Scenario-Based Fixed-point Data Format Refinement to Enable Energy-scalable Software Defined Radios

D. Novo[12], B. Bougard[1], A. Lambrechts[12], L. Van der Perre[1] and F. Catthoor[12]
[1] IMEC vzw, Kapeldreef 75, 3001 Heverlee, Belgium
[2] KULeuven, 3000 Leuven, Belgium
{novo, bougardb, lambreca, vdperre, catthoor}@imec.be

Abstract - User demand, standards and products for digital nomadic communications are evolving quickly. The combination of this changing environment together with the need for short time-to-market pushes for more flexible implementations. Software Defined Radios (SDR) have been introduced as the ultimate way to achieve such flexibility. The reduced energy budget required by battery-powered solutions makes the typical worst-case static dimensioning unaffordable under highly dynamic operating conditions. Instead, more energy-scalable algorithms and implementations are entailed to provide flexibility while maintaining the required energy efficiency. Particularly, energy-scalable implementations can exploit data format properties to offer different tradeoffs between accuracy and energy. In this paper, such a technique is developed and applied to the SDR implementation of a 2 antennas 200 Mbps+ OFDM (Orthogonal Frequency-Division Multiplexing) inner modem receiver on a C-programmable CGA (Coarse Grain Array) processor with extensive SIMD (Single Instruction Multiple Data) support. By defining separate implementations for different combinations of modulation scheme and coding rate, up to 3-fold gains can be achieved in the average energy consumption.

I. Introduction

The combination of the continuously growing variety of wireless standards and the increasing cost related to integrated circuit designs is clearly pushing for more flexible radios. Software Defined Radios, where the baseband processing is carried out on programmable and/or reconfigurable hardware have been introduced as the ultimate way to achieve this flexibility. SDR can lead to short time-to-market, rapid product derivate development and long product life cycle. However, deploying SDRs in battery-limited handheld devices makes high energy efficiency a vital feature. Although typical programmable architectures, such as General Purpose Processors (GPP) or Digital Signal Processors (DSP) provide energy efficiencies that are orders of magnitude lower than dedicated hardware solutions, Application Specific Instruction set Processors (ASIP) are intended to reduce this energy gap [1]. Nevertheless, the ASIP architecture paradigm alone is still not sufficient to reach the necessary energy efficiency.

From the application point of view, wireless communications, and especially SDR solutions, are facing very dynamic operating conditions. The propagation channel, the interfering environment and the use of the wireless interface are varying. In addition, both environment and use variations exhibit a large dynamic range. Obviously, these two characteristics have an important impact on the dimensioning of an SDR platform. The presence of the highly dynamic operative conditions would lead to an unaffordable overhead when the typical static worst-case dimensioning approach is considered. Consequently, we believe that both energy-scalable algorithms and implementations combined with adaptive performance/energy management are crucial to finally conquer the energy gap that separates flexible and dedicated implementations. Such an approach can achieve high energy efficiency as it has the potential to continuously best-fit the dynamic behaviors. For instance, energy-scalable algorithms enable a device to continuously trade off system performance and execution cost (processing load and energy) at run-time to match the dynamic requirements. In [2], this technique is applied to implement an energy-scalable channel equalization algorithm, which is able to save up to 60% of the average execution time on a DSP at negligible system performance loss.

Similarly, but at a lower implementation level, data formats can exploit the signal range and precision dynamics to offer different tradeoffs between computation accuracy and energy consumption. In communication signal processing systems, I/O correctness does not need to be preserved in the strict sense. Approximations can generally be accommodated while maintaining the desired system performance, as communication algorithms can still function under different Signal-to-Noise Ratio (SNR) conditions. However, this tolerance to inaccuracy is dependent on the system working conditions. For instance, processing the equalization and demodulation of a signal modulated with a high order constellation may require higher accuracy than in the case of a low order one. In order to reach scalability this accuracy adjustment can be performed separately for different use-cases or scenarios. Certainly, these scenarios should be sufficiently easy to detect/distinguish at run-time. In this paper, the inner modem of a 2 antenna 200Mbps+ OFDM receiver is mapped onto a CGA processor with extensive SIMD support [3]. We will see that, by defining separate implementations for different combinations of modulation scheme and coding rate, important gains can be achieved in the average energy consumption.

