_{2,k-1}}$ is used

for the LLRs of the second bit and contains other branches as shown in the right part of the figure.

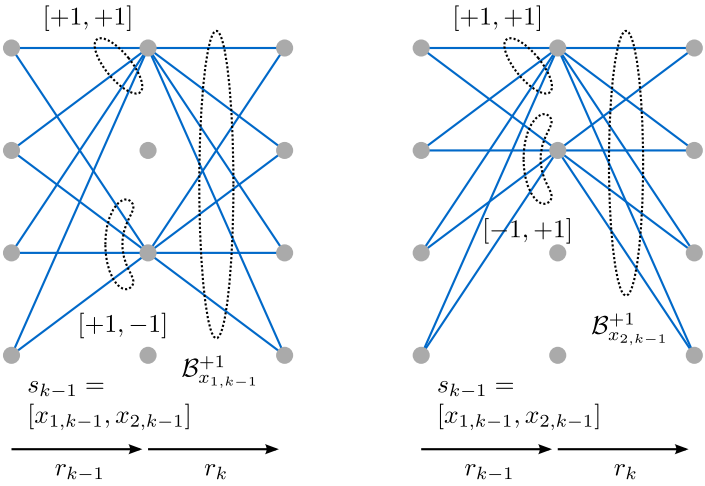


Figure 2.11: Illustration of the branches contributing to the +1 group $\mathcal{B}^{+1}_{x_{q,k-D}}$ for the first bit ($q = 1$, left) and the second bit ($q = 2$, right) in the LLR computation (2.21).

All algorithmic simplifications presented in this section result in a degraded BLER performance which is shown in Figure 2.12. The simulation has been done over a fading channel with three complex-valued, equally strong, randomly drawn channel taps ($L = 3$). Before equalization, a pre-filter has been employed to bring the channel to its minimum-phase equivalent. The used modulation and coding scheme is DAS5 with 8-PSK modulation and a coding rate of $R = 0.37$. The optimal MAP algorithm and the Max-Log MAP approximation employ a trellis with $2^{Q(L-1)} = 64$ states, while the reduced-state algorithms use only 8 states ($D = 1$) in combination with decision feedback. Naturally, the performance losses of the reduced-state variants depend a lot on the strength of the channel tap(s) for which no trellis processing is done.

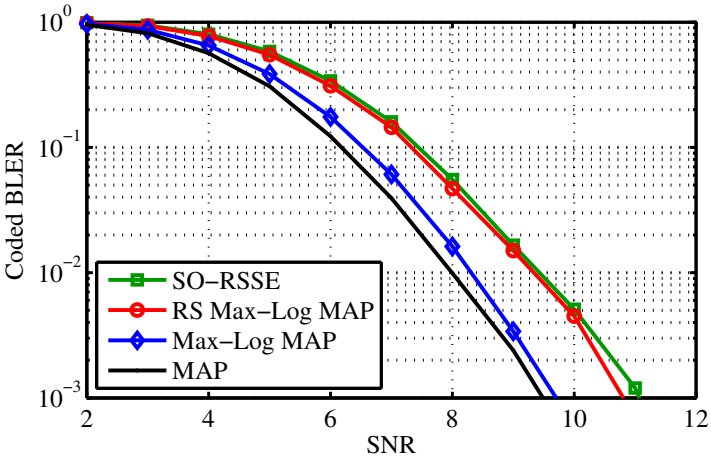


Figure 2.12: Performance loss due to algorithmic approximations for soft-output equalizers (3-tap channel, 8-PSK).

2.6.3 M-Algorithm

An alternative way to reduce the algorithmic complexity is to retain the full trellis, but to examine only a limited number of M paths. In the context of trellis-based channel equalization or decoding, this type of algorithm is typically called M-algorithm [9], while for tree-based MIMO detection problems, a similar breadth-first strategy is known as K-best algorithm [23]. At each trellis stage k, at most $|\mathcal{O}|M$ trellis states are reduced to M by selecting the states with the best state metric A_k. If the concept of the M-algorithm is applied to the BCJR algorithm, then plenty of possibilities arise for the backward recursion and the final LLR computation. In [24] it is suggested to do the backward recursion only over the part of the trellis that has survived during the forward recursion. If the backward recursion is allowed to build a completely independent subtree, then the LLR computation (2.20) can be done only over the intersection of the forward and backward subtree and this intersection could be empty. Fertonani in [25] suggests doing the completion over the union of the two subtrees thereby setting the missing α or β values to a fixed large value.

Since the omitting of the backward recursion yields good results in the case of SO-RSSE, it is obvious to do the same thing for the M-algorithm and to apply (2.21) to obtain the LLRs. The problem with this approach is that all the surviving M paths tend to converge quickly and the symbol s_{k-D} is often the same for all remaining paths. As a result, either the set $\mathcal{B}^{+1}_{x_{q,k-D}}$ or $\mathcal{B}^{-1}_{x_{q,k-D}}$ used in (2.21) remains empty and the LLR values are not defined.

To circumvent this problem, also the discarded paths could be kept until the LLRs are computed as suggested in [26]. However, for high modulation orders, this approach requires a large storage capacity. Another solution to this problem is a constrained selection of the M survivor paths such that none of the two sets will be empty. To this end, the selection of the M surviving paths is as follows:

1. Select the state with minimal state metric $A_k(c)$. The most recent D symbols of the symbol sequence that defines the state c yield a binary vector $\mathbf{x} = [x_{1,k-D+1}, \dots x_{Q,k-D+1}, \dots x_{Q,k}]$ of length QD.

2. For all positions i in $\mathbf{x}$, select the state with the smallest metric $A_k(c^\star)$ with $c^\star$ such that the ith bit is different from x_i. Note that the same state could fulfill several bit constraints at once.

3. Of the remaining states, select the ones with the best metric to complete the list of the M surviving paths.

The decision delay D determines how many of the M places could be occupied by states with non-minimal metric. If all the best states have e.g. a $+1$ on all bit positions, then the best state with $x_i = -1$ is selected for all bit positions i. In the worst case, a different state is selected for all i and the conformance to the constraint takes QD of the M positions. Choosing a large D means that the probability of finding the most likely symbol sequence is lower, but it also means that the chance of finding a good counter-hypothesis is higher what makes the LLRs more accurate. Simulations have shown that choosing $D = 1$ yields a sufficient LLR quality. Hence, at most Q places are occupied by non-optimal paths.

An advantage of the M-algorithm is that the number of trellis states can be chosen almost arbitrarily with the parameter M while for

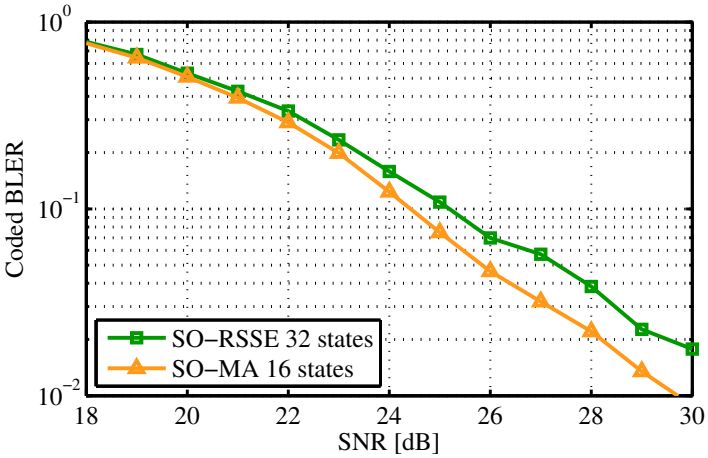


Figure 2.13: BLER performance comparison between SO-RSSE and SO-MA for 32-QAM modulated transmission over HT channel.

SO-RSSE it must be a power of the modulation order. For 32-QAM, a trellis with 32 states is the only reasonable option for SO-RSSE, but a slightly better performance can be achieved by the soft-output M-algorithm (SO-MA) with less states. Figure 2.13 shows that in the case of a channel with late reflections (HT channel profile), the SO-MA with $M = 16$ and $D = 1$ outperforms the SO-RSSE by approximately 1 dB in SNR with half the number of trellis states. However, under different channel conditions and for other modulation orders the two algorithms perform almost equally.

2.7 Exploiting Receive Diversity

The advance of wireless MIMO systems has not stopped at cellular communications and nowadays many mobile devices have two antennas. Therefore, the 3GPP specification of E-EDGE foresees two-fold receive diversity as an optional feature. Even though this setup cannot be called MIMO communication as only one data stream is transmitted, the receiver still benefits from two advantages:

- **Array gain**: Since the additive noise on the two received signals is independent, the SNR can be increased by coherently combining the two streams. The achievable SNR gain is thereby proportional to the number of receive antennas.
- **Diversity gain**: By transmitting the same signal over two independently fading channels, the probability that both channels are in a deep fade is highly reduced and the probability of a successful transmission is increased.

The system model of Figure 2.1 is now adapted as shown in Figure 2.14. A correlation factor ρ between the two streams and a gain imbalance G is introduced as specified in [15].

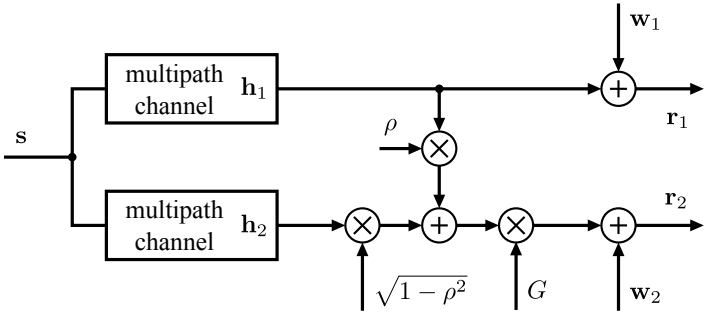


Figure 2.14: System model for two receive antennas as specified in [15].

2.7.1 Combining the Streams

The simplest approach to handle the two received streams is to select the better and discard the worse stream. To prevent useless operations on the discarded stream, the selection should occur as early as possible in the processing chain, for example based on a measurement of the received signal strength. Like this, the baseband signal processing effort is almost identical to the single-stream case, but the transmission is more reliable since it fails only if both streams experience bad channel conditions. However, no array gain is achieved with this solution as one stream is simply discarded.

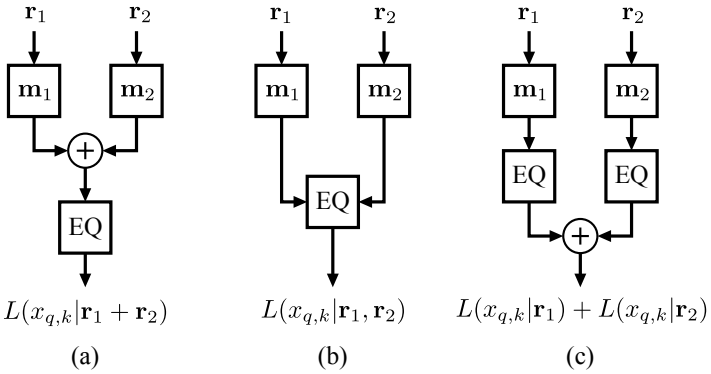


Figure 2.15: Three possible strategies for the combination of the two received signals: before (a), in (b), and after (c) the equalizer (EQ).

For an actual combination of the two streams, several approaches exist and we distinguish three of them as illustrated in Figure 2.15, where the streams are combined *before*, *after* and *in* the equalizer [27]. On the one hand, the earlier the streams are combined, the less parallel processing is required. On the other hand, some combination methods work only well under certain channel conditions as will become clear in the following.

Summation Before Equalizer

For a channel with only additive noise, i.e., $\mathbf{h}_1 = \mathbf{h}_2 = 1$, summing up the two received signals without any preprocessing is sufficient and results in an array gain of factor two (3 dB). The optimal combining method for a fading channel with a single complex-valued tap ($\mathbf{h}_1 = h_1$ and $\mathbf{h}_2 = h_2$) is called maximum-ratio combining (MRC) [28] and is achieved by multiplying the first stream with h_1^* and the second stream with h_2^* before summation. This multiplication has two effects: First, the phases of the two received signals are aligned such that they are coherently combined. Second, the streams are weighted according to their respective amplitude, giving the stronger signal more weight and diminishing the impact of the weaker signal.

In case of multipath fading channels, the MRC approach can be applied in a more general form by passing both streams through a matched filter with coefficients matched to the CIR of the respective stream before they are summed up. This approach is equivalent to the setup in Figure 2.15 (a) where the filter coefficients of the two FIR filters $\mathbf{m}_1$ and $\mathbf{m}_2$ are chosen such that $m_{1,k} = h^*_{1,L-k}$ and $m_{2,k} = h^*_{2,L-k}$ with k being the coefficient index. However, this solution exacerbates the task of the equalizer as the length of the overall channel including the filters $\mathbf{m}_1$ and $\mathbf{m}_2$ is almost doubled. Furthermore, the phase alignment between the two matched filter outputs is not optimal, since only the central taps of the symmetric overall channel add up coherently and all other taps could lead to destructive interference.

As an alternative to matched filtering, the coefficients of $\mathbf{m}_1$ and $\mathbf{m}_2$ can be chosen according to the minimum-phase criterion of the overall channel (see Section 2.4). This approach is more suitable for reduced-complexity trellis-based sequence estimation in the equalizer. Owing to the minimum-phase property, the first taps of the overall channel are phase-aligned, but all other taps are not what can again lead to destructive interference. In addition, the combining does not adhere to the maximum-ratio principle.

Hence, it can be stated that for multipath fading channels, a summation of the two streams before the equalizer as depicted in Figure 2.15 (a) is not a promising approach, even if the streams are preprocessed according to the matched-filter or the minimum-phase criterion.

Summation After Equalizer

In the approach depicted in Figure 2.15 (c), both streams are independently processed until after the equalizer and the LLRs are summed up. As long as the noise on the two streams is uncorrelated, this combining might yield good results, but from a processing effort point of view, the independent equalization of the two streams is clearly a waste of resources.

Combining In Equalizer

The optimal solution in terms of a-posteriori probability is to combine the two streams in the equalizer as shown in Figure 2.15 (b) and to condition the probabilities in the LLR formula (2.9) on both received vectors $\mathbf{r}_1$ and $\mathbf{r}_2$. The branch metric (2.12) is then given by

$$\begin{aligned} \gamma_k(c', c) &= P[C_k = c, r_{k,1}, r_{k,2} | C_{k-1} = c'] \\ &= \frac{1}{\pi^2 \det(\mathbf{R})} \exp\left(-\mathbf{d}_k^H \mathbf{R}^{-1} \mathbf{d}_k\right) \end{aligned} \tag{2.22}$$

as it now depends on two observations which are disturbed by noise which is distributed according to $[w_{1,k}, w_{2,k}]^T \sim \mathcal{CN}(\mathbf{0}, \mathbf{R})$, where $\mathbf{R}$ is the spatial covariance matrix of the noise. The column vector $\mathbf{d}_k$ is given by the differences between the reference signal and the received signal of the corresponding stream

$$\mathbf{d}_k = \begin{bmatrix} r_{1,k} - \tilde{r}_{1,k} \\ r_{2,k} - \tilde{r}_{2,k} \end{bmatrix}. \tag{2.23}$$

Finally, the branch metric in negative logarithmic domain is given by

$$\begin{aligned} \Gamma_k(c', c) &= \mathbf{d}_k^H \mathbf{U} \mathbf{d}_k \\ &= u_{11} |d_{1,k}|^2 + 2\Re(u_{12} d_{1,k}^* d_{2,k}) + u_{22} |d_{2,k}|^2, \end{aligned} \tag{2.24}$$

where the Hermitian matrix $\mathbf{U}$ denotes the inverse covariance matrix

$$\mathbf{U} = \mathbf{R}^{-1} = \begin{bmatrix} u_{11} & u_{12} \\ u_{12}^* & u_{22} \end{bmatrix}. \tag{2.25}$$

In case of uncorrelated noise with variances σ_1^2 and σ_2^2, the off-diagonal elements of $\mathbf{U}$ are zero and the branch metric is given by $|d_{1,k}|^2/\sigma_1^2 + |d_{2,k}|^2/\sigma_2^2$ what is basically the sum of the two terms that would result in the single stream case.

Receive Diversity Performance Gain

The performance gain in terms of block-error rate which is obtained from employing two receive antennas is illustrated in Figure 2.16. An 8-PSK modulation scheme and Max-Log MAP equalization that

combines the streams in the branch metric computation as in (2.23) has been used for the simulation.

Transmission over a static channel disturbed only with additive white Gaussian noise (AWGN) yields fast dropping BLER curves and the array gain resulting from the second observation is as expected exactly 3 dB as shown in Figure 2.16.

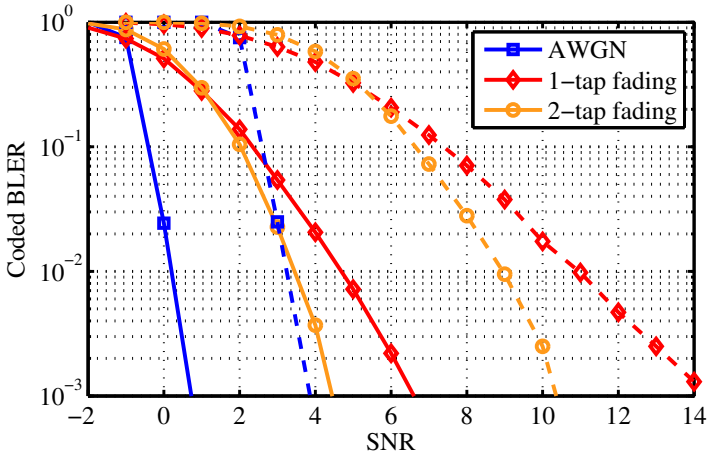


Figure 2.16: Improvement of block-error rate through receive diversity for 8-PSK modulation and Max-Log MAP equalization. Dashed lines: single-stream, solid lines: receive diversity.

The problem of fading is illustrated by the very flat BLER curve of the single-stream transmission over a one-tap fading channel, where the fluctuating received signal level leads to occasional block errors even if the average SNR is quite high. The fading problem is slightly less severe in the case of a multipath channel with two independent channel taps, since the probability that all channel taps are weak is reduced. Similarly to the case of two receive antennas, this setup results in two observations of the same transmitted symbol, even if the observations arrive shifted in time and interfere with other symbols. However, with the employed trellis-based Max-Log MAP equalization, ISI over only one symbol can be perfectly handled and the temporal

diversity gain yields steeper BLER curves compared to the single-tap channel.

By employing twofold receive diversity, the fading problem is alleviated as illustrated by the BLER curves in Figure 2.16 which are steeper in the case of two antennas. The independently fading channels reduce the risk that the total received signal level is low. Furthermore, the array gain shifts the receive diversity curves even further to the left.

2.7.2 Interference Suppression by Space-Time Filtering

Besides ISI, co-channel interference (CCI) of neighboring cells is the major source of interference in GSM/EDGE systems. Trellis-based equalization is the optimal vehicle to combat ISI, but CCI is best suppressed by an MMSE filter. In [29] Liang and Paulraj propose a two-stage approach to handle CCI and ISI separately.

The known training sequence convolved with the estimated CIR is used as a reference signal to derive the filter coefficients of a space-time filter (STF) as depicted in Figure 2.17 by an MMSE criterion. Since the reference signal contains the CIR, this first stage suppresses only CCI. The channel is then re-estimated, resulting in a better CIR estimate as the level of CCI has been reduced by the STF. Finally, the ISI distorted signals $\bar{\mathbf{r}}_1$ and $\bar{\mathbf{r}}_2$ are passed to a trellis-based equalizer.

