

PROCESSOR BASED 20MHZ 4×4 CAT-5 LTE MIMO RECEIVER WITH ADVANCED DETECTORS

Min Li, Amir Amin, Rodolfo Torrea, Ubaid Ahmad, Raf Appeltans, Antoine Dejonghe, Liesbet Van Der Perre

{Firstname.Lastname@imec.be}, IMEC, Belgium

ABSTRACT

The Category-5 (Cat-5) UE defined by LTE, as the most demanding category, requires processing 20Mhz bandwidth and 4×4 MIMO transmissions. Very little progress has been reported for its feasibility on programmable processors. In fact, most related work focus on lower categories with much less throughput. Since MIMO signal processing complexity increases non-linearly even with the simplest linear MIMO detectors, 4×4 MIMO transmissions combined with 20Mhz bandwidth is much more challenging when compared to lower UE categories. Our work explores the feasibility of software defined baseband for the most demanding UE category. On a customized SDR baseband processor, we have recently accomplished a software defined downlink inner receiver for Cat-5 LTE UE. The implemented inner receiver includes fully fledged synchronization and data detection functionalities, including coarse CFO estimation/compensation, I/Q imbalance estimation/compensation, OFDMA demodulation, channel estimation, fine SCO/CFO estimation/compensation, MIMO channel processing, MIMO data detection and LLR generation. Both linear MIMO detectors and more advanced MIMO detectors have been studied. To the best of our knowledge, this is the first work experimenting practical Cat-5 LTE receivers on baseband processors.

1. INTRODUCTION

The data rate and spectral efficiency of emerging standards are continuously pacing fast. The 3GPP Long Term Evolution (LTE) and LTE-Advanced are representative examples. In the LTE standard, even a Cat-4 User Equipment (UE) can achieve 150 Mbps in the downlink with the Multiple Input Multiple Output (MIMO) technology. The Cat-5 UE defined by LTE can process 20Mhz bandwidth and 4×4 MIMO transmissions. This can achieve about 300Mbps downlink throughput. The standardization effort for high throughput wireless systems is becoming very intensive. Hence, although the Moore's Law predicted a fast evolution of the semiconductor integration, the increment of silicon capability is rapidly exhausted by the increasing transceiver complexity. Efficient air interface implementation becomes the bottleneck when deploying advanced wireless standards.

On the other hand, with the exploding design cost in the deep sub-micron era, the current trend is to implement most baseband functionalities on programmable baseband platforms. The Software Defined Radio (SDR) paradigm, which was mainly successful in the basestation and military segments, is currently emerging in the handset market as well. Especially, massively parallel instruction set processors, as those combining Instruction Level Parallel (ILP) and Data Level Parallel (DLP) features [1][2][3], are becoming prevailing. These baseband processors offer substantial parallelism for signal processing. In addition, multiple processors can be combined to strengthen the massive parallelism further. However, although these processors could potentially support emerging air interfaces, software defined receivers in existing literatures are still not compatible. For instance, [4] supports 31.67Mbps for DVB-T/H standard, [5] supports WCDMA and IEEE 802.11a, [6] supports LTE receiver working at only 100Mbps, [7] supports IEEE 802.11a/g, [8] supports WCDMA and IEEE 802.11a, [9] supports > 200Mbps but with the simple IEEE 802.11n standard. In addition, in some of the previous work, essential receiver blocks are still missing. For instance, Carrier Frequency Offset (CFO) synchronization, channel estimation/smoothing and so on were not considered in [5]. Clearly, it is still a challenge to deliver software defined receiver for those most demanding modes in emerging high throughput standards.

Our work tackles the challenge of the most demanding LTE UE category and explore the feasibility of SDR baseband. On state of the art SDR baseband platforms, the inner receiver (for synchronization and data detection) and outer receiver (for forward error correction) are normally implemented on different Application Specific Instruction Processors (ASIPs) [2]. In this paper, we will focus on the inner part of the downlink receiver, which has fully fledged signal processing functionalities including coarse CFO estimation/compensation, In-phase/Quadrature (I/Q) imbalance estimation/compensation, OFDMA demodulation, channel estimation, fine CFO and Sample Clock Offset (SCO) compensation, MIMO channel processing, MIMO data detection and soft demodulation (LLR generation).