The rest of the paper is organized as follows: In Section 2 the related work and the motivation for this work are presented. The proposed data format refinement approach is introduced in Section 3. Its validity is illustrated in Section 4 with the extensive case study. Finally, Section 5 draws conclusions.

978-3-9810801-3-1/DATE08 © 2008 EDAA

II. Related work

Finite word-length refinement for data format selection has been an active research field for more than 30 years. Traditionally, most contributions have focused on the development of methods and tools that automatically convert a floating-point spec into an optimal fixed-point representation under a given user-defined Quantization Noise to Signal Ratio (QNSR). Most of the existing work on this area, agree on splitting the optimization problem in two steps: range analysis and precision analysis [4, 5, 6]. The range analysis provides the margin to accommodate the growth of the data (avoiding overflow), whereas the precision analysis guarantees the accuracy of the operations. For both, range and precision analysis, dynamic and static analysis methods have been proposed. Firstly, the dynamic analysis methods, also called simulation based methods, evaluate the Data-Flow Graph (DFG) of the design using representative input signals [4]. Secondly, the static analysis methods, also called analytical methods, propagate statistic characteristics of the inputs through the DFG [5]. Finally, hybrid approaches have been proposed [6], which aim to combine the advantages of both the static and dynamic methods.

This previous work assumes that the data format assignment is performed under worst-case conditions at design-time, which would lead to sub-optimal solutions under the highly dynamic operating conditions of the SDR context considered here. Alternatively, [7] proposes a word-length tunable VLSI architecture for a wireless demodulator where the word-length selection is done at run-time depending on the observed error. This approach saves up to 30% of the power. However it assumes a dedicated hardware implementation and requires the addition of a special field (containing the known sequence used to estimate the current quantization error) into the transmission packet format. The latter jeopardizes its implementation in standard-compliant systems.

In this work we propose a more industry compatible approach to exploit the variations on the instantaneous minimum required precision in an energy-scalable manner, without compromising the standard compliance of the implementation. This is achieved by partially porting the data format decisions to the run-time in a scenario-based manner [8]. Multiple implementations, corresponding to specific use-cases or scenarios, are optimized separately at design-time and selected by a simple controller at run-time. The latter decides which implementation is more efficient given the current conditions. This technique does not depend on the selected fixed-point refinement approach (dynamic vs. static) but considers the application knowledge (through the scenario definition) to effectively guide the refinement process.

III. Scenario-oriented fixed-point refinement

In current design methodologies, data formats are typically dimensioned at design-time. This dimensioning aims to satisfy the application requirements under all the possible conditions. As an alternative, a scenario-oriented data format refinement, which consists of a hybrid design-/run-time approach, is proposed. In that approach, situations/scenarios where the application exhibits a different tolerance to the quantization noise are identified. Accordingly, separated fixed-point refinements are performed for each of these scenarios, resulting in multiple software implementations. At run-time, the actual scenario that best suits the current working conditions is detected and the corresponding implementation is selected by a simple controller.

Scenarios where the application exhibits a different tolerance to the noise are very common in communication systems as the channel is considered as an unpredictable source of noise and attenuation. The degree of uncertainty is especially important in wireless communications, where the system has to deal with varying distances between transmitter and receiver under limited transmission power. This imposes different SNR situations where the system needs to be functional. Besides the distance between transmitter and receiver, other random physical phenomena, such as multipath fading, can also seriously affect the received SNR.