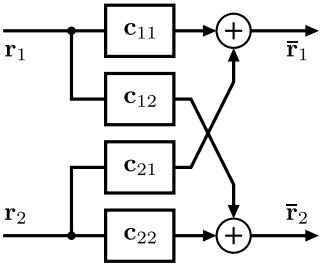


Figure 2.17: Structure of space-time filter used for CCI suppression consisting of four FIR filters with coefficients $\mathbf{c}_{i,j}$.

2.8 Estimation of Channel State Information

Many of the aforementioned algorithms require information about the transmission channel, also known as channel state information (CSI). As the wireless channel in a cellular environment is rapidly changing, the parameters need to be re-estimated for every burst. To this end, every burst contains in its middle a training sequence of 26 symbols $\mathbf{t} = [t_1 \dots t_{26}]$ which is known to the receiver (cf. Figure 2.2).

2.8.1 Channel Estimation

The most essential CSI parameter for channel equalization is the CIR. To accommodate the pulse-shaping filter plus the longest specified channel model where the latest reflections arrive at 20 μs, the number of estimated CIR taps is chosen as $L = 8$ (cf. Figures 2.3 and 2.9). Even if the received version of the training sequence is 26 symbols long, the first $L - 1$ symbols contain interference from unknown data symbols. Hence, only 19 symbols can be used to estimate the CIR.

The most prominent estimation methods are based on correlation or on the least squares technique. These two approaches are compared in [30] and the least squares approach yields clearly the better estimate. The estimate is determined by finding the $\hat{\mathbf{h}}$ which minimizes $\|\mathbf{r}_t - \mathbf{T}\hat{\mathbf{h}}\|^2$, where $\mathbf{r}_t = [r_{t_8} \dots r_{t_{26}}]^T$ is the portion of the received sequence that can be used for channel estimation and

$$\mathbf{T} = \begin{bmatrix} t_8 & t_7 & \dots & t_1 \\ t_9 & t_8 & \dots & t_2 \\ \vdots & \vdots & \ddots & \vdots \\ t_{26} & t_{25} & \dots & t_{19} \end{bmatrix} \tag{2.26}$$

is a convolution matrix containing time-reversed shifted versions of the training sequence such that $\mathbf{Th}$ represents the convolution of the training sequence with the CIR. The solution to this least squares problem is well known and given by

$$\hat{\mathbf{h}} = (\mathbf{T}^H\mathbf{T})^{-1}\mathbf{T}^H\mathbf{r}_t\,. \tag{2.27}$$

2.8.2 Noise Statistics Estimation

Another important CSI parameter is the statistical distribution of the additive noise which is used in the branch metric computation (2.22). Neither the noise variance σ^2 nor the spatial covariance matrix $\mathbf{R}$ is known to the receiver and must be estimated. The covariance matrix $\mathbf{R}$ of a two-element random vector $\mathbf{v} = [v_1, v_2]^T$ is defined by

$$\mathbf{R} = \mathbb{E}[(\mathbf{v} - \mathbb{E}[\mathbf{v}])(\mathbf{v} - \mathbb{E}[\mathbf{v}])^H] \,, \tag{2.28}$$

where $\mathbb{E}[\cdot]$ denotes the expectation. For the estimation of $\mathbf{R}$, the expectation operator is approximated by taking the average of several observations of the random vector. Since the additive noise is assumed to be zero-mean, the estimated covariance matrix is then given by

$$\hat{\mathbf{R}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{d}_t \mathbf{d}_t^H \,, \tag{2.29}$$

with $\mathbf{d}_t$ being the difference terms as defined in (2.23) corresponding to the received version of the known training sequence. These difference terms are used as observations of the noise. Since an estimate of the CIR must be available for the computation of the difference terms, the quality of $\hat{\mathbf{R}}$ depends also on the channel estimation error. As explained in Section 2.8.1, the number of usable training sequence symbols in an EDGE system is given by $T = 19$.

2.8.3 Performance Degradation by CSI Estimation

Needless to say that the estimation process inserts a certain amount of error into the CSI and the BLER performance is degraded compared to the ideal case with perfectly known CSI. Figure 2.18 illustrates the impact of CSI estimation on an 8-PSK transmission over a 2-tap fading channel with Max-Log MAP equalization.

The least squares estimation of the CIR shifts the BLER curves approximately 2 dB to the right, both in the single stream and in the receive diversity scenario. In the single stream scenario, a noise variance estimation error leads to incorrectly scaled LLRs and the resulting BLER degradation is small. In the receive diversity scenario however, an incorrect covariance matrix $\mathbf{R}$ leads to errors in the

branch metric computation and the performance loss compared to the case of perfectly known CSI is larger. If the covariance matrix is estimated based on an error-affected CIR estimate, then the BLER degradation is even more severe. In conclusion it can be stated that the estimation of CSI degrades the BLER performance stronger in case of receive diversity than it does for a single antenna case, leading to a reduction of the array gain.

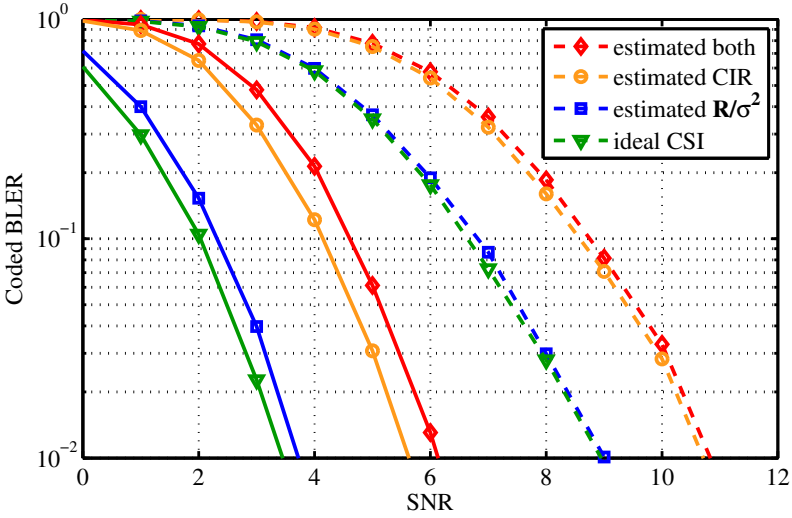


Figure 2.18: Impact of CSI estimation on BLER performance. Dashed lines correspond to the single antenna scenario, solid lines to the receive diversity scenario.

2.9 Summary and Simulation Results

Optimal channel equalization is computationally feasible for the binary modulation scheme which is used in the original GSM standard. However, the modulation order has been gradually increased without any reduction of the severe ISI which is present in the narrow-band single-carrier system.

With reduced-state trellis processing, the complexity can be drastically reduced, but good performance is only achieved if the channel is brought to its minimum phase equivalent by all-pass pre-filtering. Reducing the equalizer complexity which grows exponentially with the number of bits per symbol at the cost of a pre-filter with a complexity that does not depend on the modulation order is a very favorable setup for the 16-QAM and 32-QAM modulation schemes of E-EDGE [7].

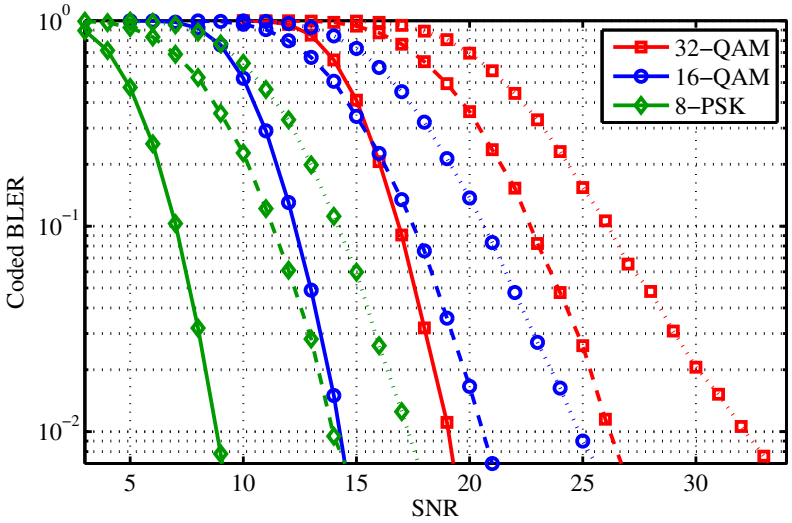


Figure 2.19: BLER performance of E-EDGE modulation schemes over a TU fading channel. The equalization is done with a combination of pre-filter and (SO-)RSSE with $D = 1$. Dotted lines: hard output, dashed lines: soft-output, solid lines: soft-output and RX-diversity.

As shown in Figure 2.19, the BLER performance is increased by approximately 3 dB by providing LLRs to the channel decoder. For reduced-state equalization on minimum-phase channels, the backward recursion does not bring much performance gain (see Figure 2.12) and leaving it out brings a huge complexity reduction.

Employing two RX antennas yields a further significant performance gain both in terms of SNR (array gain) and in terms of steepness of the BLER curve (diversity gain). The BLER curves in Figure 2.19 show that with RX-diversity it is possible to transmit one bit more under equal channel conditions compared to the single-stream soft-output case and even more when compared to the hard-output solution.

Chapter 3

VLSI Circuits for Channel Equalization

In order to come up with small and efficient dedicated VLSI circuits, the general channel equalization algorithms discussed in the previous chapter are tailored to the case of E-EDGE and optimized with regard to suitable hardware architectures. A channel shortening solution based on a novel algorithm in the cepstrum domain is presented in Section 3.1. In Section 3.2 the optimal RSSE configuration for the high modulation orders of E-EDGE is derived and a corresponding ASIC is presented. A VLSI solution of the SO-RSSE algorithm is presented in Section 3.3 and it is compared to an implementation of the SO-MA which is described in Section 3.4.

3.1 Efficient Channel Shortening

As already shown in Section 2.4, RSSE shows a much better performance if the CIR has strong first taps, what can be achieved by linear pre-filtering. Several approaches for the filter coefficients generation of channel shortening pre-filters have been proposed for GSM/EDGE [20, 31–33]. The linear prediction (LP) method has been shown to be less complex than others [20], and is suitable for integration in hardware [21]. In a corresponding ASIC implementation

with filter order $p = 32$, excessive resource sharing minimizes silicon area of the complex filter coefficients computation. However, still one third of the total GSM/EDGE receiver ASIC is occupied by the pre-filter.

RSSE channel equalization of 32-QAM modulated E-EDGE signals requires even higher filter orders to achieve best performance. Since the complexity of pre-filter coefficients computation methods [20, 31, 32] grows with $O(p^2)$ or even faster [33], corresponding implementations will require lots of hardware resources.

3.1.1 Pre-Filter Based on Homomorphic Filter

A pre-filter that transforms the CIR $\mathbf{h}$ to its minimum-phase equivalent $\mathbf{h}^{\min}$ provides exactly the behavior that is needed for a successful RSSE equalization. Computing the filter coefficients of such a pre-filter directly via LP is suitable for EDGE [20, 21]. Other approaches for EDGE have been proposed, that first compute $\mathbf{h}^{\min}$ by using, e.g., the cepstrum of the CIR, before the corresponding pre-filter coefficients are generated with the MMSE criterion [32]. In the following a new method will be explained, that uses the cepstrum directly to compute the pre-filter coefficients via inverse discrete Fourier transform (IDFT). Much of the material in this section has been published in [34].

Homomorphic Filtering

A causal impulse response $\mathbf{h}$ can be written as the convolution of a minimum-phase and an all-pass part

$$\mathbf{h} = \mathbf{h}^{\min} * \mathbf{h}^{\mathrm{ap}} . \tag{3.1}$$

The decomposition into $\mathbf{h}^{\min}$ and $\mathbf{h}^{\mathrm{ap}}$ can be achieved by homomorphic filtering [35] (cf. Figure 3.1). To this end, the real cepstrum $\mathbf{f}$ of the vector $\mathbf{h}$ is computed according to

$$\mathbf{f} = \mathcal{F}^{-1} \{\log |\mathbf{g}|\} \tag{3.2}$$

where $\mathbf{g}$ is the discrete Fourier transform (DFT) of $\mathbf{h}$, $\mathcal{F}^{-1} \{\cdot\}$ is the inverse DFT operation and $|\mathbf{g}|$ refers to the element-wise absolute

value. A causal cepstrum corresponds to minimum-phase in time domain. Hence, multiplying the real cepstrum $\mathbf{f}$ with the sequence $\ell_k^{\min}$ as defined in (3.3) yields the complex cepstrum $\mathbf{f}^{\min}$ of a vector $\mathbf{h}^{\min}$, which has the same amplitude response as $\mathbf{h}$, and is minimum-phase [35]:

$$f_k^{\min} = f_k \cdot \ell_k^{\min}, \qquad \ell_k^{\min} = \begin{cases} 1, & k = 0, \frac{F}{2} \\ 2, & 1 \leq k < \frac{F}{2} \\ 0, & \frac{F}{2} < k < F \end{cases} \tag{3.3}$$

where F is the DFT-order. Finally, transforming $\mathbf{f}^{\min}$ back from the cepstrum to time domain provides $\mathbf{h}^{\min}$. The all-pass part $\mathbf{h}^{\mathrm{ap}}$ can be obtained by re-arranging (3.1) in frequency domain (cf. Figure 3.1).

$$g_k^{\mathrm{ap}} = \frac{g_k}{g_k^{\min}} \tag{3.4}$$

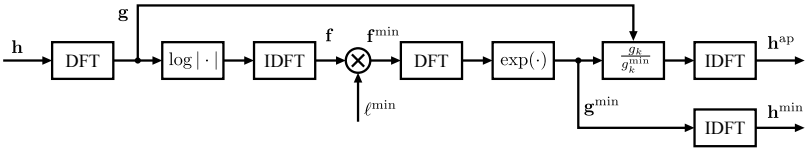


Figure 3.1: Decomposition of $\mathbf{h}$ into minimum-phase and all-pass part by homomorphic filtering.

Efficient Pre-Filter Computation with Homomorphic Filter

The homomorphic filter forms the basis for a new approach to efficiently generate the coefficients of a filter that transforms $\mathbf{h}$ into $\mathbf{h}^{\min}$, as desired for pre-filtering RSSE input signals. By re-arranging (3.4) the pre-filter coefficients $\mathbf{h}^{\mathrm{ap,inv}}$ can be defined according to

$$\mathbf{h}^{\mathrm{ap,inv}} = \mathcal{F}^{-1}\left\{\frac{\mathbf{g}^{\min}}{\mathbf{g}}\right\}, \tag{3.5}$$

and are obtained by simply interchanging numerator and denominator of the division in Figure 3.1.

In order to avoid hardware-expensive complex-valued division and exponential operations, the homomorphic filter algorithm as shown in Figure 3.2 is modified. The division can be implemented efficiently by exploiting the fact, that $\mathbf{g}^{\mathrm{ap}}$ and therefore also the desired filter $\mathbf{g}^{\mathrm{ap,inv}}$ have all-pass characteristic. The magnitude of the frequency-domain filter coefficients of an all-pass is one; hence, the result of the division in (3.5) can be computed by simply subtracting the angle of numerator and denominator.

Further, the (costly) complex exponential operation in the original homomorphic filter can be avoided completely, since the angle of a complex exponential is given by the imaginary part of its input. Thus, the generation of $\mathbf{h}^{\mathrm{ap,inv}}$ is simplified to

$$\mathbf{h}^{\mathrm{ap,inv}} = \mathcal{F}^{-1}\left\{\Im(\mathcal{F}\left\{\mathbf{f}^{\min}\right\}) - \angle\mathbf{g}\right\}\,, \tag{3.6}$$

where $\Im(\cdot)$ refers to taking the imaginary part and $\angle\mathbf{g}$ denotes the angle of the elements of $\mathbf{g}$.

The transformation from polar to Cartesian number representation and vice versa which is required for this approach (cf. Figure 3.2) can be efficiently realized with a CORDIC [36] (cf. Section 3.1.2).

Note that the proposed algorithm requires the DFT/IDFT length to be $F \geq 2p$ in order to achieve best performance. Further note that for frequency selective channels, some frequency bins of $\mathbf{g}$ can become close to zero. In order to avoid numerical instability of the algorithm, the range of the $\log|\cdot|$ function needs to be limited towards negative numbers[1].

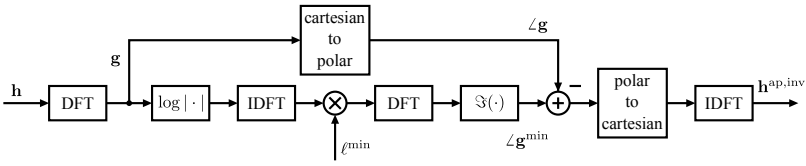


Figure 3.2: Direct *HOM* pre-filter coefficients computation.

[1]In this work, $F = 2p$ and log-clipping at -4 has been used for algorithm simulation, complexity analysis, and hardware implementation.

Algorithm Performance and Complexity Comparison

The proposed *HOM* algorithm as well as LP [20], MMSE-DFE [31], and cepstrum [32] approach have been evaluated by comparing the UBER of 32-QAM modulated signals impaired by the specified HT channel and AWGN. The pre-filter coefficients are computed based on ideal channel estimation and the pre-filtered received symbols are equalized with a 16-state hard-output RSSE.

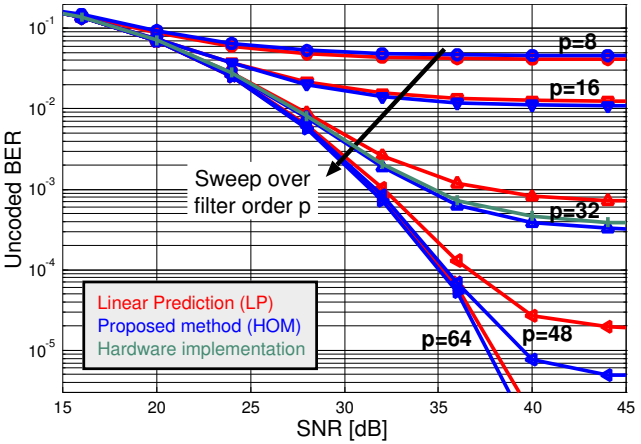


Figure 3.3: Comparison of 16-state RSSE performance with LP and *HOM* pre-filter coefficients computation with varying filter order p.

Simulation results for pre-filtering with varying pre-filter order p by using LP and the proposed *HOM* method are shown in Figure 3.3. Contrary to 8-PSK EDGE signals where RSSE performance saturates with $p \leq 30$ [20, 21], equalizer performance of 32-QAM modulated signals strongly benefits from filter orders of 32 and more. Further, the *HOM* approach performs better than the LP method, i.e., to achieve a certain BER at a certain SNR level, with the LP approach a higher p is required than with *HOM*. Simulations have shown, that the cepstrum method requires even lower p than the *HOM* approach, and MMSE-DFE filter orders are comparable to LP for $p \leq 30$, but performance saturates at 0.05% BER.

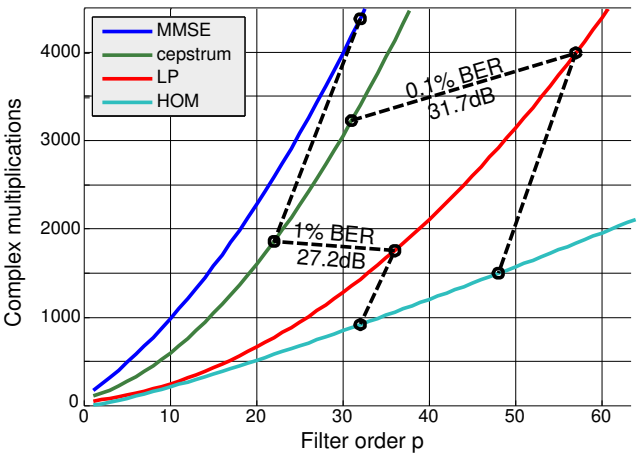


Figure 3.4: Algorithm complexity in number of complex multiplications over the filter order p (operations weighted according to implementation complexity).