We have shown the feasibility of software defined downlink inner receiver for LTE Cat-5 UE. The experiment was performed on a custom baseband processor evolved from [9]

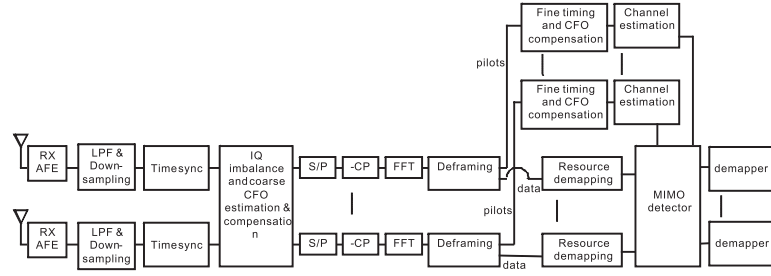


Fig. 1. Diagram of The User Data Processing Part of The Inner Receiver

[10]. Although the available parallelism on the baseband processor is not higher than previous ones such as [1] [3], the achieved data rate is substantially higher. Note that [1] [3] supports only WCDMA and IEEE 802.11a.

The rest of this paper consists of the following parts: section 2 discusses the considered receiver signal processing functionalities, section 3 discusses the architecture and brings detailed results, section 4 concludes the paper.

2. RECEIVER PROCESSING AND OPTIMIZATION

2.1. Functional Overview

The considered receiver is depicted in Fig.1. Received signals on 4 antennas pass through the analog front-end, are then low-pass filtered and down-sampled. After time synchronization on each antenna, I/Q imbalance and CFO are estimated and compensated. The I/Q imbalance estimation/compensation are accomplished per antenna, but CFO estimation/compensation are jointly done on all antennas. The I/Q imbalance is compensated because such analog impairment is present in most low cost analog front-end components used by consumer devices. Details of the I/Q imbalance and CFO estimation/compensation can be found in [11]. Note the above is a coarse estimation of the CFO, a fine estimation is performed later together with the channel estimation. After serial-to-parallel conversion and removing the cyclic prefix, FFTs are performed to move to the frequency domain, where the pilot and data signals are separated. Channels coefficients are estimated based on the pilot symbols, polar interpolation is performed in both the frequency domain and the time domain. Fine estimations of the residual SCO and CFO are jointly made at this stage. The channel estimates are forwarded to the MIMO detector together with the received data signals. The MIMO detection and demapping are then performed to generation soft information for transmitted bits. Importantly, we consider both conventional linear detector and more advanced MIMO detectors based on LR (Lattice Reduction).

For lower rank MIMO transmissions such as the 2×2 , conventional simple detectors, such as Zero-Forcing (ZF) is a very reasonable choice for practical SDR baseband implementations [12]. However, with large MIMO transmission systems, more advanced MIMO detectors becomes crucial. In practical cases, advanced detectors are often necessary to achieve coded BER lower than $10E-5$ as required in most

Table 1. List of Modules after Optimizations

Name	Functionality
procPSCH	Process PSCH for synchronization and analog impairment estimation
cpstCFO	Compensate I/Q imbalance and coarse CFO
FFT	2048pt FFT for OFDM demodulation
shuffle	Shuffle all user data carriers and group them
shuffle (Pilot, split)	Separate user data carriers and pilots then group them
compAngle	Calculate angles for fine CFO and SCO compensation
interpTime	Time interpolation
compDelta	Fine CFO and SCO compensation
QRdecomp	QR matrix decomposition
LR	Lattice reduction for advanced MIMO detection
Back subst	Back substitution for matrix inversion
HT	Channel matrix transformation based on LR
invQRExp	Matrix inversion
compLLRCoef	LLR coefficient calculation
compDelta	Channel matrix preprocessing for user data detection
equalize	MIMO equalization
softDemap	LLR generation for FEC

practical systems, we observed that ZF detector requires SNR to be higher than 40dB. This is often not feasible for most cost-constrained RF frontends and real life channel scenarios. Whereas LR aided advanced linear detector improves about 6 to 10dB SNR depending on channel conditions. The substantially relaxed SNR requirement enables the feasibility of high rank 4×4 MIMO transmission combined with high order modulation 64QAM. LR processing requires very complex operations such as rotations and size reductions, and the theoretical algorithm is almost impossible to implement. In order to bridge the gap, a number of optimizations have been proposed. More details of the advanced detector work can be found in [13].

2.2. Optimizations and Partitioning

Fig.1 shows the functional view on the inner receiver. However, implementation driven optimizations substantially change the partitioning of signal processing blocks. Extensive data flow transformation, control flow transformation and loop transformations have been applied. After the optimizations, the entire inner receiver is partitioned into modules as summarized in Table 1. Note that not all modules are always active. For instance, FFT is always required but LR is required by only advanced MIMO detection. Out of those modules, procPSCH is dedicated to process the Primary Synchronization Channel (PSCH), and all the rest are dedicated

for user data processing. All of the modules listed in Table 1 have been vectorized/parallelized to fully utilize the parallel processing capability offered by our baseband processor.

