

On the Mapping of Incremental Redundancy into a Physical Layer ASIC

Benjamin Weber · Harald Kröll · Christian Benkeser · Qiuting Huang

Received: 18 October 2013 / Revised: 31 July 2014 / Accepted: 21 August 2014 / Published online: 25 October 2014
© Springer Science+Business Media New York 2014

Abstract Incremental Redundancy (IR) was introduced in GSM/EDGE and later adopted in Evolved EDGE in order to keep the throughput at an acceptable level. Legacy 2G networks with their ubiquitous coverage are apt to provide a fallback solution for the latest LTE networks. On the other hand, they provide a reliable data link for emerging M2M or IoT applications. IR data processing and controlling is specified in several layers of the GSM/EDGE protocol stack. In many state-of-the-art designs this leads to decentralized data processing, and thus to costly data movements between memories and hardware accelerators. This is prohibitive for emerging M2M designs, which demand lowest possible hardware resources. In this work, IR is implemented as part of a dedicated hardware baseband signal processing unit in order to unburden higher layer processing units from IR, reduce area, power consumption, and costs. The open source baseband framework MatPHY is extended for packet switched operation to facilitate the development of efficient IR hardware architectures. With the design parameters obtained from performance evaluations computed with MatPHY an IR architecture in a 130 nm CMOS technology

is presented. Said architecture proves the suitability of an ASIC IR unit instead of a software solution distributed over various layers and components.

Keywords GSM · EDGE · Evolved EDGE · Incremental Redundancy · MatPHY · M2M · IoT

Acronyms

2G	2nd Generation
3G	3rd Generation
3GPP	3rd Generation Partnership Project
ACK	Acknowledgment
ARQ	Automatic Repeat Request
ASIC	Application Specific Integrated Circuit
AWGN	Additive White Gaussian Noise
BLER	Block Error Rate
BSN	Block Sequence Number
BTS	Base Transceiver Station
CMOS	Complementary Metal-Oxide-Semiconductor
CPU	Central Processing Unit
DEC	Decoder
DET	Detector
DFE	Digital Front-End
DSP	Digital Signal Processor
EDGE	Enhanced Data Rates for GSM Evolution
FSM	Finite State Machine
GE	Gate Equivalent
GMSK	Gaussian Minimum Shift Keying
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HDL	Hardware Description Language

B. Weber (✉) · H. Kröll · C. Benkeser · Q. Huang
ETH Zurich, Integrated Systems Laboratory, Gloriastrasse 35,
8003 Zurich, Switzerland
e-mail: weberbe@iis.ee.ethz.ch

H. Kröll
e-mail: kroell@iis.ee.ethz.ch

C. Benkeser
e-mail: benkeser@iis.ee.ethz.ch

Q. Huang
e-mail: huang@iis.ee.ethz.ch

IoT	Internet-of-Things
IQ	In-Phase and Quadrature
IR	Incremental Redundancy
LLC	Logical Link Control
LTE	Long Term Evolution
M2M	Machine-to-Machine
MAC	Medium Access Control
MCS	Modulation and Coding Scheme
MS	Mobile Station
NACK	Negative-Acknowledgment
NF	Noise Figure
PAN	Piggy-Backed ACK/NACK
PDU	Protocol Data Unit
PHY	Physical Layer
PSK	Phase-Shift Keying
QAM	Quadrature Amplitude Modulation
RAM	Random Access Memory
RLC	Radio Link Control
RF	Radio Frequency
ROM	Read-Only Memory
SNR	Signal-to-Noise Ratio
SoC	System-on-Chip
SPB	Split Block
TBF	Temporary Block Flow
TFI	Temporary Flow Identity
TPU	Time Processing Unit
TSC	Training Sequence Code

1 Introduction

After its first release in 1990, GSM was fast becoming one of the most successful and widely deployed cellular standards worldwide. Although initially supporting circuit switched connections only, there was a need for packet switched operation. Correspondingly, GPRS and later EDGE (also referred to as EGPRS) services were introduced by the 3GPP standardization organization. Together with EDGE, additional MCS were introduced to maximize raw bit rate and throughput. Irrespective of 3G and LTE deployments, the GSM standard and its enhancements will be in use for many years to come. Bearing this in mind, the 3GPP introduced an enhancement to EDGE called Evolved EDGE (also referred to as EGPRS2) with up to 32-QAM modulation and turbo codes. Hereafter the term throughput refers to payload bits per second between RLC and LLC layers. In order to keep the throughput even under severe radio conditions at an acceptable level, IR operations were first introduced with EDGE and inherited by Evolved EDGE. IR is also referred to as Type II Hybrid ARQ, as retransmissions occur automatically as part of the protocol. If decoding of an RLC block proves unsuccessful, the encoded

soft values (log-likelihood ratios) of the data block in question are kept at the receiver. Eventually, the data block gets retransmitted with additional redundancy information. The receiver combines the retransmission with the previous transmission before a decoding attempt is started. Correspondingly, the coding rate decreases with each retransmission whereas the decoding success rate increases. Thus, IR aids to achieve high average throughput. In the worst case an MS is asked to store up to 1024 RLC blocks [1–3]. Here, a maximum of 768¹ is considered. In both cases, memory and CPU challenges are two major concerns when implementing IR at MS side.

A hardware architecture of an MS consists of several processing components attached to an RF transceiver. The system CPU for higher layer operations is connected to memories and other peripherals. For digital baseband processing in the PHY a DSP with a number of accelerators is employed. The latest publications on GSM/EDGE transceivers tend towards SoC implementations including RF, digital baseband, application CPU, and peripherals [5, 6]. Commercially available products (e.g. [7]) use the same principle with highly multi-functional SoCs. However, these approaches have in common that the GSM/EDGE modem uses DSP and ARM architectures for digital signal processing. At times, small dedicated accelerators such as in [8] offload computationally intensive tasks such as equalization or channel decoding from the DSP or ARM CPU. However, power consumption is high. Indeed [6] claims to draw 232 mW and [5] 310 mW in transmit mode. In receive mode [5] gets as low as 176 mW. Alternatively, there exist approaches (e.g. [9]) which do not use a DSP. Instead, the digital baseband processing is implemented in dedicated hardware only. According to literature (e.g. [8]), IR implementation is traditionally distributed over various processing components in an MS. Erroneous RLC blocks are stored in an external RAM. IR control processes run on the system CPU. The decoding of RLC blocks takes place in the digital baseband on a DSP's accelerator.

Inherently, in such a conventional setup the IR mechanism claims high demands for the system CPU. And higher layer software developers need to incorporate IR operations. Furthermore, the data bus between external memory, system CPU, and digital baseband can experience high loads due to IR processing. Naturally, with loose constraints on power consumption, processing load, or circuit board area, this does not impose huge engineering challenges when it comes to implementing an Evolved EDGE data modem. However, this is not always the case.