In Figure 3.4 the algorithm complexity is compared by showing the number of operations[2] required for the computation of the filter coefficients over p. The p required to achieve a certain BER at a specific SNR is shown for 1% and 0.1% BER (dashed lines) for the different algorithms[3]. As can be seen, the *HOM* algorithm with $p = 32$ is about $2\times$ less complex than comparable LP and cepstrum realizations, and $4.5\times$ less complex than MMSE-DFE. As previously shown, even higher filter orders are required to achieve best RSSE performance with 32-QAM. For such pre-filters the complexity reduction with *HOM* is even more significant, because complexity increases only in $O(p \cdot \log_2 p)$, whereas the number of operations with other methods grows with $O(p^2)$.

[2]The number of operations is given in complex multiplications. Additions have been weighted by assuming data widths of 14 bit as used in the *HOM* design.

[3]Note that with MMSE-DFE approach 0.1% BER cannot be achieved.

3.1.2 Hardware Implementation

In order to prove the suitability of the proposed algorithm for a hardware integration, the *HOM* pre-filter computation with $p = 32$ has been implemented and compared with the LP-based pre-filter design in [21]. The architectures have been designed for (Evolved) EDGE, and optimized for low silicon area, as desired for future low-cost 2G cellular receivers.

Low-Area Sequential Pre-Filter Computation

The sequential characteristic of the *HOM* method allows excessive time-multiplexing of hardware, namely the DFT/IDFT, implemented as FFT/IFFT, and the CORDIC. The architecture depicted in Figure 3.5 mainly comprises a butterfly processing unit (BPU), the CORDIC for the Cartesian-to-polar transformations, the LOG block that computes the (natural) logarithm of an absolute value, as well as the BPU and CORDIC memories.

The BPU memory is realized with 2 two-port RAMs to provide the required memory bandwidth for the fully-sequential radix-2 (I)FFT realization. In each clock cycle, two data items are read from the BPU memory and fed into the BPU where the corresponding FFT or IFFT butterfly is computed. The results are written back to the BPU memory in the same cycle such that one butterfly per cycle can be processed. The address generation has been optimized such that memory access conflicts occur neither in FFT nor in IFFT computation mode. The complex-valued FFT/IFFT twiddle factors are stored in a look-up table (LUT) of $2{\times}16{\times}8$ bit entries. The $\frac{1}{F}$ scaling after $\log_2 F$ IFFT stages is realized with optional shift-by-one operations after each butterfly computation as shown in Figure 3.6. The same shifters are also used to perform the $\ell^{\min}$-scaling in (3.3).

The computation in the LOG block is based on the identity

$$\log(|x + iy|) = \frac{1}{2} \log\left(x^2 + y^2\right). \tag{3.7}$$

Even though the logarithm output is clipped (cf. Section 3.1.1), achieving high precision requires a large LUT that stores log-values. In this solution, the property that the logarithm of a product is the

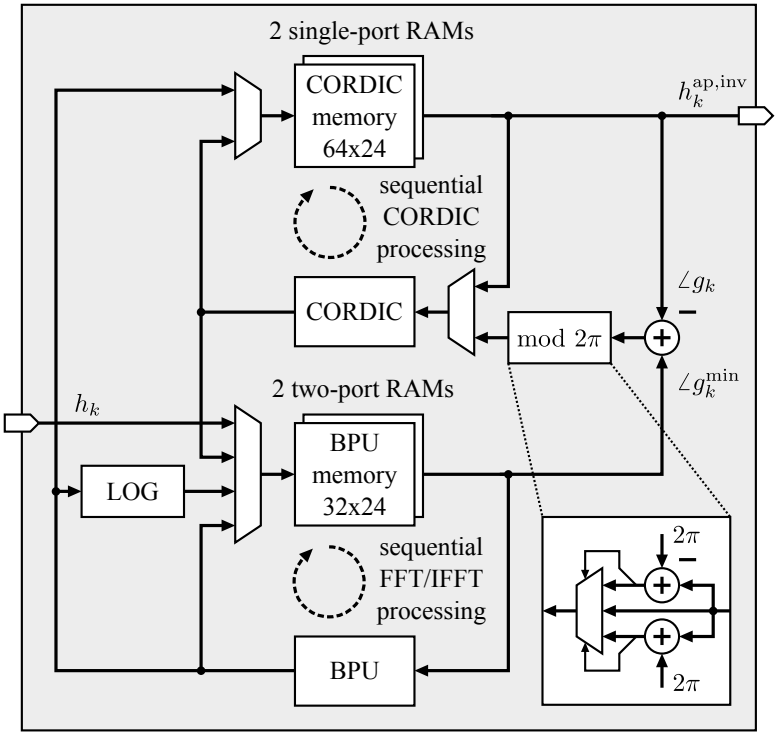


Figure 3.5: Block-diagram of the implemented *HOM* pre-filter coefficients computation architecture.

sum of the logarithm of each factor is exploited, which allows the logarithm inputs to be shifted to a suitable dynamic range.

$$\log a = \log\left(2^{-m}a'\right) = \log a' - m\log 2 \tag{3.8}$$

Only log-values within the interval $[\frac{1}{2}, 1)$ have to be stored in a LUT. The integer m that is based on the number of leading zeroes of the fixed-point representation of a indicates in which interval $[2^{-m-1}, 2^{-m})$ the value a lies (cf. Figure 3.7).

The polar-Cartesian transformations are realized with an iterative CORDIC implementation. The CORDIC in this design can operate

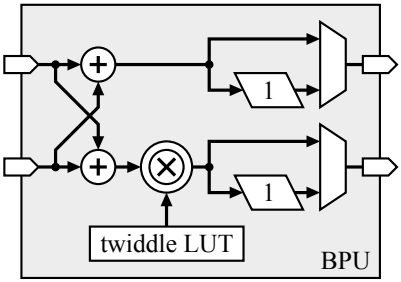


Figure 3.6: Hardware architecture of the butterfly processing unit used for the sequential FFT/IFFT processing.

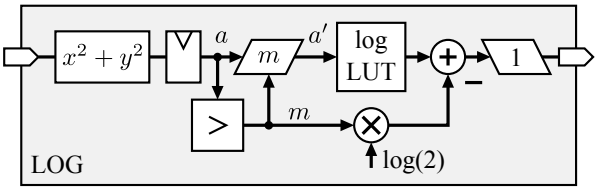


Figure 3.7: Architecture of the LOG block which computes the natural logarithm of an absolute value. The size of the LUT is minimized by shifting the input according to (3.8).

both in rotating and vectoring mode, meaning that the angle of a complex number can be determined or that an angle argument can be used to create real and imaginary part of a complex number. Both operations are performed with two CORDIC iterations per clock cycle in order to achieve a precision of 10 bit after 5 cycles for each computed value (cf., [37] for details on CORDIC architectures).

The sequence of pre-filter coefficients generation is directly given by Figure 3.2. Initially, the (estimated) CIR $\mathbf{h}$ is stored into the BPU memory. The *HOM* processing starts with the first FFT, which has been optimized to reduce the number of butterfly computations[4]. The resulting $\mathbf{g}$ is stored into the CORDIC memory, from where $\angle\mathbf{g}$

[4]The FFT input vector has only 8 non-zero values, since the specified test channels in GSM lead to maximum 8 (symbol-spaced) channel taps.

can be computed concurrently to the generation of $\angle\mathbf{g}^{\min}$. To this end, the FFT-results are directly fed into the LOG block, where the logarithm of the absolute value of one g_k-bin is computed per clock cycle, and stored into the BPU memory. Next, the first IFFT, subsequent scaling with $\ell^{\min}$, and second FFT are performed in iterations between BPU and BPU memory. Finally, the angles of $\mathbf{g}$ which have been concurrently computed during CORDIC iterations are subtracted from the corresponding angles of $\mathbf{g}^{\min}$ (modulo 2π). The results are transformed back to Cartesian coordinates with the CORDIC, and then to time domain with the final IFFT. The resulting pre-filter coefficients are stored into the CORDIC memory and can be used for FIR filtering.

3.1.3 Implementation Results and Comparison

The key figures of the design are given in Table 3.1. The architecture has been implemented in SMIC 130 nm CMOS, and occupies 28.6 k of logic gate equivalents (GE). The computation of the 32 filter taps requires 1100 clock cycles, and the maximum clock frequency is 190 MHz. The pre-filter coefficients computation requires only 1% of the GSM bust period, which relaxes the timing budget for burst-wise RSSE channel equalization in a digital baseband receiver (cf., [21]).

Design (Algorithm)	*HOM*	LP [21]
Gate count[a] [kGE]	28.6	50.0
Memory [kb]	4.6	2.6
Processing cycles N_{cyc}	1100	2900
Max. clock frequency [MHz]	190	178
Hardware efficiency[b] [kGE/MHz]	165.5	814.6

Table 3.1: Synthesis results and comparison in SMIC 130 nm CMOS.

[a] Including memories.
[b] Normalized with N_{cyc}.

In order to allow for a fair comparison, the LP-based pre-filter design in [21] has been re-implemented in SMIC 130 nm. As can be seen in Table 3.1 the *HOM* architecture requires only about half the

GEs, and is almost 3× faster. The *HOM* design improves hardware efficiency by 5×, significantly more than expected from the raw algorithm complexity (cf. Section 3.1.1), which proofs the suitability of the proposed algorithm for hardware implementation, and the efficiency of the architecture.

3.2 VLSI Implementation of a Hard-output RSSE

Even if the computational complexity of channel equalization can be drastically reduced with RSSE, it is still huge when supporting high modulation orders. Hence, dedicated hardware is required to achieve high throughput or to achieve hardware and energy efficient solutions, as desired for mobile devices.

So far, VLSI implementations for RSSE have only been published for the specific application of Ethernet with trellis-coded modulation [38–40], where only costly fully-parallel architectures for very high throughput have been considered. A VLSI implementation of a DDFSE for 2.5G EDGE and for 2.75G Evolved EDGE has been published in [41] and [10], respectively. An RSSE design space exploration for systems with moderate throughput requirements using higher order modulation as well as an efficient VLSI implementation of RSSE for E-EDGE has been published in [42] and is presented in the following sections.

3.2.1 Implementation Aspects of RSSE

In contrast to Viterbi decoding, the branch metric computation (2.5) is the most computationally intensive part of a Viterbi equalizer [7]. Especially the computation of the reference signals $\tilde{r}_k$ requires many expensive complex-valued multiplications. These can be split into two parts according to

$$\tilde{r}_k = \overbrace{\sum_{m=1}^{L-1} s_{k-m}\hat{h}_m}^{e_s} + \overbrace{s_k\hat{h}_0}^{e_b} \,. \tag{3.9}$$

The *partial reference signals* e_s and e_b depend only on the symbols associated with the state $c' \in \mathcal{C}_{k-1}$ and the branch corresponding to the symbol s_k, respectively. For each trellis stage, there are K different e_s and $|\mathcal{O}|$ different e_b, where K is the number of trellis states (cf. (2.7)) and $|\mathcal{O}|$ is the size of the symbol alphabet. Although each of the $K|\mathcal{O}|$ different reference symbols $\tilde{r}_k$ is unique, they can all be computed by adding one of the state-dependent e_s and one of the branch-dependent e_b.

In fully-parallel implementations, all reference signals have to be computed concurrently. When the completion of a trellis stage takes several cycles, however, more efficient computation schemes can be derived by applying pre-computation and storing the different incarnations of e_b or e_s. The stored partial reference signals are simply combined by addition to generate the final $\tilde{r}_k$ according to (3.9). It has been shown in [41] for the case of a DDFSE that the number of multiplications can be drastically reduced by pre-computation of e_s. In the following, the trade-off between computational and storage complexity is discussed for the more general case of RSSE and the possibility of pre-computing e_b is thereby also considered.

Trade-Off between Computational and Storage Complexity

In the following analysis it is assumed that the state metric update (2.6) of a particular state c is completed before the metric update of the next state is started. This requires that the branch metrics of all branches arriving at the current state are computed by combining the corresponding e_s and e_b. Every state metric update requires a combination of different e_s and e_b, but the same e_s and e_b are used several times. Hence, whenever the partial reference signals are reused, they must be either recomputed or loaded from a memory.

When no pre-computation is applied, both partial reference signals e_s and e_b must be calculated for all branches of a trellis stage, resulting in

$$C_{\max} = K|\mathcal{O}|L \tag{3.10}$$

complex-valued multiplications per trellis stage. The number $C_{\max}$ is composed of $K|\mathcal{O}|(L-1)$ multiplications for the computation of all e_s and $K|\mathcal{O}|$ multiplications for the computation of all e_b.

A pre-computation strategy for the e_s should have the objective, that every e_s needs to be calculated only once. This can be achieved by providing a memory capacity of J_1 words, since every state c is connected to J_1 predecessor states. The parameter J_1 corresponds to the number of subsets into which the symbol alphabet is divided for the symbol s_{k-1} (refer to Section 2.3). While the pre-computed e_s are recalled from memory, the state metric updates of all states that connect to the same predecessor states can be completed. After that, these e_s are no longer required and can be overwritten with new e_s. This reduces the number of multiplications for the computation of e_s from $K|\mathcal{O}|(L-1)$ to $K(L-1)$. Since all required e_s are available from memory, the order of the branches contributing to the state metric update can be chosen freely. Hence, the e_b of a given symbol can be used for all branches that correspond to that symbol but connect to different predecessor states, before a new e_b is calculated. This reduces the number of multiplications for e_b from $K|\mathcal{O}|$ to $K|\mathcal{P}_{k-1}^i|$, where $|\mathcal{P}_{k-1}^i| = |\mathcal{O}|/J_1$ is the number of parallel branches between two states. The total number of multiplications for the case of J_1 pre-computed e_s is thus given by

$$C_{\text{state}} = K(L-1) + K|\mathcal{P}_{k-1}^i|\,. \tag{3.11}$$

Conversely, the e_b can be pre-computed such that all e_b need to be computed only once. This reduces the number of multiplications for e_b from $K|\mathcal{O}|$ to $|\mathcal{O}|$. To achieve this, the e_b of all parallel branches must be pre-computed and stored. The availability of the e_b for all parallel branches has the advantage that the same e_s can be used for all parallel branches before a new e_s is computed, reducing the number of multiplications for e_s from $K|\mathcal{O}|(L-1)$ to $KJ_1(L-1)$. The total number of multiplications per stage can be reduced to

$$C_{\text{branch}} = KJ_1(L-1) + |\mathcal{O}| \tag{3.12}$$

by providing a storage capacity for $|\mathcal{P}_{k-1}^i|$ words.

When the two pre-computation strategies are combined, all e_s and e_b must be calculated only once. To this end, the e_b of all possible $|\mathcal{O}|$ symbols must be pre-computed and stored in memory. Furthermore, the e_s of J_1 states must be available in memory. By providing a

storage capacity of $J_1 + |\mathcal{O}|$ words, the number of multiplications can be reduced to

$$C_{\min} = K(L-1) + |\mathcal{O}|\,. \tag{3.13}$$

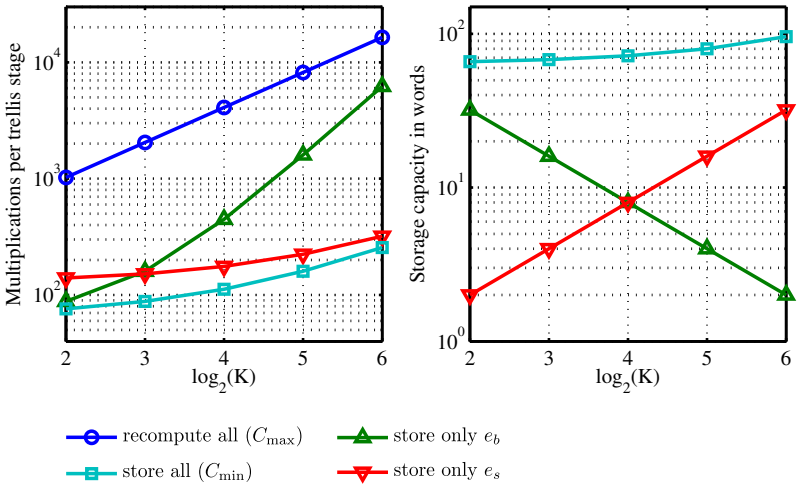


Figure 3.8: Trade-off between computational and storage complexity for different number of trellis states.

The trade-off between storage and computational complexity for the four strategies discussed above is visualized in Figure 3.8 for the case of 64-QAM ($|\mathcal{O}| = 64$) and a channel length of $L = 4$. The number of RSSE states K is varied between 2 and 64 and the parameter J_1 is always chosen as $K/2$ in this case. While the selection of K and J_m is subject to algorithmic evaluation (see Section 3.2.2), the most suitable hardware architecture for each configuration can be found by considering the options in Figure 3.8. The figure illustrates that with the presented pre-computation approach the number of multiplications per trellis stage of a 64-state RSSE can be greatly reduced from $C_{\max} \approx 16000$ to $C_{\min} = 256$ at the cost of a storage capacity of 96 words. Since storage and computational complexity are two independent optimization criteria, there are many Pareto optimal

solutions. However, a visualization of the trade-off helps to find the most suitable solution for a given design goal.

3.2.2 Configuration of RSSE for GSM Channels and High Modulation Orders

The combination of minimum-phase pre-filter and RSSE allows the number of equalizer trellis states to be significantly reduced with a performance still close to MLSE. For 8-PSK transmitted over the GSM channel models, a trellis with only 4 states seems to be sufficient, as Gerstacker has shown in [5]. For higher modulation orders it has been shown in [7] that DDFSE with $D = 1$ shows good performance. However, the performance can be slightly improved with even less trellis states thanks to the more general subset partitioning of RSSE. Figure 3.9 compares the 32-QAM bit-error rate of RSSE with two different subset configurations given by $(J_1/J_2/J_3)$ and of DDFSE with $D = 1$.

For the TU channel profile, both 16-state RSSE configurations outperform the 32-state DDFSE by a little. Both benefit from trellis processing for the second last symbol in the channel memory ($J_2 \neq 1$) and the Euclidean distance between the symbols in the first subsets $\mathcal{P}_{k-1}^{i}$ seems to be large enough. Due to the late reflections of the HT channel model, the RSSE configuration with trellis processing for s_{k-3} ($J_3 \neq 1$) shows a better performance than the (8/2/1) version with equally many states. Compared to DDFSE, the (4/2/2) RSSE configuration improves the performance by almost 1 dB in SNR. Unfortunately, numerical simulations for MLSE are not feasible for 32-QAM and $L = 8$ (number of states $> 10^{10}$), but a 4096-state RSSE can be expected to perform almost equally and hence the corresponding curves are labeled with quasi-MLSE. It has already been shown in [5] for EDGE with 8-PSK, that RSSE requires less states than DDFSE to achieve the same BER performance. Figure 3.9 now demonstrates that the same is true for 32-QAM and that a $|\mathcal{O}|/2 = 16$-state RSSE approaches MLSE performance quite closely.

Depending on the channel model, a different RSSE configuration performs best, but the (4/2/2) configuration is among the best for both TU and HT channel models and outperforms the DDFSE at half the computational complexity. On these grounds, the (4/2/2)

configuration has been chosen for the VLSI implementation described in the following.