3. IMPLEMENTATION RESULTS

3.1. Architecture Template and Customization

Our custom baseband processor is based on the ADRES template [14]. An simple illustrative example is shown in Fig.2. The parameterizable template consists of an array of densely interconnected Function Units (FUs) that have local Register Files (RFs) and configuration memory for them. A limited subset of those FUs are connected to a global RF, enabling their operation also as a standard Very Long Instruction Word (VLIW) processor. The array part can be configured as Coarse Grain Array (CGA) mode, which allows to execute a large amount of operations in parallel. A retargetable C compiler, named as DRESC, targets both the VLIW and CGA modes. With the ADRES template, we can design an baseband processor with massive parallelism by combining ILP, DLP and extended custom instructions. Key high level features of our architecture customization are summarized in Table 2.

Given that cost is the top priority for consumer devices, the amount of parallelism of the baseband processor is not made larger than pervious state of the art baseband processors. For instance, the NXP EVP processor has 10 FUs (Function Units) and 6 of them support 16-way 16-bit SIMD [3], so that 96 16-bit operations can be performed in parallel on the vector processing part. The SODA processor supports even 32-way 16-bit SIMD instructions and 4 cores are included in 1 processor [1], so that 128 16-bit operations can be in parallel on the vector processing part. With our custom baseband processor, 64 16-bit operations can be performed in parallel on the vector processing part. Without increasing parallel resources, we tackle the signal processing complexity by effectively utilizing the available parallelism and accelerate important blocks with extended instructions.

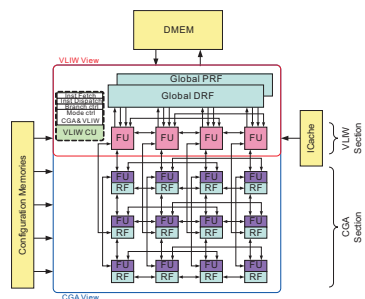


Fig. 2. Illustrative Example for The ADRES Template

The instruction set of the baseband processor has been heavily optimized. Besides basic SIMD instructions and usual signal processing instructions such as complex multiplications, the extended instruction set also contains many

Table 2. High Level Baseband Processor Architectural Features

Scalar processing	6 FUs supporting scalar operations, used for address generations, loop control, etc.
Vector processing	4 vector FUs supporting 256-bit Single Instruction Multiple Data (SIMD), which is 16 real SIMD slots of 16-bit each, or 8 complex SIMD slots of 32-bit each
Vector memory access	2 FUs that are connected to both vector memory and vector FUs
Vector/Scalar communication	3 packing/unpacking FUs connecting vector FUs and scalar FUs

Table 3. Important Extended Instructions

Functionality	Targeted signal processing blocks
Complex exp	SCO and CFO compensation,
Angle estimation	SCO and CFO estimation
Reciprocal	Matrix inversion, SCO and CFO estimation, LLR coefficient calculation, etc.
Reciprocal sqrt	Matrix inversion, SCO and CFO estimation, etc.
Soft demapping	Generating soft information
Efficient rounded division	Lattice reduction for advanced MIMO detection
Flag generation for LR	Lattice reduction for advanced MIMO detection
Execution condition for LR	Lattice reduction for advanced MIMO detection
Mask operation for LR	Lattice reduction for advanced MIMO detection

special arithmetic instructions and algorithm specific instructions. Several important extended instructions are summarized in Table 3. Significant effort has been investigated to reduce the implementation cost of these extra instructions. This is achieved by intensively reusing existing hardware to approximate the desired functionality. For instance, reciprocal and reciprocal square root instructions largely reuse existing multipliers, and the approximation incurs 1 bit error in rare cases. Complex exponent instruction incurs 3 to 6 bit errors depending on the input value. Importantly, approximation schemes of these instructions are co-optimized with algorithm development and simulation. This is to exploit the tolerable error of different signal processing blocks and minimizes the hardware cost. A detailed example can be found in [15], which accelerate the LLR generation by up to 21.9× with very low hardware cost.

3.2. Cycle Count Decomposition

As mentioned, the inner receiver consists of the PSCH processing part and the user data processing part. The PSCH processing is required only once for every 5 ms, so that the duty cycle is very low. The user data processing part is dominant for computation and memory complexity, because continuous data streams need to be processed. Hence, in the following, we will mostly focus on continuous user data processing.