¹768 RLC blocks is the maximum for a class 1 MS which can (if implemented) receive on two carriers. 1024 RLC blocks corresponds to a class 2 MS. However, the latter must have the capability of simultaneous reception and transmission but dual-carrier reception is not allowed. Therefore, a class 1 MS has been considered here [1–4].

Emerging applications such as IoT or M2M communication may use 2G GPRS and successors due to its ubiquitous coverage. The mentioned applications share a number of requirements with respect to the data modem:

- *Low-power*: Unlike other battery powered mobile platforms, such as smart phones, life times in the range of weeks, months, or even years are required without the need of human interaction. And in case of zero-power systems ultra-low power data modems are required.
- *Low-area*: Any reduction in area, be it on a silicon die or a circuit board, allows for smaller and cheaper devices.
- *Low-cost*: Even though the two previously mentioned requirements reduce costs already, further reductions are required. This can be achieved by e.g. using highly integrated systems and/or with a design which requires less components on a circuit board.

Contribution: In this paper IR implementation challenges are investigated and a dedicated hardware architecture, which hides IR operations from the system CPU and higher layers, is proposed. An IR processing block incorporated into the digital baseband comprising all IR related operations can remove IR load from the system CPU and external components. Such an architecture is apt for use in previously mentioned applications, such as IoT or M2M communication, where low-power and low-area are indispensable. In order to gain insight into IR mechanisms, an Evolved EDGE capable version of the open source MatPHY GSM framework [10] was equipped with IR functionality. The minimum IR memory size in order to meet the 3GPP throughput requirements is determined based on the receiver performance. The implemented IR hardware architecture using a dedicated on-chip memory for the storage of RLC blocks comprises depuncturing, memory management, and a control unit. A HDL model is implemented and synthesized using a 130 nm CMOS technology. The latter is apt for ASIC implementations such as [11].

Outline: The remainder of this paper is organized as follows. Section 2 places IR operation in the context of the GSM/EDGE protocol stack. Subsequently, Section 3 introduces relevant details about Evolved EDGE. Then, Sections 4 and 5 introduce the simulation framework used and corresponding results are presented. Section 6 presents a dedicated IR hardware implementation and thereafter Section 7 concludes the paper.

2 GSM/EDGE Protocol Stack

A GSM/EDGE packet switched connection is referred to as TBF. GSM/EDGE protocol layers used in the packet switching technology consist of the PHY, MAC, RLC, and LLC

layers (see Fig. 1). However, only layers in use during an active TBF are shown. Layers above the LLC are omitted, as well.

In the following it is assumed that a TBF has been established.

- The *LLC* layer provides LLC PDUs corresponding to the TBF to the RLC layer.
- The *RLC* layer provides a logical link between an MS and a BTS. It splits LLC PDUs into RLC blocks and reassembles LLC PDUs in case of data reception. A flow control mechanism is used in terms of a transmit window on an RLC block granularity. It can operate in acknowledged or unacknowledged mode. In the former case, retransmissions of RLC blocks are possible.
- Multiplexing and mapping of data to logical channels is a task of the *MAC* layer. Logical channels transport so called radio blocks consisting of an RLC/MAC header and RLC blocks as payload. It concatenates RLC blocks and prepends an RLC/MAC header.
- The *PHY* encodes the header and each RLC block of a radio block separately. A rate adaption mechanism omits certain bit positions in order to meet the amount of available bits on the air interface according to the MCS used. This rate adaption is also referred to as puncturing. The encoded header and RLC blocks are subsequently mapped onto 4 radio bursts in 4 subsequent frames on the air interface.

RLC and MAC functions can be summarized as a single RLC/MAC layer. In case of various concurrent TBFs in the same device multiple instances of some layers run in parallel [2].

2.1 IR and Protocol Stack

A 25 bit identification per encoded and punctured RLC block is sufficient at the MS for the RLC block's unique identification. It consists of the following information within the RLC/MAC header [3]:

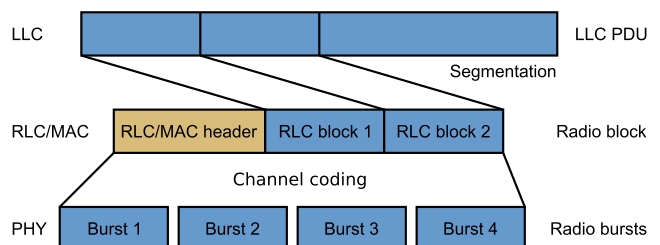


Figure 1 LLC PDU segmentation by RLC/MAC layer down to the PHY in the GSM/EDGE protocol stack [3, 12].

- The *TFI* is a 5 bit number which corresponds to an active TBF. It gets assigned to the MS by the BTS during the establishment of the TBF. The TFI in the header is the same for all RLC blocks in the payload.
- Each RLC block within a TBF is numbered with a *BSN*. This 11 bit value is used for e.g. the RLC flow control mechanism.
- A 5 bit *MCS* value is present as a coded version in the header. The coding scheme is the same for all RLC blocks in the payload.
- A 2 bit *puncturing scheme* value of each RLC block within the radio block is present as a coded version as well.
- As retransmissions of RLC blocks can not only use a different puncturing scheme but also a different MCS it might not always fit into the radio block containing the retransmission. A 2 bit *SPB* value indicates whether a retransmitted RLC block has been split and which split segment of the RLC block is present in the payload.

IR operations are performed on an RLC block basis and are only mandatory in the downlink (from BTS to MS). IR controlling (IR memory management) forms part of the RLC/MAC layer whereas IR processing (combination of redundancy versions and subsequent decoding) is a PHY operation. The MS PHY receives a radio block and makes an attempt to decode the RLC/MAC header and RLC blocks. In RLC acknowledged mode, the MS RLC/MAC layer stores the encoded RLC block's soft values and its 25 bit identification and sends a NACK message towards the BTS in case of erroneous decoding. This automatically invokes a retransmission of the RLC block in question. Should decoding succeed an ACK message is sent towards the BTS. The BTS RLC/MAC layer makes sure that the PHY uses a different puncturing pattern for the retransmission. The MS RLC/MAC layer knows when the retransmitted RLC block is received by analyzing the RLC/MAC header of each received radio block. Whenever the 25 bit identification matches, it assures that the stored RLC block gets combined with the retransmitted version before decoding. In this manner, more redundancy bits are available at the MS after each retransmission. This procedure can be repeated various times until decoding succeeds or another termination criteria is met [2].

2.2 Classical GSM/EDGE Architecture

A GSM/EDGE MS architecture is typically based on two processing units and an RF transceiver. In Fig. 2 the architecture of [8] is given. It comprises a DSP for baseband processing and a system CPU for higher layer handling. Both processors have access to RAM and ROM modules. Furthermore, the system CPU has external components attached to

it such as memory and peripherals. The DSP uses accelerator blocks for computationally intensive operations such as channel equalization or decoding. Alternatively, it is possible to employ only dedicated hardware for the PHY such as in [9].