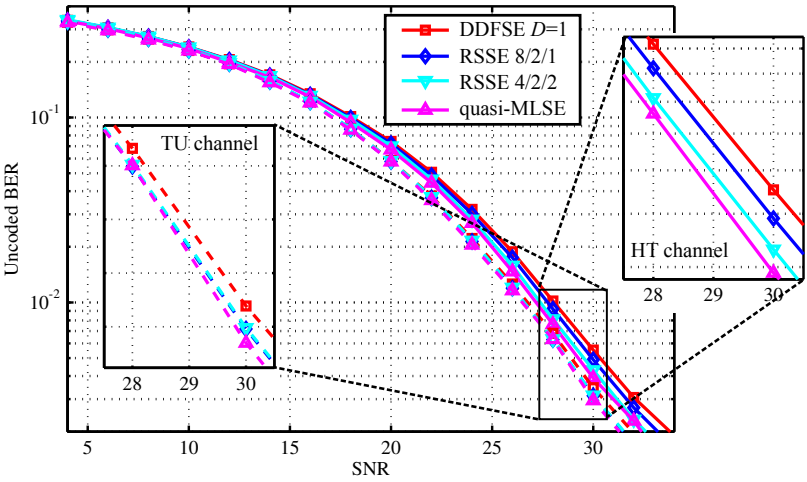


Figure 3.9: 32-QAM bit-error rate comparison between two RSSE configurations with $J_1/J_2/J_3$ and a DDFSE with $D = 1$ (32/1/1). The curves of a 4096-state RSSE are shown as quasi-MLSE reference. Simulation results for both HT and TU channel model with minimum-phase pre-filtering are shown.

3.2.3 VLSI Architecture

Even if the VLSI implementation has been optimized for the high modulation orders of E-EDGE, it also supports 8-PSK and GMSK which is used for GSM/EDGE. One GSM burst is processed in two separate trellis recursions (cf. Section 2.1). For 16-QAM and 32-QAM, the RSSE is configured (4/2/2) with 16 trellis states resulting in a close-to-ML performance as shown in Section 3.2.2. Simulations for 8-PSK and GMSK have shown that a (2/2/2) configuration with 8 states is sufficient for these modulation schemes when employing the channel shortening filter of Section 3.1.1 in front of the RSSE.

As storage requirements for a 16-state trellis are fairly low, the combined pre-computation approach was chosen and the e_s of all 16 predecessor states and the e_b of all 32 possible symbols are pre-computed (cf. Sec. 3.2.1). Thus, the number of multiplications per trellis stage has been greatly reduced from 4096 (no pre-computation) to 144 at the cost of a storage capacity of 48 pre-computed words.

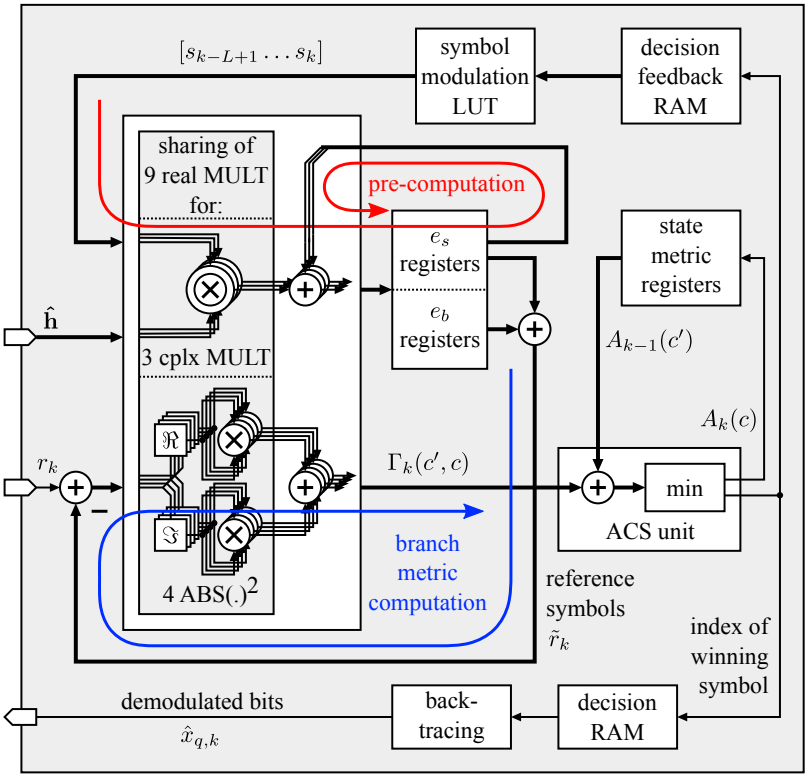


Figure 3.10: Block diagram of RSSE hardware implementation.

A block diagram of the corresponding RSSE architecture is depicted in Figure 3.10. Inputs to the block are the estimated CIR $\hat{\mathbf{h}}$ and the received samples at symbol rate r_k. In a first step, all e_s and e_b are pre-computed. Then, corresponding e_s and e_b are

summed up to reference signals $\tilde{r}_k$ which are subtracted from the received signal r_k. The branch metric computation (2.5) is completed by calculating the square of the absolute value. For all calculated branch metrics $\Gamma_k(c', c)$, the ACS unit adds the state metric $A_{k-1}(c')$ of the corresponding predecessor state to get the path metric and determines the index of the winning symbol s_k as well as the new state metric $A_k(c)$ which is then stored in the state metric memory. The index of the winning symbol is stored in both the decision RAM for the final back-tracing process and in the decision feedback RAM, where decisions are stored for each state. Based on these decisions and on the current state index, the $L-1$ modulated symbols for the e_s pre-computation of the next trellis stage are derived by accessing a look-up table (LUT).

In order to optimize the area utilization of the design, the multipliers for the pre-computation are reused for the calculation of the squared Euclidean distances. To achieve the required throughput at our target clock frequency of 52 MHz, 9 real-valued multipliers are employed which can be either used for 3 complex-valued multiplications[5] or for 4 squaring operations.

Several storage elements in our architecture require parallel access to four words in the same clock cycle. Thus, they are realized most efficiently with flip-flop arrays. The large decision memory as well as the two memories used for the storage of decision feedback information are realized with single-port RAMs, in order to save silicon area.

3.2.4 Implementation Results

The RSSE design has been manufactured in a 0.18 μm CMOS technology and a chip photograph is depicted in Fig. 3.11. It has been measured on a digital tester to operate at a clock frequency of 124 MHz. The clock frequency required to achieve the target throughput is more than two times lower. This cycle-time headroom provides the possibility to scale down the supply voltage for power savings. Table 3.2 compares our design with the only previously published 2.75G detector [10] as well as with soft-output implementations which will be

[5] With Gauss' complex multiplication algorithm, the number of real-valued multiplications can be reduced to 3.

introduced in the following sections. The performance in terms of bit-error rate of the implemented algorithm is slightly better when compared to that of DDFSE, as can be seen in Figure 3.9. Furthermore it should be noted that the design in [10] approximates the squared Euclidean distance in (2.5) with the ℓ^1-norm[6] what deteriorates the BER performance by another 0.5 to 1 dB [7]. When employing a hardware efficiency measure defined by the area in gate equivalents (GE) divided by the maximal throughput, our solution shows an efficiency gain of 1.6 when compared to the other hard-output E-EDGE solution based on DDFSE.

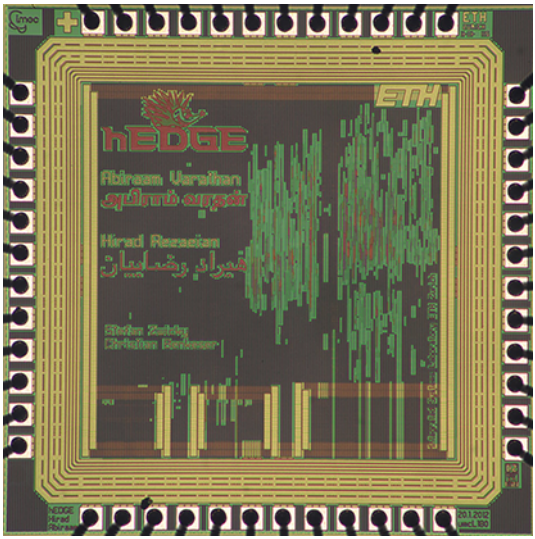


Figure 3.11: Chip photograph of the 16-state RSSE.

[6]The distance under the ℓ^1-norm is defined by $|\Re(d)| + |\Im(d)|$.

3.3 VLSI Implementation of a Soft-output RSSE

As shown in Figure 2.19, providing soft-outputs to the channel decoder and exploiting RX-diversity improves the block-error rate performance a lot. However, VLSI implementations of a trellis-based soft-output equalizer for moderate throughput, high modulation orders and severe ISI have not been reported so far. In order to render a hardware implementation feasible, the algorithmic approximations of the optimal MAP algorithm which are described in Section 2.6 have been considered. The implemented algorithm can be characterized as follows:

- Reduced number of trellis states by exhaustive search over the two most recent symbols ($D = 1$) and per-state decision-feedback for the other $L - D - 1 = 6$ symbols.
- Max-log approximation is employed.
- Ignoring backward metrics B_k in LLR computation (2.20).
- LLR computation with a delay of D symbols as in (2.21).
- Combination of two received symbols in the branch metric computation as in (2.24).
- Support of all modulation orders of E-EDGE: GMSK, 8-PSK, 16-QAM and 32-QAM.

3.3.1 Degree of Parallelization

As for the hard-output RSSE, the BM computation (2.24) is clearly the computationally most demanding task. In order to determine the required amount of parallelization to process the data fast enough the condition

$$\frac{|\mathcal{O}|^2 2N}{\#\mathrm{BMU} f_{\mathrm{clock}}} < \alpha T_{\mathrm{burst}} \tag{3.14}$$

is examined, where $|\mathcal{O}|^2 2N \approx 118'000$ is the number of BM to be computed per 32-QAM burst, f_{clock} is the frequency of the processing clock, #BMU is the number of parallel BM computing units,

$T_{\text{burst}} = 577\,\mu\text{s}$ is the duration of a GSM burst, and $\alpha < 1$ defines the fraction of T_{burst} that can be used by the equalizer. In the 130 nm target CMOS technology, a processing frequency of around 100 MHz is reasonable for designs with complex-valued multiplications. This results in $\alpha = 0.69, 0.51, 0.41$ for #BMU = 3, 4, 5. Since it must be expected that the equalizer requires some overhead cycles where no BMs are computed and to leave other units enough processing time, an architecture capable of computing four BM in parallel has been chosen.

3.3.2 VLSI Architecture

The VLSI architecture of the soft-output RSSE is shown in Figure 3.12. Each of the two CONV units contains a complex-valued multiplier which is used to calculate the partial reference symbols e_s and e_b (cf. (3.9)). All e_b are pre-computed and stored in the Reference Symbol Memory. Since the computation schedule is such that all branches connected to a particular previous state are computed in sequence, only storage for one e_s per stream is required. The e_b and e_s are then summed up to complete reference symbols and the difference $\mathbf{d}_k$ to the received signal is computed in the DIFF block. The branch metrics delivered by the four branch metric units (BMU) are added to the previous state metric A_{k-1} to obtain new temporary state metrics A_k which are compared to results of previous computations in the Current State-Metric (SM) block. Each of the four Current SM blocks contains the temporary minimal SM of eight different states and at the end of each trellis stage all 32 final state metrics are available along with the index of the corresponding predecessor state. At the end of a trellis stage, all state metrics are written to the Previous SM memory and the list of decision feedback symbols of each state is updated according to the index of the winning predecessor state.

Especially in the low SNR regime, the accumulated state metrics grow quickly and a binary representation would require many bits. In order to keep the state metrics small, the best state metric per trellis stage is determined in the Stage MIN block and subtracted from all SM values when they are written to the Previous SM memory. This common offset does not influence the final result since the LLR values

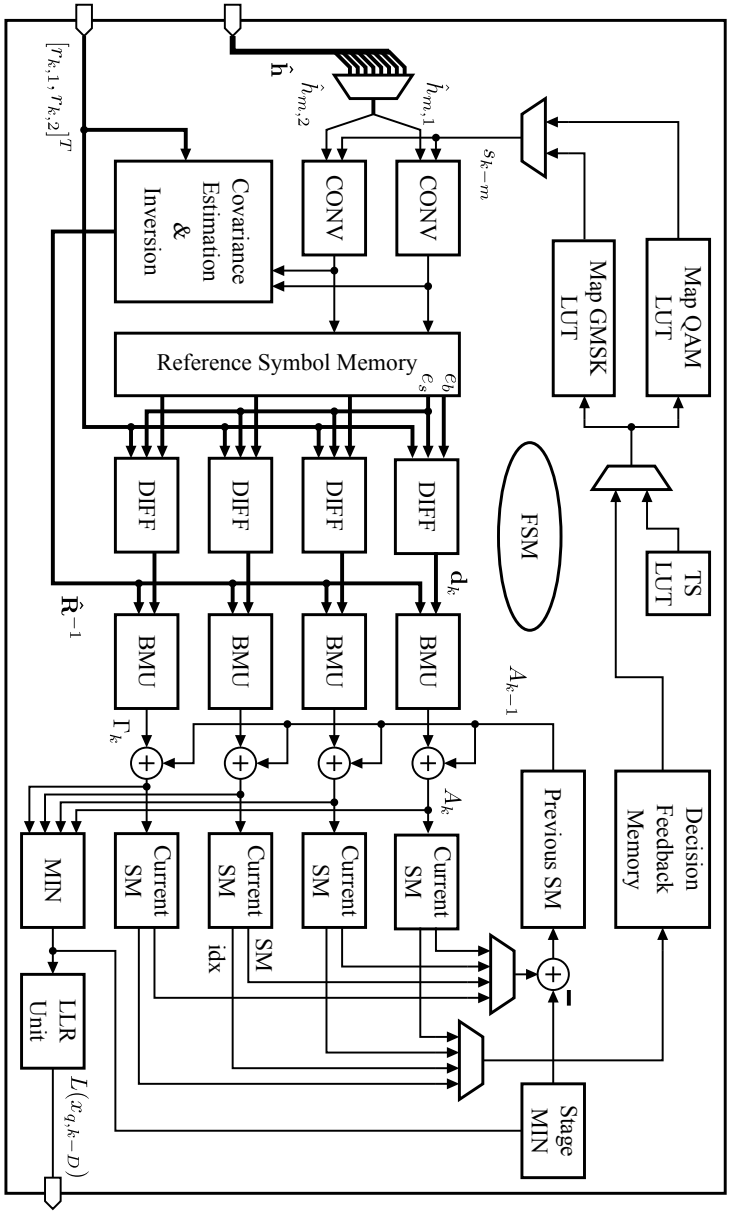


Figure 3.12: Hardware architecture of soft-output RSSE.

consist of a difference between state metrics of the same stage. However, the word width of the adders and the SM registers is significantly reduced, resulting in a much smaller silicon area.

The Decision Feedback Memory block contains two single-port RAMs, one for trellis stage $k-1$ and one for stage k. The sequence of decision feedback symbols is mapped to I/Q symbols with the aid of a look-up table (LUT). While the mapping is done symbol-wise in the case of QAM and PSK modulation schemes, the special GMSK modulation requires the joint-mapping of the whole sequence of $L = 8$ symbols.

In the LLR Unit, the reliability information is gathered by building the difference (2.21). The temporarily minimal value of the sets $\mathcal{B}^{+1}_{x_{q,k-D}}$ and $\mathcal{B}^{-1}_{x_{q,k-D}}$ for all q is constantly updated during the processing of a stage. The binary representation of the symbol corresponding to the predecessor state determines whether a value contributes to the $+1$ or the -1 group.

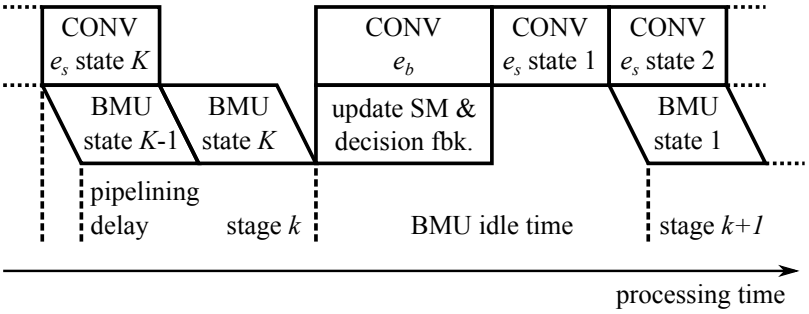


Figure 3.13: Computation schedule of the soft-output RSSE.

Figure 3.13 illustrates the computation schedule of the architecture. The branch metric computation of all 32 branches connected to the same state takes 8 clock cycles. At the same time, the partial reference symbol e_s of the next state is computed in the CONV blocks. Clearly, the e_s computation of the first state of trellis stage $k+1$ cannot start until the last BM are available (with an additional delay of 2 clock cycles due to pipelining in the BMU), and the decision feedback symbols of stage k have been updated. Hence, the BMUs are idle

for some cycles at the trellis stage boundary as shown in Figure 3.13. However, the idle time is only about 16 % of the total processing time and the circuit area saved due to multiplier sharing in the CONV unit, the adoption of area-efficient but low-bandwidth RAM blocks, and the relaxed timing in the BMU justify these overhead cycles.

Branch Metric Unit

The block with the highest computational complexity is the BMU which computes (2.24). Its architecture is depicted in Figure 3.14. The inverse of the estimated covariance matrix $\hat{\mathbf{R}}^{-1} = \hat{\mathbf{U}}$ is provided by the two real values $\bar{u}_{11}$ and $\bar{u}_{22}$, the complex value $\bar{u}_{12}$ and a shift value ℓ which is given by the pseudo floating-point representation

$$\hat{\mathbf{U}} = 2^{\ell} \begin{bmatrix} \bar{u}_{11} & \bar{u}_{12} \\ \bar{u}_{12}^{*} & \bar{u}_{22} \end{bmatrix} \tag{3.15}$$

and is chosen such that $\max\{\bar{u}_{11}, \bar{u}_{22}\}$ is between 0.5 and 1. This limits the dynamic range of the $\bar{u}_{11/22}$ values what is important, since the diagonal of $\mathbf{U}$ is proportional to the SNR and ranges over several decades.

The BMU contains several arithmetic operations in series what makes the path delays through this block clearly longer than any other paths in the design. Hence, two pipeline stages have been inserted as shown by the dashed lines in Figure 3.14 with the objective to balance the path delays in the design and thus to improve the AT-product even if four additional pipelining registers are inserted [43].

Covariance Estimation and Inversion

As explained in Section 2.8, the spatial covariance matrix of the additive noise is estimated by comparing the known reference symbols of the training sequence with the received samples. Since the computation of these reference symbols involves the mapping of the training sequence (TS) to I/Q symbols and the convolution with the estimated CIR, it makes a lot of sense to reuse the mapping LUTs and the CONV blocks of the equalizer architecture for this as shown in Figure 3.12. Another argument for the resource sharing is the fact that the equalizer can anyway not start until $\hat{\mathbf{R}}^{-1}$ is available.