3.2.1. With Linear Detector

With conventional ZF MIMO detector, the cycle counts of user data processing is summarized in Fig.3, where the X-

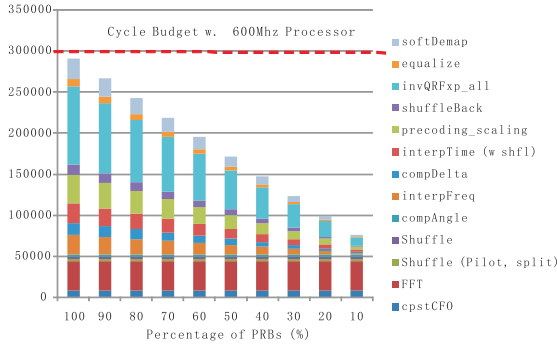


Fig. 3. Cycle Count with Conventional ZF MIMO Detector axis is the percentage of Physical Resource Blocks (PRBs) and the Y-axis is the number of required cycles. The maximum tested case is 100% PRB occupation, which corresponds to 100 PRBs with 12 carriers each. The cycle count numbers correspond to 4×4 20Mhz data streams and 1 sub-frame. 1 sub-frame consists of 7 size-2048 OFDM symbols and it is 0.5ms long. The dashed horizontal line marks the cycle budget corresponding to 0.5ms and a possible clock frequency of the targeted baseband processor. Currently, physical synthesis (with place and route) based results show that the processor implemented with 40nm technology can be clocked at at least 700Mhz under normal conditions. When supplied with higher voltage, the processor can reach up to 900Mhz.

We can observe that, even with 100% PRB occupation (all data sub-carriers are assigned to the specific user), the software defined receiver can easily run at real time with the baseband processor at only 600Mhz clock frequency. Low clock frequencies allow to reduce supply voltage and hence reduce power consumption. When clocked at higher frequencies, the slacks of cycle count may be utilized by other signal processing tasks. The cycle decomposition of the Cat-5 receiver, which supports the most demanding UE category, substantially differs from Cat-4 receivers. With 4×4 MIMO transmissions, MIMO channel related processing blocks become dominating. For instance, channel matrix inversion occupies nearly 1/3 and precoding block occupies 12%. In total, MIMO channel related processing blocks occupy about 2/3 cycle counts. The dominance of MIMO signal processing comes from the fact that the complexity increases very fast when increasing the MIMO transmission rank. For instance, channel interpolation complexity increases at least quadratically, whereas MIMO channel inversion has a cubic complexity growth.

3.2.2. With LR based Advanced Detector

With LR aided advanced MIMO detector, the cycle counts of user data processing is summarized in Fig.4. The figure is presented in a similar way as Fig.3. Two possible clock frequencies (800Mhz and 600Mhz). Although lattice reduction itself incurs a substantial complexity increment, we have proposed a number of techniques to make practical implementa-

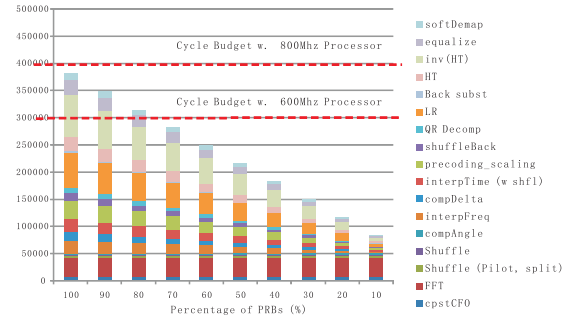


Fig. 4. Cycle Count with LR Aided MIMO Detector

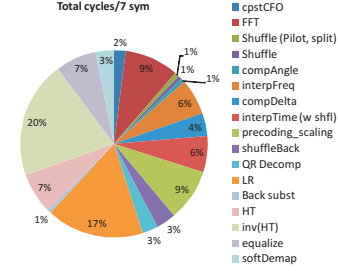


Fig. 5. Cycle Decomposition With LR Aided Detector (100% PRB))

tions feasible. It is worthwhile to mention that time and frequency down-sampling is very effective for reduce complexity 2. It has been implementable in this work and we perform LR only once for each sub-frame (7 OFDM symbols).

Although the receiver performance has been substantially improved with a more advanced MIMO detector, the implemented receiver can still run at real time when the baseband processor is clocked at 800Mhz. About 5% slack is available at this frequency. In practical cases, an UE often needs to process only a fraction of available sub-carriers, this offers a lot of opportunities for frequency scaling and resulted power reduction. The power consumption aspect is currently being studied. Fig.5 shows the cycle count decomposition with 100% PRB occupation, matrix inversion and LR are the two dominant blocks. In total MIMO channel processing occupies about 3/4 of the totally cycle count.