IR operations are logically spread over RLC/MAC and PHY layers as RLC/MAC is responsible for controlling and the PHY for rate adaption and decoding. Physically, they are spread over a DSP, system CPU, and external RAM for storing soft value information. A large amount of data transfers between PHY (DSP), RLC/MAC (CPU), and IR memory (external RAM) is the result.

2.3 Dedicated Hardware Acceleration

In [8] an attempt was made to minimize IR data moves by relieving the RLC/MAC layer from IR controlling operations. In fact, separate IR controlling and processing blocks were employed. They run partly on the system CPU as well as in dedicated hardware as depicted in Fig. 2. The RLC/MAC layer still needs to know whether decoding of an RLC block failed in order to produce corresponding ACK/NACK messages for the network. However, awareness of IR operations on an RLC/MAC layer level is no longer required as long as the IR controlling module is informed whether the TBF is operating in acknowledged mode. Still, a lot of data moves across layers remain. Across layer data moves are between IR memory (external RAM), IR controlling and processing modules (CPU and dedicated hardware), and the PHY (where the equalizer resides).

Moving all IR related operations into dedicated hardware without external components allows further reduction of IR processing overhead in the system CPU. Correspondingly, the system CPU is unburdened from these tasks and a smaller and more economical (with respect to power consumption and costs) CPU can be deployed instead. We focus on PHYs as presented in [9] which allow minimization of data moves between various parts of a GSM/EDGE receiver with respect to IR. Moreover, IR operations are completely invisible for the system CPU. The vision of moving all IR related operations into a dedicated module in PHY is depicted in Fig. 2.

However, moving IR controlling and processing into the baseband inherently requires control data moves between RLC/MAC layer and the PHY. More detailed, a list of TFIs in acknowledged mode currently assigned to the MS and their corresponding transmit window sizes are required in the PHY. Naturally, the transmit window size could simply be replaced with a hard coded value corresponding to the maximum transmit window size for the multislot class (maximal timeslot configuration) supported (see [3]). Technically, status information for each BSN within the current transmit window per TBF is required for IR controlling

Table 1 Data moves for IR operations.

Variant	As in [8]	Dedicated PHY
Across layers	s_p	0
Off-chip memory read	$s_p n_v (s_d)$	0
Off-chip memory write	$s_p (s_d)$	0
On-chip memory read	0	$s_p n_v (s_d)$
On-chip memory write	0	$s_p (s_d)$

3.1 IR and Evolved EDGE

IR procedures as introduced with EDGE are mapped with-out changes to Evolved EDGE. This means that the handling of retransmissions and combining redundancy versions is not altered. However, Evolved EDGE level A DAS12 MCS (with 32-QAM and turbo coding) achieves a throughput of 98.7 kbit/s/timeslot as opposed to 59.4 kbit/s/timeslot EDGE. This 66 % increase in throughput forces tremendous requirements with respect to IR memory size and IR controlling [13].

3.2 Evolved EDGE Hardware Support

So far, only test equipment and academic implementations with support for Evolved EDGE exist. There exists no commercial deployment in networks or hand-set chips. Fortunately, the hardware vision as depicted in Fig. 2 can be applied to EDGE Evolution, as well. The focus on IR in PHY alone rises the question of determining a reasonable memory size for storing soft values as an on-chip memory shall be used. The 3GPP specifications dictate test cases with a minimum throughput requirement given that IR is enabled [1]. Two approaches exist in order to fulfill the test case requirements:

1. A memory with a capacity of storing all possible RLC block redundancy versions within the largest possible transmit window. Clearly, the advantage of this approach lies in the fact that the memory size can immediately be determined.

Assuming the highest Evolved EDGE level A MCS DAS12 (3 redundancy versions, 700 bit per punctured RLC block) and a window size of 768^3 corresponding to the highest multislot class the IR memory needs a capacity of [1–4, 12]

$$3 \times 700 \times 768 \approx 1,6 \text{ M soft values.} \quad (1)$$

2. A distinct approach is to perform simulations incorporating IR test case specifications from 3GPP with the IR memory size as parameter. Unfortunately, these simulations require not only IR and baseband operations but the entire receive chain including the RF part to

be taken into account. Furthermore, RLC/MAC procedures ensuring that the transmit windows is not violated need to be taken into account, as well.

Naturally, an on-chip memory of 1.6 M soft values capacity is not feasible for low-cost and low-power applications as mentioned in the introduction. Therefore, simulations need to be performed in order to determine the minimum required memory size. In spite of the complexity of such a simulation setup, the advantage of minimizing IR memory and thus also minimizing silicon die size and power consumption dominate. Keeping these savings and simplifications in mind a dedicated IR implementation inside the PHY is developed starting from an IR extension for the MatPHY simulation and development framework.

4 Simulation Setup

IR performance simulations have been split into two parts, one that uses parallelization and a sequential one. The former is done with an extended version of MatPHY, an open source PHY development framework [10]. It outputs BLER p_i for $i = 0, 1, 2, \dots$ denoting the amount of transmissions already stored in IR memory. The BLER results are then used in the second simulation step to evaluate IR memory size requirements. This second step takes into account RLC/MAC behavior.

First, The IR test case is presented. Subsequently, the two steps of the simulation setup are explained.

4.1 IR Test Case

The required memory capacity for storing soft value information required for IR processes is not given in the 3GPP specifications. However, a long-term throughput per timeslot needs to be achieved [1].

Table 2 summarizes the test conditions under which the overall IR performance needs to be evaluated. Values such as acknowledgment polling period and transmit window size and their impact on the IR test case are described in [2]. For these simulations, the overall receiver needs to be taken

³See note 1 on page 2

Table 2 IR test case description [1].

	EDGE	Evolved EDGE
Required throughput	20 kbit/s/timeslot	33 kbit/s/timeslot
Propagation conditions	AWGN, input level -97 dBm	AWGN, input level -94 dBm
MCS	MCS9	DAS12
ACK polling period	32 RLC blocks	32 RLC blocks
Roundtrip time	120 ms	120 ms
Number of timeslots	Maximum capability of the MS	Maximum capability of the MS
Transmit window size	Maximum for the MS capability	Maximum for the MS capability

into account. Finite word widths for all data operations are required in order to mimic hardware behavior. Bit-true models of hardware arithmetic operations need to be used. Furthermore, the NF of the RF part must be counted. RLC/MAC operations need to be simulated as well. These take into account acknowledgment polling and prevent violation of the transmit window. As only the downlink is specified in the test case the uplink can be considered ideal⁴.