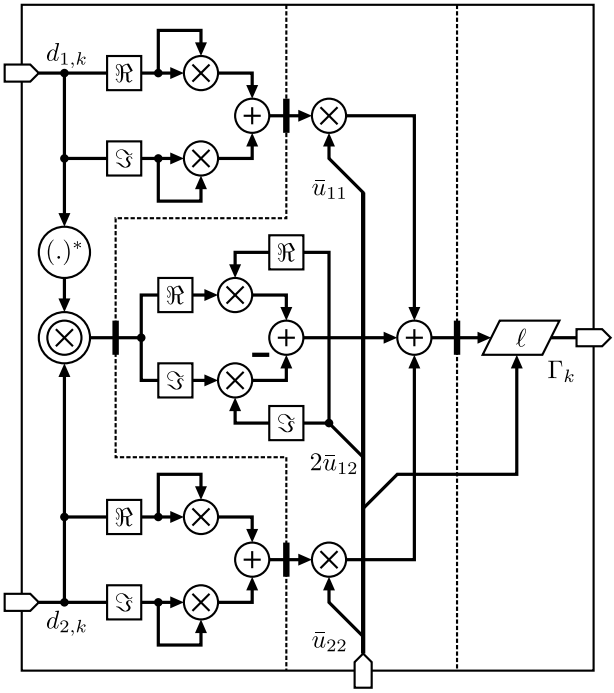


Figure 3.14: Architecture of BMU. Complex-valued multiplication is drawn with a double circle. All other operations are real-valued.

Furthermore, the covariance estimation is only active for a short time and the dedicated hardware for a task that is mostly idle would be a waste of resources.

Based on the reference signals, the estimate $\hat{\mathbf{R}}$ is then computed according to (2.29). The difference terms $d_{1,t}$ and $d_{2,t}$ are multiplied with each other with one single time-shared multiplier. To limit the width of the multiplier inputs, the difference terms are arithmetically shifted by ℓ_1 positions according to $\mathbf{d}_t = 2^{\ell_1}\bar{\mathbf{d}}_t$ such that the dynamic range of $\bar{\mathbf{d}}_t$ is greatly reduced and less bits are required for the binary representation.

The inversion of the 2x2 matrix $\hat{\mathbf{R}}$ is achieved by rearrangement and negation of the elements and by the division with the determinant. To avoid the costly division operation, it is approximated by multiplication by $2^{-\ell_2}$ what corresponds to a right-shift by ℓ_2 bit positions. Furthermore, the division by the number of accumulated values $T = 19$ is approximated by a shift by 4 positions. Since the BMU expects the diagonal elements of $\hat{\mathbf{R}}^{-1}$ to be between 0.5 and 1, another shift value ℓ_3 is introduced, completing the total shift to $\ell = \ell_1 + \ell_2 + \ell_3 + 4$.

Implementation Results

The design has been synthesized to a 130 nm CMOS technology with different constraints for the processing clock and the resulting AT-diagram is shown in Figure 3.15. The design with minimal AT-product would run with a clock frequency of up to 166 MHz, but the architecture is laid out to achieve the required throughput also with 100 MHz. Hence, the implementation at 125 MHz is chosen for the comparison with other equalizer designs (cf. Section 3.5), leaving still some margin to scale the supply voltage down if the design runs at the required 100 MHz. As expected, the BMUs make up over 40 % of the equalizer area of 0.43 mm^2, followed by the CONV blocks and the reference symbol memory which occupy together almost 25 % of the area.

3.4 VLSI Implementation of a Soft-output M-Algorithm

In Section 2.6, it has been shown that the soft-output M-algorithm (SO-MA) with a constraint selection of the M best states has an equivalent BLER performance than the SO-RSSE with only half the number of trellis states and under certain channel conditions, the BLER performance is even slightly better as shown in Figure 2.13. As the number of trellis states is directly related to the computational complexity, a VLSI implementation of the SO-MA can be expected to achieve a good hardware efficiency. However, in contrast to the regular structure of the SO-RSSE algorithm, the selection of the best states

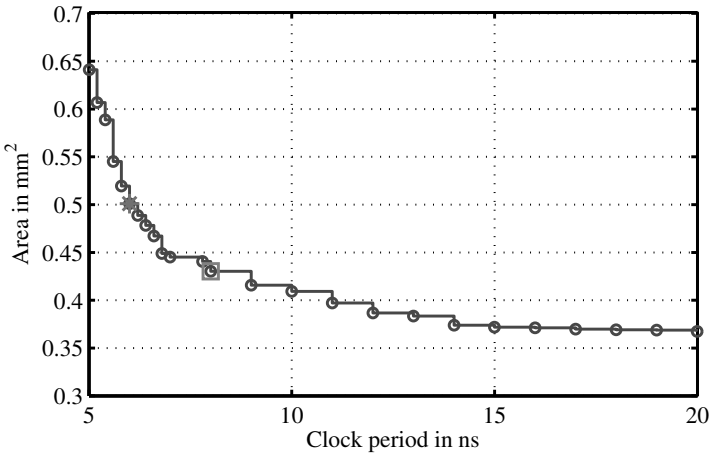


Figure 3.15: AT-diagram of the soft-output RSSE. The optimal implementation in terms of area/delay tradeoff is marked with a star and the version used for the comparison with other designs is marked with a square.

by adhering to the bit-wise constraint is quite complex and it is hence not clear if the SO-RSSE of Section 3.3 can be outperformed in terms of hardware efficiency. In order to find an answer to that question, a dedicated hardware solution for the SO-MA has been developed [44].

The VLSI design supports all modulation schemes of E-EDGE and keeps a list of the $M = 16$ best states at each trellis stage. Up to QD of the M places are occupied by possibly non-optimal states to ensure a good counter-hypothesis for all bit positions. The decision delay has been fixed to $D = 1$.

The overall architecture and processing schedule of the SO-MA implementation is very similar to that of the SO-RSSE. A block diagram of the architecture is shown in Figure 3.16. The pre-computation of the partial reference symbols and the branch metric computation is almost identical to that of the SO-RSSE (cf. Figure 3.12) with the difference that due to the halved number of trellis states, only two instead of four branch metrics need to be computed per cycle.

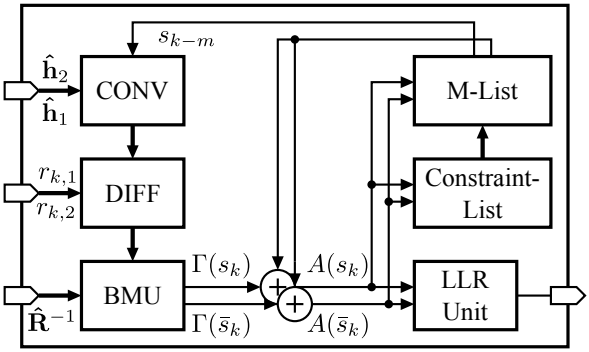


Figure 3.16: Hardware architecture of the SO-MA.

Also the LLR Unit is similar in the sense that for each bit q, the best metric of all states with $x_{q,k-D} = \pm 1$ is determined, separately for $+1$ and for -1. The difference to the SO-RSSE is that the number of metrics to choose from can be different for the $+1$ and the -1 group. Thanks to the constraint selection of survivor states in the previous trellis stage it is however guaranteed that at least $|\mathcal{O}|$ metrics with a $+1$ or a -1 at bit $x_{q,k-1}$ are available.

The processing order of the branches is such that all branches with the same predecessor state are processed consecutively. As a consequence, every state-dependent partial reference symbol e_s needs to be computed only once. Furthermore, the two branches being processed in the same cycle correspond to two symbols with complementary binary representation. These two symbols are denoted by s_k and $\bar{s}_k$. In the Constraint-List block, the currently best states with a $+1$ and a -1 on all bit positions of the most recent symbol are determined. To this end, the state metric $A(s_k)$ is compared to the Q candidates for the bits being equal to s_k and $A(\bar{s}_k)$ is compared to the other Q candidates. This BM processing schedule facilitates the implementation of the Constraint-List block as each of the two state metrics is compared to an equally large group of candidates and these two groups do not intersect.

In the M-List block, the two new state metrics $A(s_k)$ and $A(\bar{s}_k)$ are inserted into a sorted list of the $M = 16$ currently best states.

First, it is checked if the current state is already in the list. In this case, only the state with the better metric is kept in the list. If the state is not yet in the list, it is compared to the last entry and the winner is kept. Subsequently, the list, which now can be in any order, is sorted in ascending order again. A sorting-network with a topology as proposed by Batcher in [45,46] is employed. The network is composed of 63 Metric Compare Units with a circuit as shown in Figure 3.17. The smaller state metric is forwarded to the $A(s_k^{\min})$ port and the larger metric to the $A(s_k^{\max})$ port. Both metrics come along with corresponding lists of symbols defining the states c^1 and c^2 which are multiplexed in the same way as the metric values.

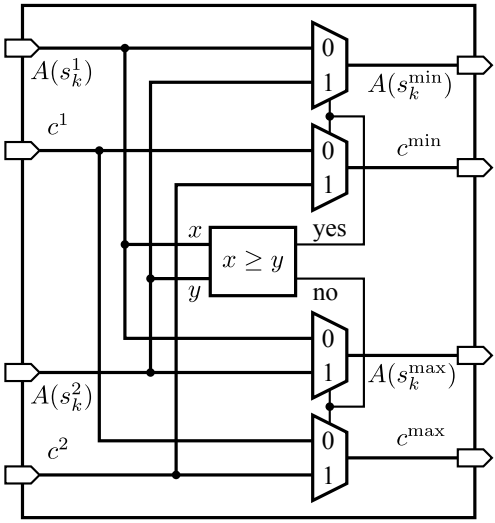


Figure 3.17: Metric Compare Unit which is used to sort two state metrics and the corresponding state description in ascending order.

At the end of each trellis stage, the states in the Constraint-List block need to be merged into the M-List. To this end, the states with the largest metric in the M-List are replaced by states that are required to fulfill the constraint. However, a state is only replaced if it is not already in the list. This replacement is done sequentially, one state after the other, since a fully parallel merging of the lists would

require too many comparators. The required overhead cycles do not affect the throughput significantly, since the merging needs to be done only once per trellis stage.

To conclude the description of the SO-MA VLSI implementation, it can be stated that the effort to maintain a list of the 16 best states such that the bit-wise constraint is fulfilled and without having any duplicate states in the list is large. This is highlighted by the fact that in the SO-RSSE design, the convolution and branch metric computation occupy over 70 % of the area, while the percentage of these blocks is reduced to 45 % in the SO-MA design, but it is the Constraint-List and the M-List block that use over 40 %. A summary of the key figures is shown in Table 3.2.

3.5 Discussion and Comparison

A total of three VLSI solutions for E-EDGE equalization have been presented in this chapter. They all rely on minimum-phase channels which can be obtained by employing a pre-filter as presented in Section 3.1. Implementation results of the equalizers are summarized and compared to a previously published solution [10] in Table 3.2. Two of the designs provide only hard outputs and do not support RX-diversity. It has been shown that the DDFSE can be outperformed in terms of BER with only half the number of trellis states by employing the more general subset partitioning of RSSE. At the same time, the resulting VLSI solution is 1.6 times more efficient. Hence, the RSSE algorithm is clearly suited for very low-area dedicated hardware.

Providing reliability information to the channel decoder improves the BLER performance significantly as shown in Figure 2.19. The performance gain comes however at an increased complexity in both equalizer and channel decoder. A further performance improvement is achieved by employing twofold RX-diversity, but combining the two streams during the branch metric computation also increases the equalizer complexity. By using the Max-Log approximation and by neglecting the B_k values of the backward iteration, the algorithmic complexity of the optimal MAP algorithm can be drastically reduced with only minor performance degradation as shown in Figure 2.12. The computational complexity is still very high, but the SO-RSSE

VLSI solution presented in Section 3.3 is capable of handling this complexity very efficiently by keeping the word width small. The circuit size is even slightly smaller than for the HO-RSSE solution, but the smaller feature size of the used CMOS technology and the lower throughput lead to a hardware efficiency which is 2.5 times worse. However, with the high BLER performance improvement in mind, this increased complexity is acceptable.

The SO-MA implementation however cannot match the SO-RSSE solution, even if from an algorithmic point of view, due to the smaller number of required trellis states, an efficient solution could be expected. One reason is certainly the exceptionally efficient implementation of the SO-RSSE design. Another reason is, however, the high number of sorting operations required to maintain the list of the $M = 16$ best states.

The SO-RSSE VLSI implementation demonstrates that it is possible to realize a high-performing channel equalization solution supporting RX-diversity in dedicated hardware. The required silicon area measured in gate equivalents (GE) is similar to the previously published DDFSE solution and only 50 % larger than for the HO-RSSE. Therefore, the SO-RSSE design has been adopted by the E-EDGE transceiver ASIC presented in the next chapter.

Table 3.2: Comparison of Evolved EDGE Viterbi equalizer implementations. The numbers are based on the designs after circuit synthesis.

Design	DDFSE [10]	HO-RSSE	SO-RSSE	SO-MA
Soft outputs	✗	✗	✓	✓
Number of trellis states	32	16	32	16
RX-diversity	✗	✗	✓	✓
Vector norm	ℓ^1	ℓ^2	ℓ^2	ℓ^2
Technology	ST 130 nm	UMC 180 nm	SMIC 130 nm	SMIC 130 nm
Area [mm^2]	0.573	0.575	0.430	0.730
Gate count[a] [kGE]	95	61	94	160
Memory [kb]	11	6.2	1.9	-
f_{max} [MHz]	151	124 (172[b])	125	100
f_{target} [MHz]	40	52	104	100
Maximal throughput [Mbit/s]	4.4	3.3 (4.6[b])	2.1	1.8
HW efficiency [kGE/(Mbit/s)]	21.5	18.6 (13.3[b])	33.8	87.7

[a] Including memories.
[b] Scaling to 130 nm [47]: $t_{pd} \sim 1/l$

Chapter 4

Evolved EDGE Transceiver ASIC

The emerging M2M communication market calls for complete physical layer (PHY) solutions suitable for deployment in plenty of devices. By developing a solution with a form factor that fits even into the smallest embedded devices, and a power consumption as low as required by mobile devices, the possible range of applications and thus the market potential can be extended. Furthermore, the solution should be inexpensive to target low-cost sensor solutions. All of these targets can be achieved by developing an ASIC for two reasons. First, the large market size justifies the high investment costs and asymptotically the price depends only on the silicon die size which is minimal for an ASIC when compared to other signal processing platforms. Second, the signal processing in the PHY is challenging and a GMAC/s digital signal processor (DSP) would be required in a processor-based solution. An ASIC is expected to be both smaller and more power-efficient.

In this chapter, an E-EDGE transceiver ASIC is presented in the context of a complete PHY system, including radio frequency (RF) subsystem and processor. The ASIC is realized by bringing efficient VLSI solutions of the most challenging digital baseband blocks together. Furthermore, the BLER performance of the PHY system is

measured and compared to simulation results, standard requirements, and prior art. The ASIC has already been presented at ESSCIRC 2014 [48].

4.1 PHY System

The signal processing tasks of the E-EDGE PHY system are depicted in Figure 4.1. The signal coming from one of the two antennas is first passed through a direct-conversion receiver consisting of a bandpass (BP) filter, a low-noise amplifier (LNA), an I/Q-demodulator, and an anti-aliasing lowpass (LP) filter before it is digitized by an analog-to-digital converter (ADC) at a rate of 26 MHz what corresponds to 96 times the symbol rate. The sampling rate is further reduced to fourfold oversampling by a CIC decimation filter and later, after the adjustable adjacent channel (AC) suppression filter, to symbol rate. One of the drawbacks of a direct-conversion receiver is the strong DC-offset that can be present in the received signal even if the analog circuits contain a sophisticated DC compensation. The remaining DC-offset needs therefore to be corrected in the digital domain. Furthermore, synchronization in time and frequency is achieved and the CIR is estimated. Based on the estimated CIRs of both received streams, the coefficients of the space-time filter (STF) are computed according to [29] and the co-channel interference (CCI) is suppressed by linear filtering. After that, the CCI level is lower and an improved CIR estimate is obtained. The channel is then brought to its minimum-phase equivalent by a pre-filter as described in Section 3.1.1, before reliability information about the transmitted bits is derived in the soft-output RSSE equalizer. Finally, the originally transmitted block of data is reconstructed by forward error correction in the channel decoder.

The digital domain signal processing in the transmit path is much simpler and consists of encoding, interleaving and puncturing before the bits are mapped to constellation points. Both I- and Q-component are then converted to analog domain and modulated in an I/Q modulator before a power amplifier is employed and the signal is transmitted over one of the two antennas.

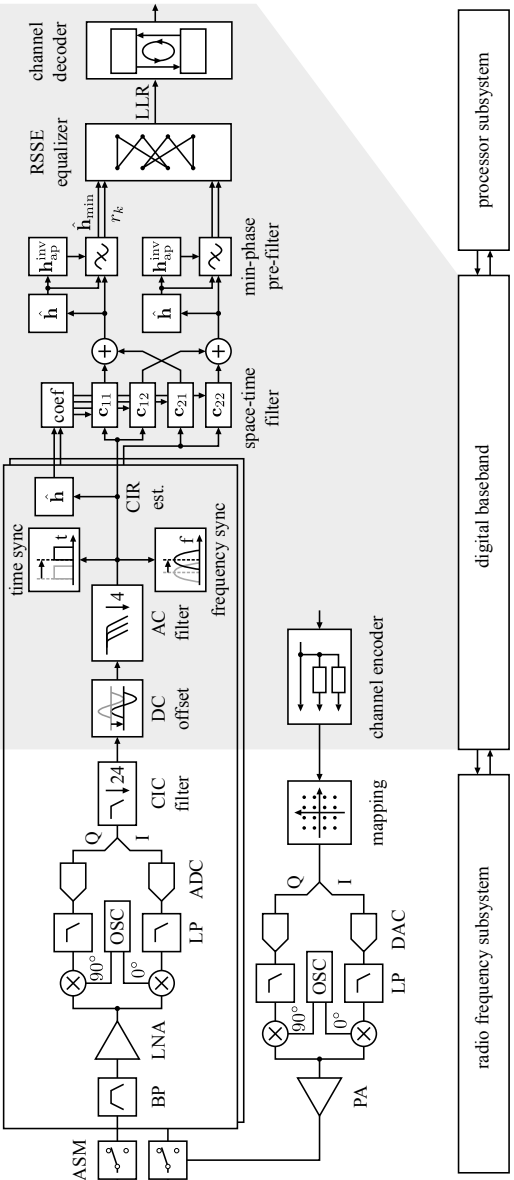


Figure 4.1: Transceiver signal processing tasks and the mapping into PHY subsystems.

The physical layer system is divided into three hardware components as shown in Figure 4.1. The first one is the radio-frequency (RF) subsystem: analog signal processing, ADC, DAC, decimation filter and the I/Q-mapping are attributed to this component. A state-of-the-art RF transceiver of ACP AG is employed of which the receiver is published in [49]. All other signal processing tasks are counted to the digital baseband whose hardware architecture is presented in Section 4.2. For this prototype system, an auxiliary Xilinx Spartan 3 FPGA has been placed between the digital baseband and the RF chip in order to facilitate the configuration of the RF chip and to add the possibility of preprocessing the data prior to passing it to the baseband. However, the utilization of this intermediate FPGA, which was designed prior to the baseband ASIC, limits the PHY system to single-stream operation as data from only one antenna can be captured. The third component is a processor subsystem which does not do any signal processing, but is responsible for interfacing and configuring the baseband as well as for the execution of the higher communication layers. In the presented PHY system, a PowerPC as part of a Virtex4 device of Xilinx is used as processor. A Linux operating system with TCP/IP stack is running on the PowerPC, facilitating the attaching of the device to the network. A picture of the PHY system with the three hardware parts is shown in Figure 4.2.