4. CONCLUSIONS AND FUTURE WORK

In our work, we explore the feasibility of software defined baseband for the most challenging UE category defined in LTE. We perform algorithm and processor architecture co-optimizations to enable highly efficient signal processing. In this paper, we have presented the software defined downlink inner receiver for Cat-5 LTE UE. On a custom baseband processor, the presented implementation can run at realtime even with a LR aided MIMO detector. We have shown the feasibility to have a software defined Cat-5 downlink receiver on baseband processor even with an advanced MIMO detector.

5. REFERENCES

- [1] Yuan Lin, Hyunseok Lee, Mark Woh, Yoav Harel, Scott Mahlke, Trevor Mudge, Chaitali Chakrabarti, and Krisztian Flautner, "Soda: A high-performance dsp architecture for software-defined radio," *IEEE Micro*, vol. 27, no. 1, pp. 114–123, 2007.
- [2] Ulrich Ramacher, "Software-defined radio prospects for multistandard mobile phones," *Computer*, vol. 40, no. 10, pp. 62–69, 2007.
- [3] Kees van Berkel, Frank Heinle, Patrick P. E. Meuwissen, Kees Moerman, and Matthias Weiss, "Vector processing as an enabler for software-defined radio in handheld devices," *EURASIP J. Appl. Signal Process.*, vol. 2005, no. 1, pp. 2613–2625, 2005.
- [4] D. Liu, A. Nilsson, E. Tell, D. Wu, and J. Eilert, "Bridging dream and reality: Programmable baseband processors for software-defined radio," *Communications Magazine, IEEE*, vol. 47, no. 9, pp. 134–140, sep. 2009.
- [5] H. Lee, C. Chakrabarti, and T. Mudge, "A low-power dsp for wireless communications," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 18, no. 9, pp. 1310–1322, sep. 2010.
- [6] C. Jalier, D. Lattard, A.A. Jerraya, G. Sassatelli, P. Benoit, and L. Torres, "Heterogeneous vs homogeneous mpsoc approaches for a mobile lte modem," mar. 2010, pp. 184–189.
- [7] Zong Wang and T. Arslan, "A low power reconfigurable heterogeneous architecture for a mobile sdr system," may. 2009, pp. 2025–2028.
- [8] D. Auras, S. Girbal, H. Berry, O. Temam, and S. Yehia, "Cma: Chip multi-accelerator," jun. 2010, pp. 8–15.
- [9] V. Derudder, B. Bougard, A. Couvreur, A. Dewilde, S. Dupont, L. Folens, L. Hollevoet, F. Naessens, D. Novo, P. Raghavan, T. Schuster, K. Stinkens, J.-W. Weijers, and L. Van der Perre, "A 200mbps+ 2.14nj/b digital baseband multi processor system-on-chip for sdrs," jun. 2009, pp. 292–293.
- [10] Bruno Bougard, Bjorn De Sutter, Sebastien Rabou, David Novo, Osman Allam, Steven Dupont, and Liesbet Van der Perre, "A coarse-grained array based baseband processor for 100mbps+ software defined radio," *Design, Automation and Test in Europe, 2008. DATE '08*, pp. 716–721, 10–14 March 2008.
- [11] E. Lopez-Estraviz, S. De Rore, F. Horlin, and L. Van der Perre, "Optimal training sequences for joint channel and frequency-dependent iq imbalance estimation in ofdm-based receivers," jun. 2006, vol. 10, pp. 4595–4600.
- [12] Min Li, Raf Appeltans, Amir Amin, Rodolfo Torrea Duran, Hans Cappelle, Matthias Hartmann, Hidekuni Yomo, Kiyotaka Kobayashi, Antoine Dejonghe, and Liesbet Van der Perre, "Overview of a software defined downlink inner receiver for category-e lte-advanced ue," in *ICC*, 2011, pp. 1–5.
- [13] U. Ahmad, A. Amin, Min Li, S. Pollin, L. Van Der Perre, and F. Catthoor, "Scalable block-based parallel lattice reduction algorithm for an sdr baseband processor," in *Communications (ICC), 2011 IEEE International Conference on*, june 2011, pp. 1–5.
- [14] Bingfeng Mei, Andy Lambrechts, Diederik Verkest, Jean-Yves Mignolet, and Rudy Lauwereins, "Architecture exploration for a reconfigurable architecture template," *IEEE Des. Test*, vol. 22, no. 2, pp. 90–101, 2005.
- [15] Min Li and et. al, "Instruction set support and algorithm-architecture for fully parallel multi-standard soft-output demapping on baseband processors," *Signal Processing Systems, 2010 IEEE Workshop on*.