4.2 Simulation Step 1: MatPHY Extended Towards EDGE Evolution

The open source project MatPHY presented in [10] uses OsmocomBB (an open source GSM protocol stack [14]) and a custom PHY called *phydev* written in Matlab (see Fig. 3). The OsmocomBB distinct L1CTL protocol is used for data exchange between PHY and higher layers. MatPHY models digital baseband operations and has been structured into PHY controller, TPU, auxiliaries, and three large signal processing blocks which there are DFE, DET, and DEC. The signal processing blocks consist of controllers and various signal processing primitives. Each signal processing primitive can be replaced with an arbitrary version. In fact, signal processing primitives can be replaced by their bit-true representations thus mimicking an actual hardware implementation. Moreover, Matlab bit-true models are available for simulation and MatPHY, already being written in Matlab, was therefore an excellent candidate for the evaluation of hardware behavior.

Following the EDGE evolution MatPHY has been expanded into three directions:

- Support for packet switched data (GPRS, EDGE, Evolved EDGE level A).
- Enhancement for the OsmocomBB L1CTL protocol.
- Configurable data source for on-the-fly IQ samples generation (BTS and radio channel).

⁴DAS12 downlink uses 32-QAM modulation, MCS9 uses 8-PSK, and ACK/NACKs in the uplink use GMSK modulation. Even if one would use the same channel conditions for the uplink and the same receiver performance as the MS (BTS has more stringent requirements [1]) the BLER in the uplink direction is 0.

- Minimalistic RLC/MAC software.

No extensive model of the RF part is required, though. When converting from receive power as in Table 2 to SNR the NF can be considered as an additional noise source.

Due to the high computational load of the performance simulations, parallelization of simulation sets, each processing a small amount of RLC blocks, was performed.

4.2.1 Packet Switching Technology Support

Enabling packet switching technology support is a twofold effort.

First, as OsmocomBB software does not support packet switched communication, a minimalistic RLC/MAC test-software written in Matlab is required. Depending on the decoding results it informs the data source to retransmit the same RLC blocks with a different puncturing scheme or to generate new RLC data for the receiver. The latter can be used in an interactive manner to test MatPHY packet switched operations. The original MatPHY presented in [10] has a memory mapped file with a handshake protocol as interface between *phydev* and higher layers. In order to reuse the same interface in *phydev*, the minimalistic RLC/MAC software implements the same interface. This interface is also used for communication with the configurable data source script explained in Section 4.2.3.

Second, the PHY part of MatPHY has been enhanced for packet switched operations. This includes additional controller functionality in the PHY controller. In addition, DET and DEC were enhanced accordingly. Most notable is the required support for additional modulation schemes in DET. To this end, the SOVE algorithm as presented in [11] has been used. For an incorporation of a turbo decoder for Evolved EDGE operation the solution of [15] was taken. Naturally, so far unused primitives in DEC such as depuncturing and straight forward IR with infinite memory needed to be added. Soft values quantized to a finite width w bit have been used in the DET and DEC blocks. Simulation results have shown that $w = 5$ bit results in a negligible performance degradation compared to $w > 5$ bit. The DARP requirements [1] can be achieved with our implementation.

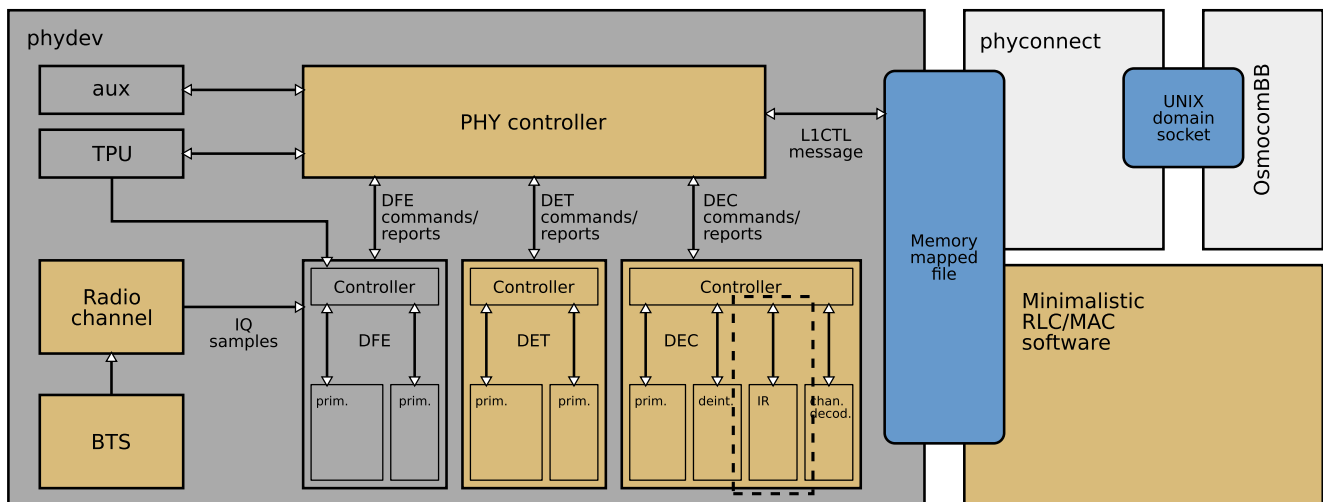


Figure 3 MatPHY Evolved EDGE extensions: Data source (BTS, radio channel), minimalistic RLC/MAC software, enhancements in DET and DEC. A zoom of the rimmed selection in DEC can be seen in Fig. 7.

Packet switched operation requires blind detection of the modulation order inside DFE [2]. However, this task was omitted and a priori knowledge of the modulation order in the receiver was assumed. Naturally, this assumption is not generally applicable. As only the IR test case is of interest, this holds, because it defines already which MCS to use and therefore which modulation order to expect (see Section 4.1).

4.2.2 L1CTL Enhancement

Even though the L1CTL protocol of the OsmocomBB project does not support packet switched operation, it can be easily extended. In fact, only two additional L1CTL request messages are required:

TBF_REQ: This message holds a list of TFIs corresponding to TBFs for which the MS is currently configured. This is necessary as resource sharing is possible on the down-link [16]. As described in Section 2 this is a 32 bit value. The MS may distinguish payloads intended for decoding by analyzing the TFI field in the decoded header. In addition, this message indicates whether a TBF corresponding to a certain TFI is in RLC acknowledged mode or, in other words, whether IR operations are required.

DATA_CONF_REQ: This message holds an information structure for each timeslot containing the TSC and the GSM mode (GPRS, EDGE, or Evolved EDGE). In addition, it indicates whether in a certain timeslot transmission or reception is required. Furthermore, a starting frame number indicating the first valid frame of the new configuration forms part of this message. Naturally, this message can be enhanced with additional information as the need arises.

From PHY towards RLC/MAC a total of two additional L1CTL messages were defined:

TBF_CONF: This is a simple confirmation of the corresponding request.

P_TRAFFIC_IND: This message is used to indicate and report data on packet switched channels. It can be used to report RLC/MAC headers, RLC blocks, and control information as soon as it is decoded.