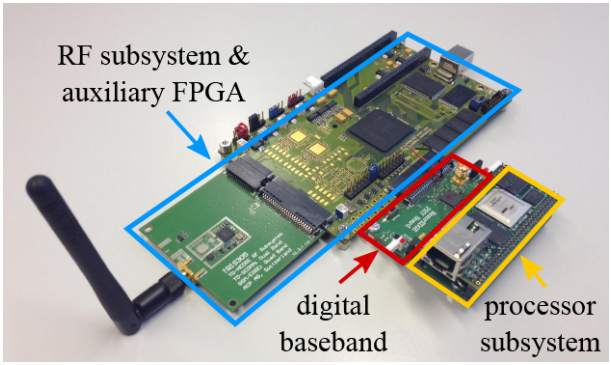


Figure 4.2: Hardware components of the E-EDGE PHY system.

4.2 Transceiver Architecture

The hardware architecture of the digital baseband is partitioned into several sub-blocks as shown in Figure 4.3. The TRX Controller manages the activity in both Receiver and Transmitter, controls the RF subsystem, and has a configuration interface to the processor.

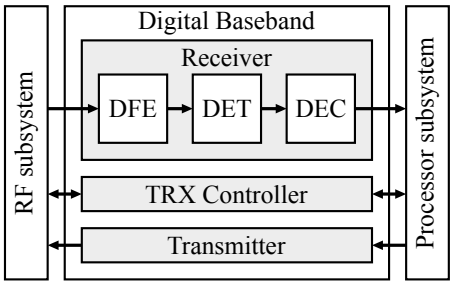


Figure 4.3: Top-level transceiver architecture.

4.2.1 Receiver

The receiver consists of three main blocks, namely the digital front-end (DFE), the detector (DET), and the decoder (DEC).

Digital Front-End

Figure 4.4 shows the DFE architecture which is organized in such a way that the samples flow in a continuous stream from one block to the next, allowing most blocks to work in parallel and thus minimizing the idle time of each block. Whenever the processing of a new burst starts, the first samples are both passed to the DC Estimation block and are kept in the Input Memory. The mean value of the first samples of a burst determines the estimated DC-offset which is then subtracted from all samples of the burst as shown in Figure 4.4. The Input Memory serves at the same time as the delay line of the AC Filter. Eight different sets of filter coefficients are stored in a LUT and the cutoff frequency of the filter can be chosen according to the strength of the signal in the adjacent channel. After the AC

Filter, the data stream is at symbol rate and the samples are passed through the RX Power Estimation block where the strength of the signal is reported to the DFE Controller. The FB Detection and Frequency Offset Estimation block is only active during the initial synchronization phase. A recurring block called frequency burst (FB) which is basically a sinusoid at a known frequency is first detected and subsequently used to estimate the frequency offset. Finally, the Modulation Detection block can detect the modulation scheme of a burst if it is not already known to the DFE Controller.

Since the GSM symbol rate is much lower than any practical processing clock, all blocks can minimize the required hardware by iteratively decomposing their task. This is illustrated in Figure 4.4 by means of the AC Filter which is an FIR filter of order 32, but is realized with a single multiply-accumulate unit.

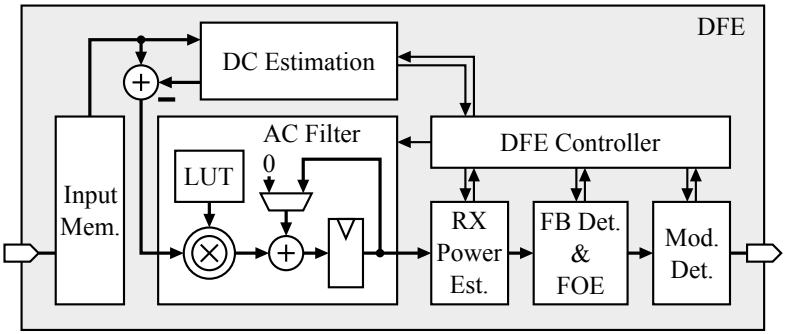


Figure 4.4: Block diagram of the DFE sub-block.

Detector

The next sub-block in the processing chain is the detector whose architecture is depicted in Figure 4.5. In contrast to the stream-based processing in the DFE, the architecture of the DET has a central memory which holds a complete burst and which is accessed by most of the processing blocks. The access is mutually exclusive and is arbitrated by the DET Controller according to the schedule shown in Figure 4.6. While the DET blocks operate on the current burst,

the next burst is processed by the DFE and is simultaneously written to the Burst Memory. Hence, the Burst Memory must be capable of storing both the current and the next burst and, due to the supported RX-diversity, the RAM requires a capacity of four bursts in total.

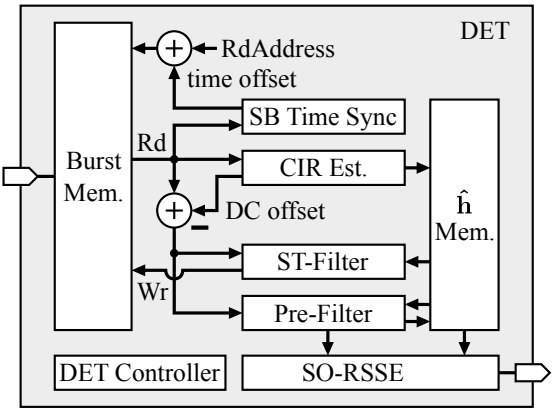


Figure 4.5: Block diagram of DET sub-block.

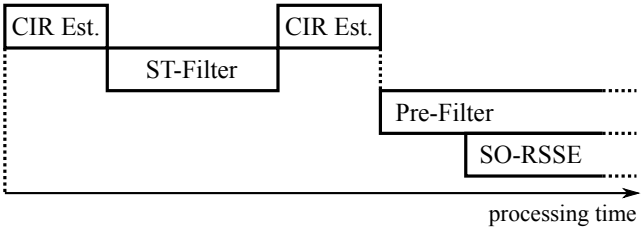


Figure 4.6: Processing schedule of DET.

As shown in Figure 4.6, the DET processing is started by estimating the CIR according to (2.27) in Section 2.8. At the same time, the residual DC-offset is estimated by extending the matrix $\mathbf{T}$ containing shifted versions of the training sequence by an additional column of ones as described in [50]. The estimated CIR and the DC-offset are obtained by multiplying the received version of the

training sequence with the Moore-Penrose pseudoinverse of $\mathbf{T}$. Since the training sequence is a priori known, the inverse matrix can be pre-computed and stored in a LUT. However, this matrix is different for each of the four modulation schemes and for each of the eight possible training sequences, resulting in 32 matrices. Even if the LUT has a significant size, the presented approach is advantageous, since the matrix inversion is omitted and the CIR is estimated in only $19 \cdot 8 = 152$ cycles by employing a single time-shared multiplier. The fast processing time is essential as this minimizes the idle time of the large equalizer block.

The estimated channel taps of both RX streams are stored in the $\hat{\mathbf{h}}$-Memory block, from where they are accessed by ST-Filter, Pre-Filter and SO-RSSE. Since the memory size of 2×8 words does not justify the overhead of using a dedicated RAM macro and since some blocks require a read access of more than one word per cycle, the memory is realized by a flip-flop array. The estimated residual DC-offset is subtracted from all samples that are henceforth read from the Burst Memory as shown in Figure 4.5, thereby omitting the need to read all samples and write the DC-free versions back to the RAM.

Once the CIR estimate is available, the space-time filter coefficients are derived in the ST-Filter block by running a mini-program on a CPU-like hardware architecture. This architectural choice is beneficial as the coefficient calculation involves many different tasks such as matrix multiplication and inversion and since only one coefficient set per burst is computed it is not reasonable to instantiate dedicated hardware for every operation. Subsequently, the content of the Burst Memory is filtered with a filter structure as shown in Figure 4.1 and written back to the same place in the RAM. Since the ST-Filter reduces the level of the CCI, the CIR estimation is repeated and the quality of the second estimate is expected to be of better quality as it is based on data with a higher signal-to-noise-plus-interference ratio.

The Pre-Filter block has both read and write access to the $\hat{\mathbf{h}}$-Memory and first computes the filter coefficients as described in Section 3.1.2 and then filters both the CIR and the received samples. The filtered CIR is written back to the memory and the received samples are directly forwarded to the SO-RSSE whose architecture is described in Section 3.3. Finally, the obtained reliability information is passed to the decoder in the form of LLRs.

Decoder

The last block in the processing chain is the decoder (DEC) with an architecture as depicted in Figure 4.7. Due to the block-rectangular interleaving, the data of a code block is distributed over four consecutive frames. Furthermore, the supported multislot class allows the usage of up to six downlink slots per frame with all bursts belonging to different code blocks. Hence, the Burst Demux block stores the LLRs of all bursts until a complete block is available for deinterleaving. Subsequently, neutral information for the bits that have been punctured in the transmitter is inserted.

The presented architecture has built-in incremental redundancy (IR) management [51]. If IR is enabled, then erroneous blocks are retransmitted with a different puncturing scheme and the probability of successful decoding is increased by taking the mean of the LLRs of the two (or more) transmissions. Since the round-trip time until the retransmission is high (120 ms), it must be expected that many erroneous blocks need to be stored and a RAM capacity of 32′700 LLRs is provided to fulfill the specified IR throughput.

The actual channel decoding is done by a Viterbi decoder for the GPRS/EDGE modes and by a Turbo decoder for the E-EDGE modes. The Turbo decoder has already been published as a standalone ASIC in [52]. Since only one of the two decoder cores is active at a time, the overall memory overhead is reduced by sharing input memory, output buffer, and intermediate memory. The two decoders are both optimized for code rates close to one as required by modern wireless standards with IR.

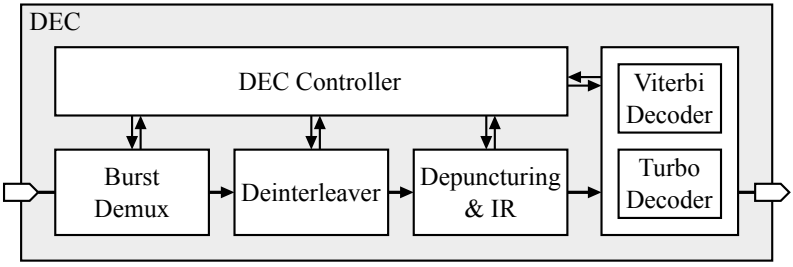


Figure 4.7: Block diagram of DEC sub-block.

4.2.2 Transmitter

Since the mapping to I/Q-constellation points and the pulse-shaping filter is part of the employed RF subsystem, the Transmitter block of the digital baseband is reduced to purely binary operations. Compared to the receiver, the computational complexity is low and the main hardware cost comes from the memories. Figure 4.8 depicts the main blocks of the transmitter. Data coming from the higher communication layers is stored in an input memory. A parity check sum of a code block is calculated and the block is then passed through a convolutional encoder with rate one third. Subsequently, the Puncturer block removes bits from the bit stream according to a predefined pattern to achieve the desired effective code rate before the data is reordered in the Interleaver. The bits of one block are then mapped to four bursts and stored in one of the two Output Buffers. While one buffer is filled with up to 6×4 bursts, the content of the other buffer is passed to the RF subsystem.

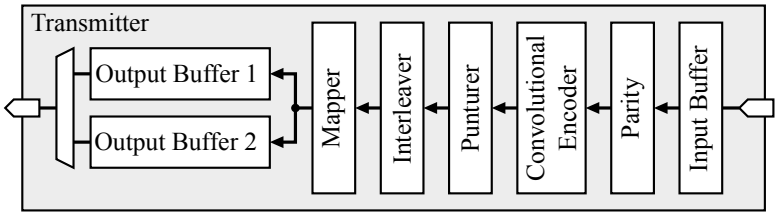


Figure 4.8: Block diagram of the baseband transmitter.

4.2.3 TRX Controller

The TRX Controller block issues commands to both transmitter and receiver and evaluates reports coming from the receiver in order to adjust the controlling of the RF subsystem.

Synchronization in Time

One of the main tasks of the TRX Controller is to maintain the GSM time base. This task is split into three parts. First, the coarse timing

is achieved by detecting the frequency burst in the FB Detection block. This allows the reception of the synchronization burst (SB) which follows exactly eight bursts later. Second, by doing a cross-correlation with the known SB midamble, the exact time offset in symbols is determined in the SB Time Sync block. The RAM address of subsequent read accesses on the SB is now corrected by the time offset, such that all other DET blocks do not need to care about this offset (see Figure 4.5). Third, the synchronization in time is maintained by tracking the strongest CIR tap. If the strongest channel tap is not in the desired location, an offset is reported to the DET Controller and the time-base is adjusted accordingly.

Automatic Gain Control

The total gain of the various gain stages in the RF processing chain can either be fixed at a constant level or the result of the RX Power Estimation block in the DFE can be used to automatically adjust the gain. To this end, the reported power level is compared to a configurable target level and the difference is subsequently low-pass filtered by a low-complexity first-order IIR filter. The filter output is then used to determine a new gain setting for the reception of the next burst.

Interfaces

Also part of the TRX Controller are configuration registers which are mapped to the address space of the Linux system running on the processor. The access to the registers is realized as a standard compliant APB interface [53] with eight-bit wide read, write and address busses. The same interface is used to read and write uplink and downlink data. On the other side, the digital baseband is connected to the auxiliary FPGA with a parallel data interface and from there the RF chip is accessed over a serial DigRF v1.12 interface [54].

4.2.4 Clocking Strategy

The integration of the ASIC into a complete PHY system is facilitated by providing independently clocked interfaces. Both APB and RF

interface operate in the clock domain of the interface clock and the asynchronous clock domain transition is handled on-chip as shown in Figure 4.9. Most of the transitions are realized by inserting an asynchronous FIFO into the data stream and the configuration registers are protected by a handshake protocol such that they can be accessed from both domains. The majority of the FIFOs offer a storage capacity of only a few elements and are thus realized with a flip-flop array.

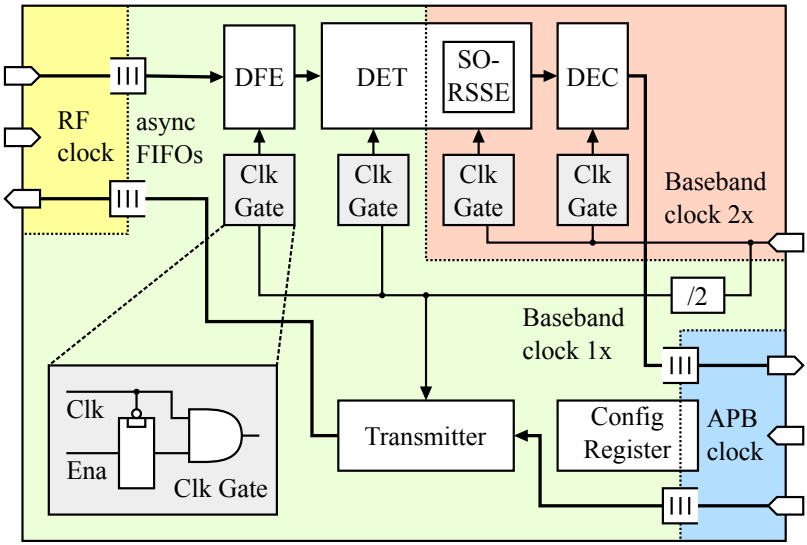


Figure 4.9: Clock domains and manual clock gates.

The scope of the interface clocks is limited to a very small area for two main reasons. First, the small number of clocked instances allows for a small and flat clock tree structure with a low insertion delay what is favorable in source-synchronous interfaces. Second, the clock frequency is fitted to the interface and not to the signal processing blocks and it is therefore advantageous to have an independent clock for the baseband processing.

The ASIC is provided with a baseband processing clock over a separate pin as shown in Figure 4.9. This clock is internally divided,

resulting in two synchronous clocks at single and double speed. The computationally complex equalizer and channel decoding blocks are clocked with the double-speed clock whereas all other blocks run at single speed. On the one hand, providing SO-RSSE and DEC with a fast clock ensures a sufficiently high throughput while the degree of parallelization is kept at a reasonable level. On the other hand, the computationally less complex blocks can do excessive resource sharing even at the lower clock speed and the longer clock period results in a smaller circuit size for the combinational logic.

Even if flip-flops with a common enable signal are automatically equipped with a clock gate by the synthesis tool, these clock gates are distributed over the whole chip and the connecting clock tree is of substantial size. Furthermore, due to inattention of the designer, some counters or state machines could be implemented in such a way that they also run during the idle time of the block. To save the energy required to constantly charge and discharge these nets, manual clock gates have been instantiated at the root of the tree to control the clock of the large processing blocks as shown in Figure 4.9. Whenever a block is idle, the whole clock tree of this block is shut down.

4.3 VLSI Implementation Results

The complete baseband transceiver has been manufactured in the SMIC 130 nm CMOS technology. A micrograph of the ASIC named RazorEDGE is shown in Figure 4.10. The total core area is 6.0 mm^2 and the approximate regions of the large blocks are indicated on the micrograph. As expected, the DEC block is the largest block and occupies almost 50 % of the area whereby the RAMs for the burst demultiplexing at the DEC input and the storage for IR make up almost half of this area. Another 30 % of the area is taken by the DET block with the SO-RSSE and the ST-Filter being the largest blocks.

The ASIC has been bonded into a QFN56 package and functional tests have been run on a digital tester. The maximal operating frequency of the fast baseband clock has been found to be 155 MHz. Since the longest path is in the ST-Filter which is only used for

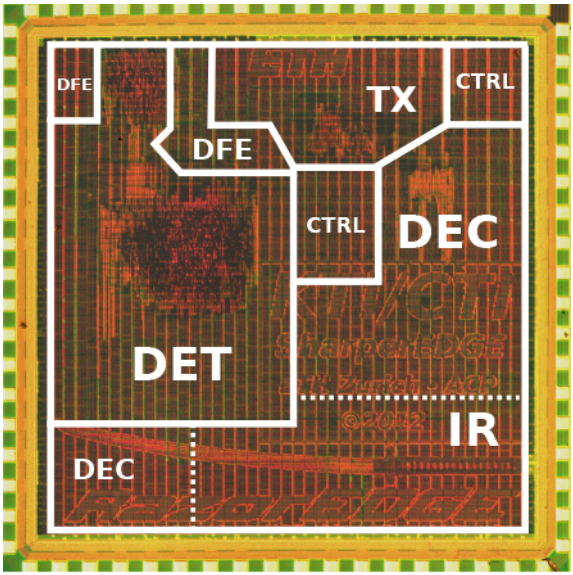


Figure 4.10: Micrograph of the RazorEDGE baseband transceiver ASIC.

RX-diversity modes, the clock frequency can be increased to 164 MHz when operating in single-stream mode.

Power Consumption

Besides the identification of the maximal clock speed, the digital tester has also been used to assess the power consumption of RazorEDGE under different multislot scenarios and for different clock frequencies. Even if the architecture would allow the clock speed of the baseband clock to be chosen arbitrarily, the tester setup limits the frequency to powers of two of the RF clock which is fixed to 26 MHz by the GSM standard. To achieve a sufficiently high throughput in the highest multislot configuration with 6 out of 8 bursts used for data reception, the fast baseband clock must run at 104 MHz. Also the long combinational path used in RX-diversity modes requires the clock to

run at this speed. For 5 and less time-slots in single-stream mode, a clock frequency of 52 MHz is sufficient.

While the leakage current of the core supply is only 120 μA at the nominal supply voltage of $V_{\mathrm{DD}} = 1.2\,\mathrm{V}$, the dynamic power consumption is of more interest. The power consumption of a single node in the circuit is given by the well-known formula [43]

$$P_{\mathrm{dyn}} = \alpha f C V_{\mathrm{DD}}^2 \tag{4.1}$$

where f is the clock frequency, C the overall capacitance of the node, and α is the activity which is given by the average proportion of cycles in which the polarity of the node changes. Reducing the clock frequency would not reduce power consumption in an ideal implementation where everything is completely turned off during idle time as α would increase by the same factor. However, there are some blocks and clock trees in the ASIC which run continuously and halving the clock frequency reduces the power consumption by approximately 5 mW as illustrated in Figure 4.11.