4.2.3 Configurable Data Source

The data source consists of a BTS and a radio channel for on-the-fly IQ samples generation. It can be configured in an interactive manner. The BTS can produce burst wise IQ samples for decoding. In addition, it holds state information needed for RLC block retransmissions. The radio channel is able to modify the transmit burst according to 3GPP channel profiles. As soon as a radio block has been reported to the minimalistic MS RLC/MAC software, the data source can create retransmissions (if necessary) by encoding and puncturing the same RLC block using a different puncturing scheme. Naturally, the TFI and BSN information within the RLC/MAC header needs to be correctly adapted in the BTS. Hence, primitive IR simulations without actually modeling complete RLC and MAC layers can be enabled.

4.3 Simulation Step 2: RLC/MAC Layer Simulation

As MatPHY is a PHY simulation framework, a separate simulation has been set up using the BLER results from MatPHY. A separate RLC block level simulation flow implementing a BTS and an MS was set up, see Fig. 4. It comprises the various requirements from Table 2. In

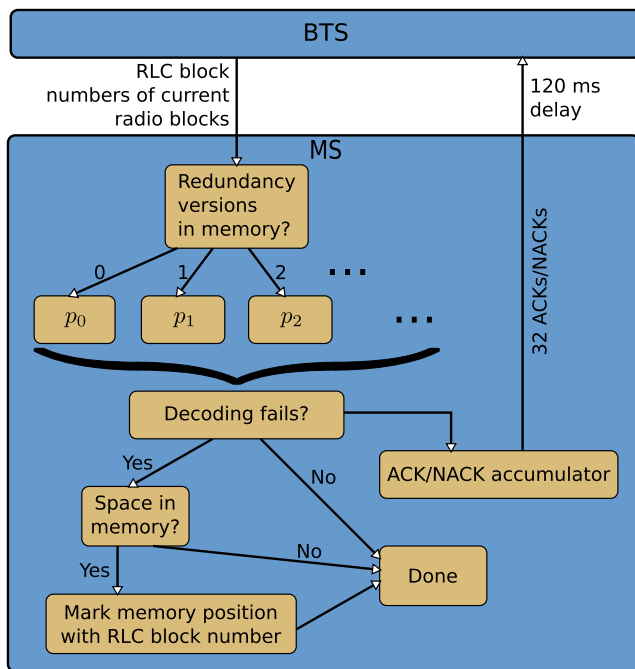


Figure 4 IR flow setup used in the simulation.

addition, the MS can be configured to various IR memory sizes in terms of punctured RLC blocks. The BTS decides using RLC/MAC procedures which RLC blocks need to be transmitted. The MS evaluates whether the decoding fails using the BLER results from the previous section. They depend only on the channel conditions and the presence of previous transmissions in IR memory. Therefore, the BLER still hold even if complete RLC/MAC behavior is simulated. For various IR memory sizes the average throughput per timeslot can be computed. Six downlink timeslots were used according to multislot class 45 as specified in [3]. As no IQ data is equalized or decoded in the RLC block level simulation, computation time is less critical and parallelization is not required.

5 Simulation Results

Performance was assessed using the MatPHY extension from the previous section for different modulation and coding schemes using static channel conditions. The 3GPP specifies the static channel as an AWGN channel without fading and without multi-path propagation [1]. The BLER is evaluated. Subsequently, the second simulation step is used to determine an explicit IR memory size.

5.1 BLER

RLC BLER for MCS9 and DAS12 with IR enabled are plotted in Fig. 5. The BLER after the first transmission (p_0)

corresponds to the case without IR. The BLER after the first retransmission (p_1) considering only those RLC blocks where at least one retransmission was required is rendered as well. In other words, this is the BLER given that exactly one redundancy version is in IR memory. By the same token, BLERs after more than one retransmission are depicted and so on. After each retransmission the BLER diminishes. It should be noted that the performance gain from the second to the third retransmission is lower than the gain from the first to the second retransmission. Both MCS9 and DAS12 RLC blocks can be punctured in 3 different manners. Consequently, the third retransmission is in fact a retransmission of the initially transmitted bits, thus bringing less gain than newly transmitted bits.

5.2 Throughput and Memory Size

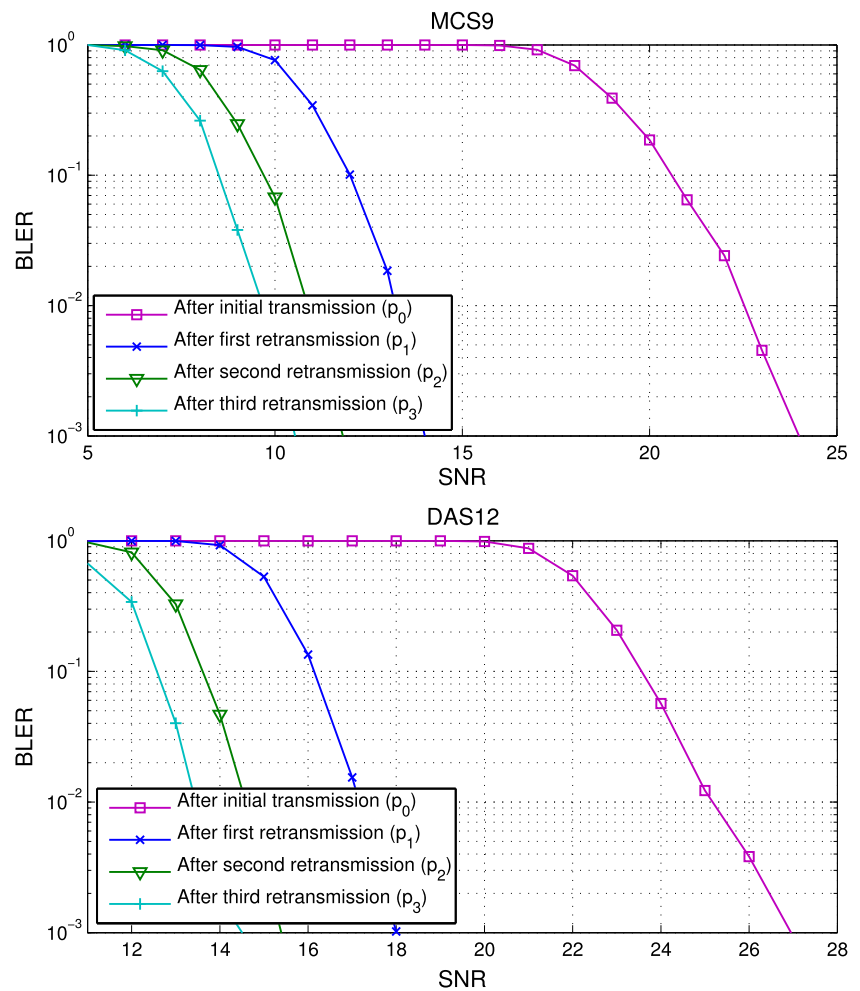
It has been found that under the IR test case conditions at most 1 retransmission for both EDGE MCS9 and Evolved EDGE DAS12 is required. Hence, only p_0 and p_1 from the BLER simulations with MatPHY are relevant as $p_i = 0, \forall i = 2, 3, 4, \dots$ Figure 6 shows throughput simulation results for MCS9 and DAS12, respectively. As required by the 3GPP IR performance requirements the throughput per timeslot is plotted on the vertical axis against the IR memory capacity in terms of punctured RLC blocks. Two receiver performances have been simulated. In fact, the RF performance model in terms of NF was altered whereas the digital baseband performance was not modified.

The target throughput for MCS9 with an NF of 6 dB can be met with almost no IR memory. However, if an NF of 8 dB is used roughly 40 punctured RLC blocks need to be stored. A similar increase of the required IR memory capacity can be observed using the same two NF and the Evolved EDGE DAS12 test case. This clearly shows that in order to evaluate IR performance and to find a suitable IR memory size the entire PHY processing chain needs to be taken into account, including the RF transceiver. The overall receiver performance has a huge impact on IR performance. Unfortunately, there exists no analytical expression for the memory size. Only simulations determine the throughput taking into account the entire receiver performance.

A punctured MCS9 RLC block consists of 612 bit and a punctured DAS12 RLC block consists of 700 bit [12]. Put into more concrete numbers the following memory requirements have been found:

- *NF 6 dB*: 4 punctured MCS9 RLC blocks and 39 punctured DAS12 RLC blocks, which corresponds to $39 \times 700 \times 5 \text{ bit} = 136.5 \text{ kbit}$.
- *NF 8 dB*: 39 punctured MCS9 RLC blocks and 54 punctured DAS12 RLC blocks, which corresponds to $54 \times 700 \times 5 \text{ bit} = 189 \text{ kbit}$.

Figure 5 MCS9 (*top*) and DAS12 (*bottom*) RLC BLER against receive SNR for an AWGN channel.



As stated in subsection 3.2 a memory capacity of 1.6 M soft values or 8 Mbit⁵ would in any case be sufficient. For both the NF 6 and NF 8 cases a tremendous memory size reduction could be achieved compared to the 8 Mbit without any simulations.

5.3 Diminish Memory Requirements

A specific memory size in terms of punctured RLC block soft values has been found. There exist a number of methods which store the data differently, some of them are candidates to further reduce IR memory size. An option is to use data compression on the soft values before storing, thus possibly reducing the required memory size. A loss-less compression algorithm would indeed reduce the required memory without altering the throughput. However, loss-less data compression can be complex and computationally demanding and has, therefore, not been pursued.

In contrast, lossy compression by simply truncating the soft values before storage has been analyzed as it requires

absolutely no additional computation. The BLER simulations from the previous section have been redone using the same receiver performance but with soft value truncation from 5 to as low as 1 bit. The latter actually corresponds to storing hard decisions in IR memory. When extracting the hard decisions from IR memory they were mapped to the largest soft values with positive or negative sign,

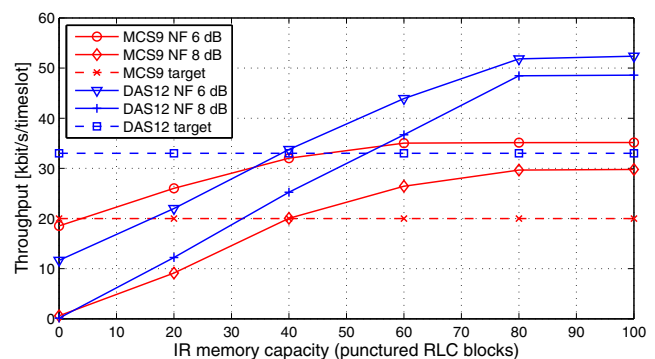


Figure 6 Simulation results in terms of throughput per timeslot against punctured RLC block memory capacity.

⁵1.6 M soft values with a width of 5 bit.

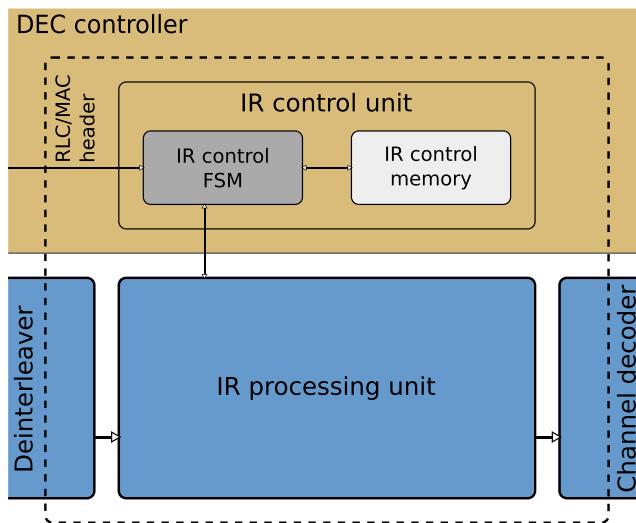


Figure 7 Top level block diagram of the proposed IR architecture. This corresponds to the shaded area inside DEC of Fig. 3.

respectively. Subsequent RLC/MAC layer simulations have shown, that storing hard decisions instead of soft values do not alter the throughput under the IR test case conditions, irrespective of the memory size. Correspondingly, the explicit memory sizes from the previous section can be divided by 5 which leads to⁶:

- *NF 6 dB*: 4 punctured MCS9 RLC blocks and 39 punctured DAS12 RLC blocks, which corresponds to $39 \times 700 \times 1 \text{ bit} = 27.3 \text{ kbit}$.

Compared to the roughly 8 Mbit memory using 5 bit soft values and no simulations only 3.4 % of the 8 Mbit are required. This reduction is significant.

6 Dedicated Hardware Architecture

The simulation framework described in the previous section was used to determine the memory size for storing hard decisions for IR operations. Such an explicit memory size is required whenever dedicated on-chip memory modules shall be used. This can be the case for low-power M2M platforms with a low component count. In order to check the feasibility of having IR operations in PHY alone a dedicated hardware architecture was developed.

6.1 Architecture

An enlarged version of the shaded section of Fig. 3 is depicted in Fig. 7. It consists of an IR control unit as part

of the DEC controller and an IR processing unit, the latter a signal processing primitive between deinterleaver and channel decoder.