Since the maximal clock frequency of 155 MHz is well above the target clock frequency of 104 MHz, all combinational paths have settled long before the arrival of the next clock edge. This margin can be exploited to save power by reducing the supply voltage and accept the longer settling time. Functional tests have shown that the circuit works with a supply voltage of only 80 % of the nominal value, leading to a power consumption reduced by 36 % as shown in Figure 4.11. The timing margin for the RX-diversity mode is slightly smaller and voltage scaling can push the power consumption from 56 mW to 39 mW (-30 %).

Multislot configurations with up to 5 slots can be operated at half the clock frequency, resulting in an increased timing margin which allows a further reduction of the supply voltage. With the combination of both clock frequency reduction and voltage scaling it is possible to reduce the power consumption by 60 %.

The effect of the manually instantiated clock gates at the root of the clock tree has also been measured for the case of 6 RX time-slots and DAS12. Surprisingly, if these clock gates are disabled, the power consumption is 44 % higher. It cannot be distinguished whether this large difference stems from the toggling of the clock tree beyond the

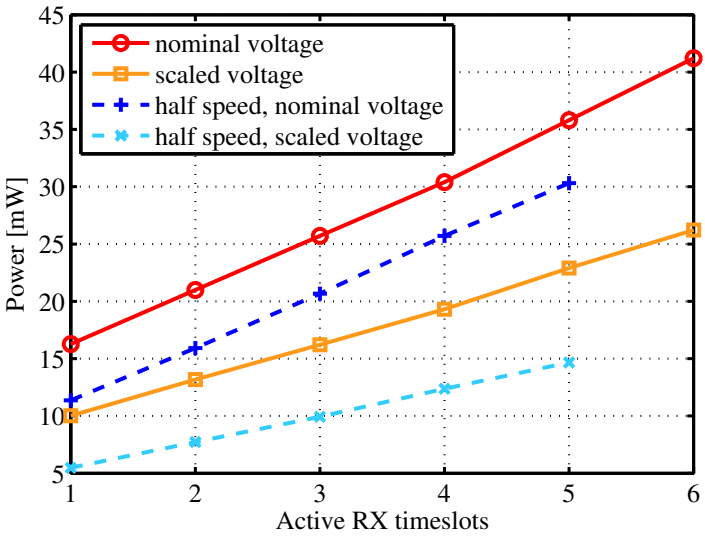


Figure 4.11: Power consumption of the highest throughput mode DAS12 (32-QAM) as a function of the number of active time-slots per 8-slot frame at nominal and scaled supply voltage.

clock gate or from circuit parts that are not properly shut down during idle time. It is however certain that manual clock gating on a very coarse level helps to substantially reduce the power consumption of a large digital system.

4.4 Receiver Performance

The performance of the receiver has been assessed in a Matlab simulation framework where all impairments of the channel and the RF chain have been modeled. In order to verify the simulation framework and to demonstrate the real-time capability of the ASIC, the BLER under both sensitivity and interference scenarios has been measured.

4.4.1 Measurement Setup

The PHY system of Figure 4.2 has been put into a measurement setup as depicted in Figure 4.12. A standard compliant E-EDGE traffic channel and a beacon channel used for synchronization are generated by Rhode & Schwarz equipment. To this end, the transmitted data is encoded and modulated according to the desired modulation and coding scheme of the traffic channel and the dedicated synchronization bursts FB and SB are compiled to the beacon channel. The baseband signals of beacon and traffic channel are then shifted apart in frequency and summed up to a multicarrier baseband signal as depicted in Figure 4.13. For adjacent channel measurements, an additional GSM signal with the desired strength is added to the baseband signal 200 kHz above the traffic channel. The R&S AMIQ sends the I- and Q-component of the baseband signal to the R&S SMIQ on which the multipath channel is emulated. Subsequently, the signal is converted to radio frequency and transmitted at a level as adjusted by the Matlab script that measures the BLER performance of the system.

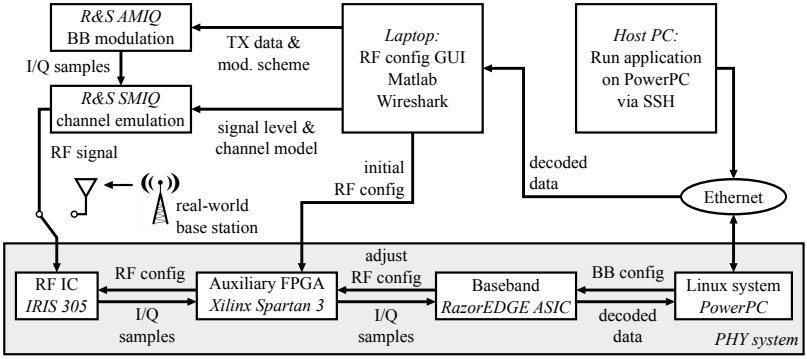


Figure 4.12: Measurement setup based on the PHY system described in Section 4.1.

The RF IC is initialized by software running on the laptop via the auxiliary FPGA. Configuration parameters related to timing, gain and frequency are automatically adjusted during the measurement based on feedback from the baseband system. The configuration of

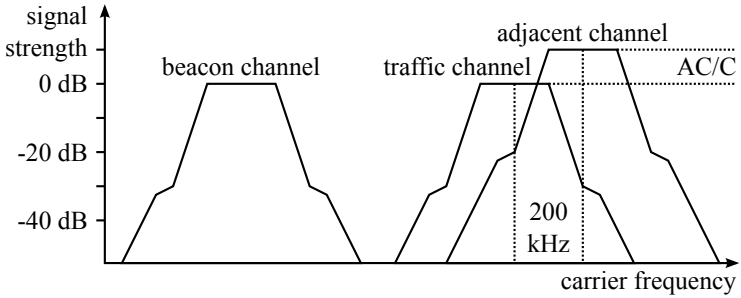


Figure 4.13: Channel allocation in the measurement setup. Each channel is represented by the spectral mask specified in [15].

the baseband ASIC is done by running an application on the PowerPC/Linux system which is started by logging in from a host PC via SSH. This application also fetches the decoded data from RazorEDGE and forwards it over the Ethernet to the Laptop where the Matlab script evaluates the BLER by comparing the decoded data with the originally transmitted version.

4.4.2 Sensitivity Performance

For the GMSK modulated voice channel, the standard requires a frame-erasure rate of 10^{-3} at a signal level of -102 dBm [15]. As shown in Figure 4.14, this requirement is exceeded by almost 10 dB. As expected, the array gain offered by the second RX antenna further improves the sensitivity by 3 dB. Also illustrated in Figure 4.14 is the good match between simulation results and measurements [1] and the small implementation loss stemming from the fixed-point number representation in the ASIC. In Table 4.1, the TCH/FS voice channel sensitivity performance for the single antenna case which is a common benchmark for GSM/EDGE systems is compared to prior art and the presented solution shows the best performance even if the margin is very small.

[1] For the conversion of simulated SNR to received signal power, the noise figure of the RF receiver as reported in [55] was used.

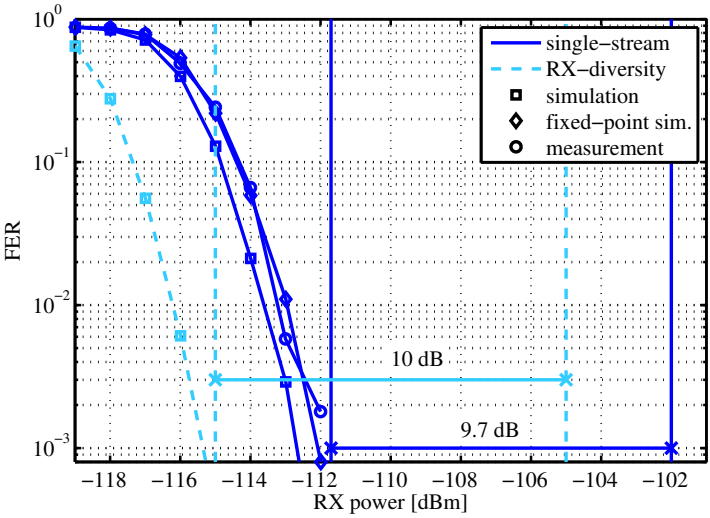


Figure 4.14: Measured and simulated frame-erasure rate for the TCH/FS voice channel compared to the reference sensitivity performance.

By employing the channel emulator module of the R&S SMIQ, the BLER performance under the specified TU channel conditions with a relative velocity of 50 km/h between base station and cell phone has been measured. The movement of the mobile station leads to time-varying channel taps and since in our solution the CIR is estimated only once per burst based on the midamble, the detection reliability decreases for symbols at the beginning and the end of the burst. However, the CIR estimation error at 50 km/h is only small and eventually all errors can be corrected by forward error correction. The measured BLER performance for an 8-PSK and a 32-QAM E-EDGE modulation scheme is shown in Figure 4.15 where it is compared to the specified reference sensitivity which is met with a large margin. Unfortunately, the match between measured and simulated single-stream performance is not as good as in the case of a static channel. The array gain as well as the diversity gain offered by the second RX stream significantly improves the BLER performance

Sensitivity performance [dBm], static channel			
3GPP	[56]	[57]	this work
-102	-110.1	-111.3	-111.7

Table 4.1: Comparison of single antenna TCH/FS voice channel sensitivity performance.

under fading conditions as shown in Figure 4.15. The RX-diversity curves cannot be compared to any reference sensitivity, since no values are specified to date for these E-EDGE scenarios by 3GPP.

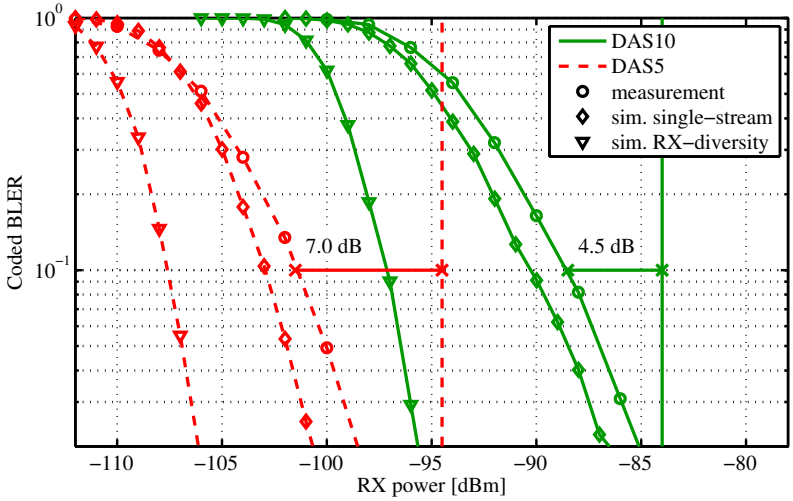


Figure 4.15: Measured and simulated BLER of the E-EDGE modes DAS5 (8-PSK) and DAS10 (32-QAM) under TU50 channel conditions put in relation to the reference sensitivity performance.

4.4.3 AC Interference Performance

The GSM frequency bands are divided into 200 kHz frequency slots which are assigned to different users. Base stations adjust the signal

levels of different users independently of each other, thereby transmitting with more power to users far away. A mobile station located close to the base station may therefore receive a desired signal at normal strength and at the same time a much stronger signal intended for the far-away user in the adjacent channel (AC). Since the spectral shape of the transmitted signal is not strictly limited to 200 kHz [15], a strong signal on the adjacent channel leads to a significant spectral overlap of the desired signal by the interfering signal as illustrated in Figure 4.13. The ratio of the adjacent channel power to the power in the desired traffic channel is typically given in dB and is denoted by AC/C.

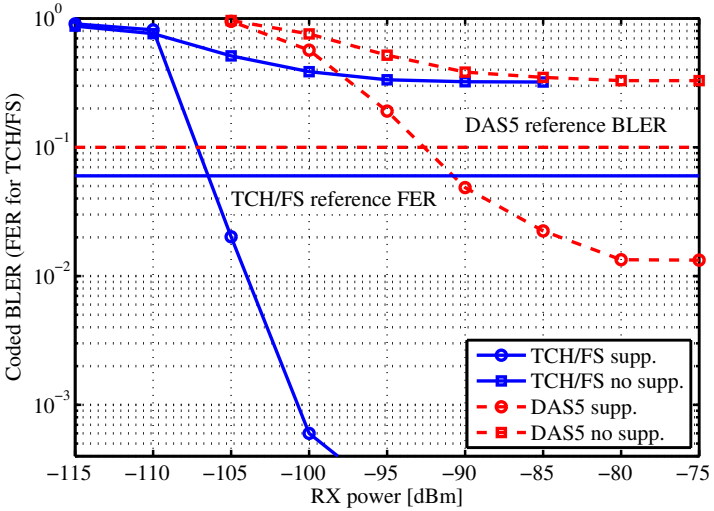


Figure 4.16: Measured BLER/FER performance in the presence of an adjacent channel of strength AC/C=9 dB for TCH/FS and AC/C=2 dB for DAS5, respectively. The employed channel model is TU50.

In order to reduce the interference of the adjacent channel, the cutoff frequency of the decimation filter in the DFE can be lowered to values close to 100 kHz or even below. Naturally, a low cutoff frequency also reduces the bandwidth of the desired signal, resulting in

colored noise and additional ISI, what exacerbates the signal detection in the equalizer. Figure 4.16 shows the BLER performance of a GMSK and an 8-PSK modulated scheme with an interfering adjacent channel at an AC/C ratio for which the reference interference performance [15] must be achieved. The performance has been measured both with a large cutoff frequency resulting in no AC suppression and with a cutoff frequency which is optimal for the given modulation order and AC/C ratio. The measurement shows that the implemented AC interference suppression mechanism is effective and the reference interference performance can be achieved. The margin to the specification in terms of AC/C ratio is illustrated in Figure 4.17, where the AC/C is varied at a given RX power of the desired signal.

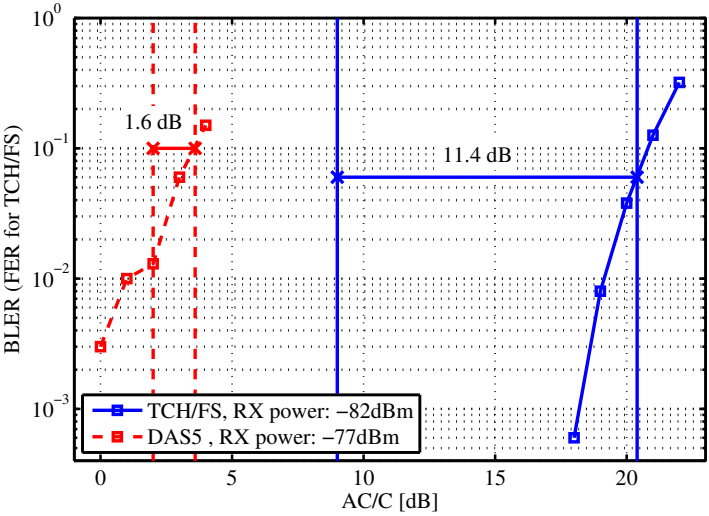


Figure 4.17: Measured BLER/FER performance at constant RX power of the desired signal and varying AC strength under TU50 channel conditions. The margin to the reference interference performance is also shown.

4.4.4 Synchronization to Base Station

As an alternative to the BLER measurements based on artificial data from R&S equipment, the synchronization to real-world base stations has been demonstrated and the system information which is broadcast on the beacon channel has been displayed on the laptop in a GSM-specific Wireshark extension. This shows that the developed PHY solution is capable of handling real-world conditions and can synchronize in time and frequency.

Chapter 5

Conclusion and Outlook

The evolution of the GSM/EDGE standard is a necessity to consolidate its position as a reliable fallback solution where 3G/4G systems are not available and to reduce the risk of 2G network decommissioning in some countries what would degrade its advantage of being the only universal cellular standard. This advantage is one of the reasons that could make Evolved EDGE the technology of choice for the emerging machine-to-machine (M2M) communication systems. However, small, power-efficient and inexpensive solutions must be available to fulfill the requirements of as many of the numerous M2M applications as possible. The first-ever E-EDGE baseband transceiver ASIC presented in this thesis demonstrates that such a solution is feasible, even if the high modulation orders and the exploitation of receive diversity requires careful evaluation and optimization of different equalization algorithms as well as sophisticated VLSI architectures.

In the following, the key findings are summarized separately for the two main topics of this thesis and ideas for future improvements are presented. In Section 5.1, conclusions about the investigated channel equalization solutions are drawn and in Section 5.2, the benefits of the developed PHY system are summarized and an outlook to the ultimate goal of a complete modem for M2M communication is given.

5.1 Channel Equalization

Since the complexity of the optimal maximum-likelihood sequence estimation (MLSE) algorithm for hard-output symbol detection grows exponentially with the number of bits per symbol and inter-symbol interference (ISI) can extend to as many as 7 symbols in 2G systems, algorithms with reduced complexity must be employed to tackle the 16-QAM and 32-QAM modulated data of Evolved EDGE. The same applies to the maximum a-posteriori (MAP) symbol detection which is optimal for soft-outputs, where the exploding number of states affects the hardware complexity even more since the forward state metrics must be stored for the final log-likelihood ratio (LLR) computation during the backward recursion.

The combination of reduced-state sequence estimation (RSSE) with a linear pre-filter that transforms the overall channel to its minimum phase equivalent drastically reduces the computational complexity [5–7]. Hence, the part of the thesis about channel equalization is focused on these two aspects and the following conclusions are drawn:

1. The closely-spaced constellation points of 32-QAM require an increased accuracy in the minimum-phase pre-filter which can be achieved by a higher filter order. The presented homomorphic pre-filter coefficient computation VLSI circuit improves the hardware efficiency by a factor five when compared to a state-of-the-art solution based on linear prediction.

2. The optimal RSSE configurations for the high modulation orders of E-EDGE in combination with our homomorphic pre-filter solution have been derived. An RSSE with 16 states for 32-QAM is sufficient to outperform [10] in terms of bit-error rate and hardware efficiency.

3. Algorithmic evaluation has revealed that soft-output RSSE with omitted backward recursion as required for (max-log) MAP symbol equalization offers the most favorable tradeoff between complexity and performance. The developed SO-RSSE VLSI solution is optimized to such an extent that it requires even less

silicon area than the hard-output solution[1] even if the SO-RSSE delivers reliability information and exploits RX-diversity.

4. Even if the block-error rate (BLER) performance of the soft-output M-algorithm is promising, the resulting VLSI solution shows significantly worse area and throughput measures than the SO-RSSE solution due to the high number of sorting operations.

Soft-output equalization in combination with RX-diversity yields BLER performance gains of up to 10 dB (see Figure 2.19) compared to the single-stream hard-output variant of [10]. A further significant performance improvement could be expected by feeding the result of the channel decoder back to the equalizer and improve the quality of the estimated bits iteratively [58, 59]. However, such a solution would drastically change the receiver architecture, significantly increase the storage requirements, and impose increased throughput requirements on both equalizer and decoder.

5.2 System Design

The highly optimized VLSI designs of homomorphic pre-filter and soft-output RSSE build the foundation of the digital transceiver ASIC presented in Chapter 4. The architecture of the receiver part is divided into the three main blocks digital front-end, detector, and decoder and all blocks operate concurrently on different junks of data what reduces the idle time of the individual blocks and facilitates the memory management compared to a solution with one central memory.