The IR control unit is responsible for all IR operations which do not concern the information bits of RLC blocks. In order to perform IR operations the header bits containing the MCS, TFI, BSNs, and so on need to be analyzed. While processing a radio block the DEC controller keeps a copy of a portion of the decoded header bits. In order to enable IR operations in PHY it makes sense to simply store the entire header bits in the DEC controller. Consequently, it is feasible to have an IR control unit inside the DEC controller accessing the decoded header bits. All IR operations become thus completely transparent to RLC/MAC processes. And, what's more, IR control is an autonomous system inside PHY. The IR control unit has a small memory containing a list of information sets regarding the stored RLC blocks in IR memory. Each information set contains the 25 bit identification from Section 2.1 as well as the position in IR memory. In order to ease memory access and management, IR memory is fragmented into memory blocks. A punctured RLC block occupies one or more such memory blocks depending on its size. In addition, the IR control unit stores the current transmit window position of each active TFI. Whenever an RLC block is ready to be deinterleaved, the IR control unit checks whether there already exists a version with the same BSN and transfers the required information to the IR processing unit. In addition, if a newly received RLC block moves the transmit window

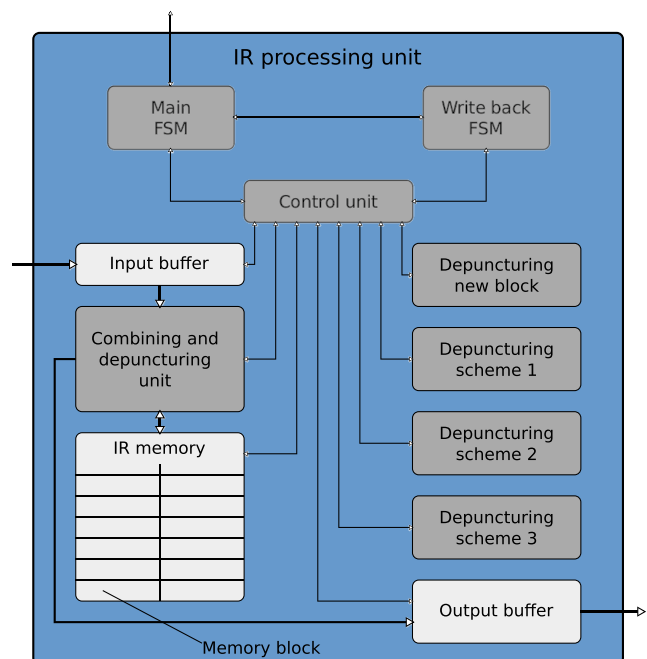


Figure 8 Block diagram of the dedicated IR processing unit including IR memory.

⁶The NF 8 case was not considered for data compression.

Table 3 Memories and their sizes for the implemented architecture.

Memory	Size (kbit)	Size (μm^2)
Control memory	2.067	25728
Input buffer	6.24	46424
Output buffer	10.11	66851
IR memory	27.3	133417
Total memory	45.717	272420

such that older RLC blocks drop out, the corresponding IR memory can be recycled. Once the decoding terminates, the IR control unit updates the control memory according to the result from the channel decoder. Should decoding fail, the corresponding control entries remain, otherwise they can be recycled.

The detailed IR processing unit is depicted in Fig. 8. Besides the actual IR memory it consists of two FSMs, an input buffer, an output buffer, and a control unit. Furthermore, it comprises four depuncturing units, 3 for existing RLC versions inside IR memory and an extra unit for the newly received RLC block. Even though there exist at most 3 different puncturing schemes per MCS the extra unit is indispensable as the newly received RLC block might have undergone additional PAN puncturing at the transmitter. The *Main* FSM is responsible for receiving soft values from the upstream unit into the input buffer, depuncturing and combining previous transmissions with the newly received soft values and writing them into the output buffer, and moving the results towards the downstream unit out of the output buffer. The *Write Back* FSM is responsible for transferring newly received soft values from the input buffer and/or the output buffer into the IR memory. Only hard decisions are saved. Newly received RLC blocks are written back to the IR memory without waiting for the result of the channel decoder or, in other words, as soon as the combined version is in the output buffer. Correspondingly, the IR unit can perform write back operations and process the next RLC block while the channel decoder is processing the previous RLC block. As soon as the Main FSM finishes combining the Write Back FSM starts with the write back operation. In the simplest case the latter comprises writing the input buffer content into the IR memory. However, in case the same punctured version is already present in the IR memory the two transmissions are combined before write back in order to save IR memory. What's more, if the newly received RLC block has undergone additional PAN puncturing conversion to regular puncturing is performed as, for the sake of simplicity, only regularly punctured RLC blocks are stored in IR memory. When the entire depunctured and combined RLC block has been transferred to the downstream unit the Main FSM stalls until the Write Back FSM is finished or the other way around, whichever is first.

This approach of writing back immediately brings the advantage that the IR processing unit is free to prepare the next RLC block while the previous block is being decoded. Alternatively, a dedicated write back memory could be used for temporary storage of RLC blocks of which it is not yet known whether decoding failed or will fail. The IR processing unit could still prepare the next RLC block for decoding. However, the additional memory is considered unpractical and this approach was therefore discarded.

6.2 Synthesis and Clocking

The architecture as described above was implemented in HDL using the same bit-true representations for data operations as in the MatPHY simulations. Thus, the performance of the MatPHY simulation coincide with the hardware implementation. This has been verified⁷ using Mentor Graphics ModelSim software suite. Subsequently, it was synthesized using Synopsys Design Compiler software with a 130 nm technology.

The input buffer must meet the largest punctured RLC block which consists of 1248 soft values whereas the output buffer must be able to store the largest depunctured RLC block which consists of 2022 soft values [12]. The memory capacity for the NF 6 case of 27.3 kbit was used. The memory sizes of input buffer, output buffer, IR memory, and control memory are listed in Table 3. The circuit can be clocked with a maximum clock frequency of $f_{c,\max} = 230$ MHz which is sufficient for the fastest modes of Evolved EDGE level A. The circuit (without memory) corresponds to 45k GE at $f_{c,\max}$ and 46 kbit on-chip memory. The dedicated digital baseband solution in [9] has a size of 184k GE and 40 kbit on-chip RAM and comprises equalization and channel decoding, but no IR processing. Therefore, combining the two solutions would lead to 230k GE and 86 kbit of memory or, in terms of mm^2 using a 130 nm technology, 2.5 mm^2 . The proposed IR architecture has been incorporated into the 130 nm CMOS digital baseband

⁷Verification of an ASIC implementation with respect to 3GPP throughput requirements has not been considered. Throughput compliance with IR enabled can only be tested under lab conditions with e.g. a protocol tester, where the required static channel conditions are not an issue.

processing ASIC presented in [11]. In the latter, the power consumption of the entire digital baseband can be as low as 5 mW.

7 Conclusions and Future Work

The open source MatPHY simulation and development framework has been extended with packet switched operation including Evolved EDGE and L1CTL extensions to connect with RLC/MAC software. Together with a higher layer simulation setup an explicit and extremely small IR memory size has been justified. EDGE and Evolved EDGE IR operations can be implemented in dedicated hardware as part of the PHY without burdening the system CPU. In fact, IR operations have been completely hidden from MAC and RLC layers.

The proposed IR architecture can be incorporated into a dedicated hardware baseband processing chip from previous work [9]. The digital baseband processing ASIC presented in [11] comprises the proposed IR architecture. The latter consumes in the best case not more than 5 mW of power as opposed to 176 mW of [5].