The baseband ASIC has been joined with an RF ASIC and a processor to a complete physical layer (PHY) system. Owing to the autonomous interaction between baseband and RF ASIC, the PHY system can synchronize in frequency and in time to real world base stations. Furthermore, the system has been put into a measurement

[1] The SO-RSSE solution implemented in 130 nm CMOS profits from the smaller technology when compared to the HO-RSSE fabricated in 180 nm. In terms of gate-equivalents, the hardware complexity of SO-RSSE is 50 % more.

setup and the assessed BLER performance meets the standard requirements with a large margin. Especially the measured sensitivity performance is excellent both under fading channel conditions and over a static channel. For the legacy GSM modes, a comparison of the static channel sensitivity shows that the developed system matches the performance of prior art.

Towards an E-EDGE Modem for M2M Communication

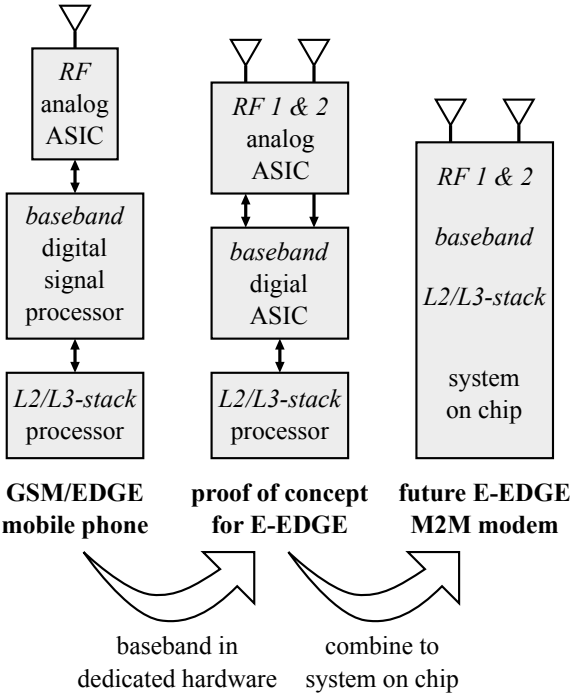


Figure 5.1: Evolution of 2G modem from a single-antenna GSM/EDGE solution based on a digital signal processor towards a system on chip with dedicated hardware for baseband processing as required for M2M communication based on Evolved EDGE.

For single-antenna GSM/EDGE mobile phones, a solution based on a digital signal processor as illustrated in Figure 5.1 is appropriate since the computational load of channel equalization and decoding is efficiently manageable on such a processing platform. However, the increased receiver complexity for handling the Evolved EDGE features like RX-diversity, Turbo coding, and high-order modulation calls for baseband processing on dedicated hardware. Owing to the careful algorithmic evaluation and architectural circuit optimization, the baseband ASIC presented in this thesis can cope with these challenges at low power consumption and minimal silicon area as required for M2M communication modems. Hence, it is a first step towards the ultimate goal of an Evolved EDGE modem for M2M communication with RF-, baseband- and processor-subsystems united on a single system on chip.

Appendix A

Notation and Acronyms

Symbols

$\mathbf{b}$	B-dimensional binary-valued input vector of transmitter
$\hat{\mathbf{b}}$	estimate of $\mathbf{b}$ at output of receiver
R	coding rate
$\mathbf{x}$	encoded binary vector of length B/R
$\hat{\mathbf{x}}$	estimate of $\mathbf{x}$
Z	number of bursts per encoded block
$\mathbf{a}$	vector of known tail symbols at start and end of each burst
$\mathbf{t}$	vector of known midamble symbols (training sequence)
$\mathbf{T}$	matrix consisting of shifted time-reversed versions of $\mathbf{t}$
$\mathbf{s}$	vector of transmitted symbols
N	number of information symbols per half-burst
$\hat{\mathbf{s}}$	estimate of information symbols in $\mathbf{s}$
Q	number of bits per symbol
$\mathcal{O}$	complex-valued symbol alphabet with cardinality 2^Q
s_k	transmitted symbol at time k
$x_{q,k}$	q-th bit in k-th symbol of $\mathbf{s}$
$\mathbf{h}$	L-dimensional complex-valued vector containing the CIR
$\hat{\mathbf{h}}$	estimate of CIR

$\mathbf{w}$	vector of additive white Gaussian noise
$\mathbf{r}$	vector of received samples
$\mathcal{CN}(x, y)$	complex normal distribution with mean x and variance y
σ^2	noise variance
$\mathbf{I}_N$	N-dimensional identity matrix
$\mathcal{C}_k$	set of all trellis states at time k
J_m	number of subsets at position m in state description
$\mathcal{P}^i_{k-m}$	subset i at position m of state description at time k
K	number of trellis states
D	decision delay
M	number of retained paths in the M-algorithm
$\alpha_{k-1}(c')$	forward state metric of state $c' \in \mathcal{C}_{k-1}$
$\beta_k(c)$	backward state metric of state $c \in \mathcal{C}_k$
$\gamma_k(c', c)$	branch metric of branch between states c and c'
A_{k-1}	forward state metric α_{k-1} in negative logarithmic domain
B_k	backward state metric β_k in negative logarithmic domain
Γ_k	branch metric γ_k in negative logarithmic domain
$\chi(c)$	set of all predecessor states of $c \in \mathcal{C}_k$
$\Psi(c')$	set of all successor states of $c' \in \mathcal{C}_{k-1}$
$\mathcal{B}^{\pm 1}_{x_{q,k}}$	set of all branches corresponding to $x_{q,k} = \pm 1$
$\tilde{r}_{i,k}$	reference signal of i-th stream at time k
e_s	partial reference signal depending on state symbols
e_b	partial reference signal depending on branch symbol
$\mathbf{d}_k$	difference vector between reference and received signals of both streams at time k
p	filter order of pre-filter
$\mathbf{h}^{\min}$	minimum-phase equivalent of $\mathbf{h}$
$\mathbf{g}$	DFT of $\mathbf{h}$
$\mathbf{f}$	real cepstrum of $\mathbf{h}$

Operators

$\Re(\cdot)$	real part of complex number
$\Im(\cdot)$	imaginary part of complex number

$\exp(\cdot)$	natural exponential function
$\log(\cdot)$	natural logarithm
$\log_2(\cdot)$	base-2 logarithm
$(\cdot)^H$	hermitian transposition
$(\cdot)^{-1}$	matrix inversion
$\lvert\cdot\rvert$	cardinality of set or absolute value of complex number
$\lVert\cdot\rVert$	Euclidean vector norm (ℓ^2-norm)
$\mathrm{p}(\cdot)$	probability density function
$P[\cdot]$	probability of an event
$\mathbb{E}[\cdot]$	expectation of a random variable

Acronyms

2G	second generation of mobile telecommunications technology (GSM/(E-)EDGE)
2.75G	marketing name for Evolved EDGE
3G	third generation of mobile telecommunications technology (UMTS/HSDPA)
3GPP	Third Generatioin Parnership Project
4G	fourth generation of mobile telecommunications technology (LTE)
ASIC	application-specific integrated circuit
BCJR	MAP algorithm by Bahl, Cocke, Jelinek and Raviv
(U)BER	(uncoded) bit-error rate
BLER	block-error rate
CCI	co-channel interference
CIR	channel impulse response
CSI	channel state information
DDFSE	delayed decision feedback sequence estimation
(I)DFT	(inverse) discrete Fourier transform

DSL	digital subscriber line
E-EDGE	Evolved EDGE
EDGE	Enhanced Data rates for GSM Evolution
(I)FFT	(inverse) fast Fourier transform
FIR	finite impulse response
GMSK	Gaussian minimum-shift keying
GPRS	general packet radio service
GSM	Global System for Mobile Communications
HSDPA	High-Speed Downlink Packet Access
HT	hilly terrain
ISI	inter-symbol interference
LLR	log-likelihood ratio
LP	linear prediction
LTE	Long-Term Evolution
M2M	machine-to-machine
MAP	maximum a-posteriori
MLSE	maximum likelihood sequence estimation
MMSE	minimum mean-square error
PHY	physical layer
PSK	phase-shift keying
QAM	quadrature amplitude modulation
RF	radio frequency
RSSE	reduced-state sequence estimation
RTTI	reduced transmission time interval

RX	receiver
SNR	signal-to-noise ratio
STF	space-time filter
TCP/IP	Transmission Control Protocol / Internet Protocol
TU	typical urban
TX	transmitter
UMTS	Universal Mobile Telecommunication System
VLSI	very large scale integration
WLAN	wireless local area network
WPAN	wireless personal area network (Bluetooth, ZigBee, etc.)

Bibliography

[1] OECD, "Machine-to-machine communications: Connecting billions of devices," *OECD Digital Economy Papers*, no. 192, 2012.

[2] S. Kechiche, D. George, and N. Jain, "From concept to delivery: The M2M market today," *GSMA Intelligence*, 2014.

[3] Mapviewer of Federal Office of Communications. [Online]. Available: map.geo.admin.ch

[4] 3G Americas, "The case for Evolved EDGE," *3G Americas White Papers*, 2008.

[5] W. H. Gerstacker and R. Schober, "Equalization concepts for EDGE," *IEEE Trans. on Wireless Communications*, vol. 1, no. 1, pp. 190–199, 2002.

[6] M. Eyuboglu and S. Qureshi, "Reduced-state sequence estimation for coded modulation of intersymbol interference channels," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 6, pp. 989–995, Aug 1989.

[7] C. Benkeser, "Power efficiency and the mapping of communication algorithms into VLSI," Ph.D. dissertation, ETH Zürich, Switzerland, Series in Microelectronics, vol. 209, Hartung-Gorre Verlag Konstanz, 2010.

[8] W. Koch and A. Baier, "Optimum and sub-optimum detection of coded data disturbed by time-varying intersymbol interference," in *Proc. of IEEE Globecom*, vol. 3, Dec 1990, pp. 1679–1684.

[9] J. B. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Trans. on Communications*, vol. 32, no. 2, pp. 169–176, 1984.

[10] C. Benkeser, A. Bubenhofer, and Q. Huang, "A 4.5mW digital baseband receiver for level-A Evolved EDGE," in *IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2010, pp. 276–277.

[11] T.-H. Wu, H.-H. Chang, S.-F. Chen, C.-S. Chiu, and Lai, "A 65-nm GSM/GPRS/EDGE SoC with integrated BT/FM," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 5, pp. 1161–1173, May 2013.

[12] A. Duel-Hallen and C. Heegard, "Delayed decision-feedback sequence estimation," *IEEE Trans. on Communications*, vol. 37, no. 5, pp. 428–436, May 1989.

[13] *Physical layer on the radio path; General description*, 3GPP TS 45.001, Rev. 12.0.0, Sep. 2014.

[14] *Modulation*, 3GPP TS 45.004, Rev. 12.0.0, Sep. 2014.

[15] *Radio transmission and reception*, 3GPP TS 45.005, Rev. 12.0.0, Sep. 2014.

[16] R. G. Gallager, "Circularly-symmetric gaussian random vectors," 2008.

[17] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.

[18] G. D. Forney, "Maximum-likelihood sequence estimation of digital sequences in the presence of intersymbol interference," *IEEE Trans. on Information Theory*, vol. 18, no. 3, pp. 363–378, 1972.

[19] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. on Information Theory*, vol. 28, no. 1, pp. 55–67, Jan 1982.

[20] W. Gerstacker, F. Obernosterer, R. Meyer, and J. Huber, "On prefilter computation for reduced-state equalization," *IEEE Trans. on Wireless Communications*, vol. 1, no. 4, pp. 793–800, Oct 2002.

[21] C. Benkeser, A. Bubenhofer, and Q. Huang, "A 1mm^2 1.3mW GSM/EDGE digital baseband receiver ASIC in 0.13μm CMOS," in *Proc. IEEE/IFIP VLSI-SoC*, Sept 2010, pp. 183–188.

[22] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. on Information Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.

[23] K.-w. Wong, C.-y. Tsui, R.-K. Cheng, and W.-h. Mow, "A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels," in *Proc. of IEEE Int. Symp. on Circuits and Systems (ISCAS)*, vol. 3. IEEE, 2002, pp. III–273.

[24] V. Franz and J. B. Anderson, "Concatenated decoding with a reduced-search BCJR algorithm," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 2, pp. 186–195, 1998.

[25] D. Fertonani, A. Barbieri, and G. Colavolpe, "Reduced-complexity BCJR algorithm for turbo equalization," *IEEE Trans. on Communications*, vol. 55, no. 12, pp. 2279–2287, 2007.

[26] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 491–503, 2006.

[27] J. Widmer, "Channel equalization for Evolved EDGE - VLSI implementation of a soft-output Viterbi equalizer," Master's thesis, ETH Zürich, Switzerland, 2012.

[28] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[29] J.-W. Liang and A. Paulraj, "Two stage CCI/ISI reduction with space-time processing in TDMA cellular networks," in *IEEE Conference on Signals, Systems and Computers*, vol. 1, Nov. 1996, pp. 607–611.

[30] E. Yakhnich, "Channel estimation for EGPRS modems," in *Proc. of IEEE Vehicular Technology Conf. (VTC),* Spring, vol. 1. IEEE, 2001, pp. 419–422.

[31] M. Schmidt and G. Fettweis, "Fractionally-spaced prefiltering for reduced state equalization," in *Proc. of IEEE Globecom*, vol. 5, 1999, pp. 2291–2295.

[32] T. Detert, "Prefiltered low complexity tree detection for frequency selective fading channels," in *Proc. of IEEE Vehicular Technology Conf. (VTC),* Spring, Apr 2007, pp. 2364–2368.

[33] U. Dang, W. Gerstacker, and D. Slock, "Maximum SINR prefiltering for reduced-state trellis-based equalization," *Proc. of IEEE Int. Conf. on Communications (ICC)*, June 2011.

[34] C. Benkeser, S. Zwicky, H. Kröll, J. Widmer, and Q. Huang, "Efficient channel shortening for higher order modulation: Algorithm and architecture," in *Proc. of IEEE Int. Symp. on Circuits and Systems (ISCAS).* IEEE, 2012, pp. 2377–2380.

[35] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing, 1st edition.* Prentice-Hall, Inc., 1989.

[36] J. Volder, "The CORDIC trigonometric computing technique," *IRE Trans. Electronic Comp.*, vol. EC-8, no. 3, pp. 330–334, Sep. 1995.

[37] R. Andraka, "A survey of CORDIC algorithms for FPGA based computers," in *Proc. ACM/SIGDA FPGA*, New York, USA, 1998, pp. 191–200.

[38] E. Haratsch and K. Azadet, "High-speed reduced-state sequence estimation," in *Proc. of IEEE International Symposium on Circuits and Systems*, vol. 3, May 2000, pp. 387–390.

[39] E. Haratsch and K. Azadet, "A 1-Gb/s joint equalizer and trellis decoder for 1000BASE-T Gigabit Ethernet," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 3, pp. 374–384, March 2001.

[40] E. Haratsch and Z. Keirn, "Digital signal processing in read channels," in *Proc. of IEEE Custom Integrated Circuits Conference*, 2005, pp. 683–690.

[41] C. Benkeser and Q. Huang, "Design and optimization of a digital baseband receiver ASIC for GSM/EDGE," in *VLSI-SoC: Forward-Looking Trends in IC and Systems Design*, ser. IFIP Advances in Information and Communication Technology. Springer Boston, 2012, vol. 373, pp. 100–127.

[42] S. Zwicky, C. Benkeser, A. Burg, and Q. Huang, "Efficient VLSI implementation of reduced-state sequence estimation for wireless communications," in *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2013, pp. 2528–2532.

[43] H. Kaeslin, *Digital integrated circuit design: from VLSI architectures to CMOS fabrication*. Cambridge University Press, 2008.

[44] A. Varathan, "Evaluation and implementation of an M-BCJR based soft-output equalizer for Evolved EDGE," Master's thesis, ETH Zürich, Switzerland, 2014.

[45] K. E. Batcher, "Sorting networks and their applications," in *Proceedings of the joint computer conference*. ACM, 1968, pp. 307–314.

[46] M. Kumar and D. S. Hirschberg, "An efficient implementation of Batcher's odd-even merge algorithm and its application in parallel sorting schemes," *IEEE Transactions on Computers*, vol. 100, no. 3, pp. 254–264, 1983.

[47] B. Razavi, *Design of analog CMOS integrated circuits*. McGraw-Hill, 2002.

[48] H. Kröll, S. Zwicky, B. Weber, C. Roth, C. Benkeser, A. Burg, and Q. Huang, "An Evolved EDGE PHY ASIC supporting soft-output equalization and rx diversity," in *Proc. of IEEE European Solid-State Circuits Conf. (ESSCIRC)*, Sept 2014, pp. 203–206.

[49] T. Dellsperger, D. Tschopp, J. Rogin, Y. Chen, T. Burger, and Q. Huang, "A quad-band class-39 RF CMOS receiver for Evolved EDGE," in *IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2010, pp. 56–57.

[50] M. Krueger, R. Denk, and B. Yang, "The training sequence code dependence of EDGE receivers using zero IF sampling," *IEEE Trans. on Wireless Communications*, vol. 5, no. 2, pp. 274–279, 2006.

[51] B. Weber, H. Kröll, C. Benkeser, and Q. Huang, "An efficient incremental redundancy implementation for 2.75G Evolved EDGE," in *SDR'13-WinnComm-Europe*, 2013.

[52] C. Benkeser, C. Roth, and Q. Huang, "Turbo decoder design for high code rates," in *Proc. IEEE/IFIP VLSI-SoC*. IEEE, 2012, pp. 71–75.

[53] *AMBA 3 APB protocol specification v1.0*, ARM Limited, 2004.

[54] *DigRF baseband/RF digital interface specification, version 1.12*, MIPS, TTPCom, Feb. 2004.

[55] T. Dellsperger, "Reconfigurability of RF receivers for evolved cellular communications," Ph.D. dissertation, ETH Zürich, Switzerland, 2010.

[56] I. Lu, C.-y. Yu, Y.-h. Chen, L.-c. Cho, C.-h. E. Sun, C.-C. Tang, and G. Chien, "A saw-less gsm/gprs/edge receiver embedded in a 65nm cmos soc," in *IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*. IEEE, 2011, pp. 364–366.

[57] A. Cicalini, S. Aniruddhan, R. Apte, F. Bossu, O. Choksi, D. Filipovic, K. Godbole, T.-P. Hung, C. Komninakis, D. Maldonado *et al.*, "A 65nm CMOS SoC with embedded HSDPA/EDGE transceiver, digital baseband and multimedia processor," in *IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2011, pp. 368–370.

[58] R. Koetter, A. C. Singer, and M. Tuchler, "Turbo equalization," *Signal Processing Magazine, IEEE*, vol. 21, no. 1, pp. 67–80, 2004.

[59] C. Studer, "Iterative MIMO decoding: Algorithms and VLSI implementation aspects," Ph.D. dissertation, ETH Zürich, Switzerland, 2009.

Curriculum Vitae

Stefan Altorfer-Zwicky was born in Baden, Switzerland, in 1981. He received the MSc degree in Information Technology and Electrical Engineering in 2007 from the ETH Zurich. He then joined Celestrius AG, an ETH-spinoff in the field of MIMO wireless communication, where he worked in the digital ASIC development. Subsequently, he worked one year for Enclustra GmbH as an FPGA design engineer. Since 2010, he has been a research and teaching assistant at the Integrated Systems Laboratory (IIS) at ETH Zurich, working on digital signal processing solutions for the physical layers of recent communication standards. His research interests include the design of digital VLSI circuits for wireless communications and the development of system prototypes.