Of course, RF front-end and higher layer CPU have not been taken into account. In combination with an RF solution a single-chip complete PHY ASIC can be built. Power consumption and footprint area of such a solution remain to be analyzed. And, the incorporation of a higher layer CPU in order to achieve an ultra-low power SoC Evolved EDGE modem for the use in IoT and M2M applications remain to be analyzed, as well.

References

1. 3GPP TS 45.005 V11.4.0 (2014). Radio transmission and reception, January.
2. Seurre, E., Savelli, P., Pietri, P.J. (2003). *EDGE for mobile internet*. Artech House Publishers.
3. 3GPP TS 44.060 V11.8.0 (2014). General Packet Radio Service (GPRS); Mobile Station (MS) - Base Station System (BSS) interface; Radio Link Control/ Medium Access Control (RLC/MAC) protocol, March.
4. 3GPP TS 45.002 V11.3.0 (2013). Multiplexing and multiple access on the radio path, September.
5. Darabi, H., Chang, P., Jensen, H., Zolfaghari, A., Lettieri, P., Leete, J.C., Mohammadi, B., Chiu, J., Li, Q., Chen, S.-L., Zhou, Z., Vadipour, M., Chen, C., Chang, Y., Mirzaei, A., Yazdi, A., Nariman, M., Hadji-Abdolhamid, A., Chang, E., Zhao, B., Juan, K., Suri, P., Guan, C., Serrano, L., Leung, J., Shin, J., Kim, J., Tran, H., Kilcoyne, P., Vinh, H., Raith, E., Koscal, M., Hukkoo, A., Hayek, C., Rakhshani, V., Wilcoxson, C., Rofougaran, M., Rofougaran, A. (2011). A quad-band GSM/GPRS/EDGE SoC in 65 nm CMOS. *IEEE Journal of Solid-State Circuits*, 46(4), 870–882.
6. Wu, T.-H., Chang, H.-H., Chen, S.-F., Chiu, C.-S., Lai, L.-S., Wang, C.-H., Yang, S.-Y., Lin, T.-H., Chen, J.-R., Tsai, H.-C., Yu, C.-Y., Su, S.-Y., Yu, T.-Y., Chin, C.-C., Dehng, G.-K., Marques, A., Wang, C., Chien, G. (2013). A 65-nm GSM/GPRS/EDGE SoC with integrated BT/FM. *IEEE Journal of Solid-State Circuits*, 48(5), 1161–1173.
7. Texas Instrument (2014). TCS family of advanced GSM/GPRS chipset solutions. http://www.ti.com/corp/docs/webemail/ctiamar03/GSM_GPRS_021903.pdf.
8. Chang, L.F., & Wang, Y. (2009). EDGE incremental redundancy memory structure and memory management July 23. *US Patent App.*, 12/507, 835
9. Benkeser, C., Bubenhofer, A., Huang, Q. (2010). A 4.5 mW digital baseband receiver for level-A evolved EDGE. In *Solid-state circuits conference digest of technical papers (ISSCC), 2010 IEEE International* (pp. 276–277). IEEE.
10. Kröll, H., Zwicky, S., Weber, B., Benkeser, C., Huang, Q. (2013). Physical layer development framework for OsmocomBB. *Journal of Signal Processing Systems*, 73(3), 301–314.
11. Kröll, H., Zwicky, S., Weber, B., Roth, C., Benkeser, C., Burg, A., Huang, Q. (2014). An evolved EDGE PHY ASIC supporting soft-output equalization and RX diversity. In *IEEE 40th European solid-state circuits conference (ESSCIRC)*.
12. 3GPP TS 45.003 V11.1.0 (2013). Channel Coding, February.
13. Säily, M., Sébire, G., Riddington E. (2011). *GSM/EDGE - evolution and performance*.
14. OsmocomBB (2013). *An open source GSM baseband software implementation*. <http://bb.osmocom.org>.
15. Benkeser, C., Roth, C., Huang, Q. (2012). Turbo decoder design for high code rates. In *2012 IEEE/IFIP 20th international conference on VLSI and system-on-chip (VLSI-SoC)* (pp. 71–75).
16. 3GPP TS 43.064 V11.0.0. (2012). General Packet Radio Service (GPRS); Overall description of the GPRS radio interface; stage 2, October.



Benjamin Weber received his BSc and MSc in Electrical Engineering and Information Technology from the Swiss Federal Institute of Technology (ETH) in August 2010 and in June 2012, respectively. Currently, he is a PhD student at the Department of Information Technology and Electrical Engineering at ETH, more particularly, at the Integrated Systems Laboratory (IIS). His research interests include low-power physical layer architectures, open-source protocol stacks, and cross-layer optimization in cellular communications.



Harald Kröll received his Dipl.-Ing. degree (summa cum laude) in Telematics from Graz University of Technology in 2010. During his Master's thesis he worked on Ultra Wideband Location Fingerprinting at the Wireless Communications Group at ETH Zurich. In the same year he joined the Integrated Systems Laboratory (IIS) at ETH Zurich where he is working towards a PhD degree. His research interests include signal processing and physical

layer architectures for wireless communication systems such as Evolved EDGE and LTE-Advanced.



Christian Benkeser received his Dipl.-Ing. degree in electrical engineering from the Karlsruhe Institute of Technology (KIT), Germany, in 2004. In the same year, he joined the Integrated Systems Laboratory (IIS) of the Swiss Federal Institute of Technology (ETH) Zurich, Switzerland, from where he graduated with the Dr. sc. degree in 2009. From 2004 to 2009, he was a research assistant with the IIS, and a consultant for Advanced Circuit Pursuit (ACP) AG, an

IC company in RF transceivers for cellular communications. From 2009 to 2012 he held positions as postdoctoral researcher at IIS and as senior design engineer at ACP AG. During this time he was leading a group of system and design engineers developing VLSI circuits and systems for wireless communications. In 2013, he joined RUAG Space, an independent supplier of space technology, where he is currently working as system engineer for opto-electronical systems for space applications. His research interests include signal processing, circuits and systems for wireless communication and space applications.



Qiuting Huang received his PhD degree in applied sciences from the Katholieke Universiteit Leuven, Belgium, in 1987. Between 1987 and 1992 he was a lecturer at the University of East Anglia, Norwich, UK. Since January 1993, he has been with the Integrated Systems Laboratory, Swiss Federal Institute of Technology (ETH), Zurich, where he is Professor of Electronics. In 2007 he was also appointed as a part-time Cheung Kong Seminar Professor

by the Chinese Ministry of Education and the Cheung Kong Foundation and has been affiliated with the South East University, Nanjing, China. Prof. Huang's research interests span RF, analog, mixed analog-digital as well as digital application specific integrated circuits and systems, with an emphasis on wireless communications applications in recent years. He has published widely on those topics in leading solid-state circuits conferences and journals.