As an example, OFDM systems, when used in the context of wireless communications (e.g. IEEE 802.11 family), are designed to provide several compromises between data rate and coverage [9]. Accordingly, they offer various operational modes by implementing different combinations of sub-carrier modulation scheme and coding rate. The modulation scheme defines the amount of bits that are grouped together and transmitted in one sub-carrier (e.g. 1 bit for BPSK, 2 for QPSK, 4 for 16QAM and 6 for 64QAM) and thus importantly impacts the physical data rate. The coding rate determines the amount of redundancy added to the transmitted bit-stream to enable Forward Error Correction (FEC) to be performed at the receiver. This recovers transmission errors by collecting time and frequency diversity. Reducing the modulation order or/and reducing the code rate decreases the data rate but improves the robustness of the system to the channel noise and attenuation. In the implementation of such systems, the varying noise-robustness exhibited by the application should be taken into consideration. The extra degradation that would be introduced by moving to a cheaper fixed-point implementation could be tolerated, depending on the situation.

In [10] the quantization of a signal is modeled by the sum of this signal and a random variable. This additive noise is a stationary and uniformly distributed white noise that is not correlated with the signal and the other quantization noises. Thus, the effect of refining an ideal (infinite precision) algorithm into a fixed-point implementation can be modeled as the initial algorithm of ideal operators fed with the sum of the ideal operands and a noise component (quantization noise). Then this noise can be forward propagated towards the inputs of the algorithm and be assumed to belong to the channel.

Consequently, the transmission modes that tolerate higher levels of channel noise in the received signal should also be able to accept higher levels of quantization noise on their processing. These modes will require fewer bits to maintain the necessary accuracy.

As the end goal is to achieve energy scalable radios and because of the large amount of data parallelism that is typically exploitable in wireless baseband algorithms, we leverage on the SIMD paradigm to translate reduced bit-width to less energy. In particular, we exploit the fact that multiple data (sub-words) can be packed together and operated on as a single word. The size of these sub-words is variable and can be selected from a discrete set, typically of powers-of-2. The different sub-word configurations share the same hardware operators, which are configured depending on the current sub-word size (e.g. cutting the carry propagation in an adder). The result is called a sub-word parallel instruction-set datapath. So in this paper, we will not consider pure vector processors (with fixed sub-word size) which form the other typical SIMD class. The cost in energy and time associated with the operation (operand load, execution, result storage) is shared by all the sub-words. Consequently, the fewer the bits required to represent the data, the more the data we can pack together and the cheaper the processing per sub-word becomes.

Finally a controller needs to track the environmental conditions, identify the current scenario and react by selecting the most efficient implementation. The overhead introduced by this controller needs to be kept low in order to maximize the benefits given by the splitting of the application into different optimized scenarios.

IV. A Case study: Multi-antenna OFDM receivers for next generation wireless LAN

In this section, a high-rate OFDM receiver is presented as a case study to illustrate the validity of the scenario-oriented data format refinement approach. The receiver functionality is implemented on a C-programmable CGA processor.

A. *A multi-antenna OFDM receiver*

Wireless communication systems must generally deliver 10x more data rate from generation to generation. In Wireless LAN (Local Area Network) systems in particular, the emerging generation, which answer to the IEEE 802.11n standard, achieve this data rate increase by leveraging on multiple antenna transmission techniques, especially on the so-called Space Division Multiplexing (SDM), and on a larger bandwidth. In SDM, multiple unique data streams are transmitted in the same frequency band at the same time through different antennas. Accordingly, the system data rate grows about linearly with the number of parallel data streams. Channel bonding provides higher data rates by expanding the operating bandwidth.

TABLE I

SNR required to deliver 10^{-3} BER of the modes considered.

mode	# ant.	mod. scheme	cod. rate	SNR [dB] BER = 10^{-3}
1	1	BPSK	1/2	3.0
2	1	BPSK	3/4	7.5
3	1	QPSK	1/2	6.5
4	1	QPSK	3/4	10.5
5	1	16QAM	1/2	12.5
6	1	16QAM	3/4	25.5
7	1	64QAM	2/3	20.5
8	1	64QAM	3/4	22.3
9	2	BPSK	1/2	5.5
10	2	BPSK	3/4	10.5
11	2	QPSK	1/2	11.5
12	2	QPSK	3/4	16.5
13	2	16QAM	1/2	18.0
14	2	16QAM	3/4	25.5
15	2	64QAM	2/3	31.0
16	2	64QAM	3/4	34.0

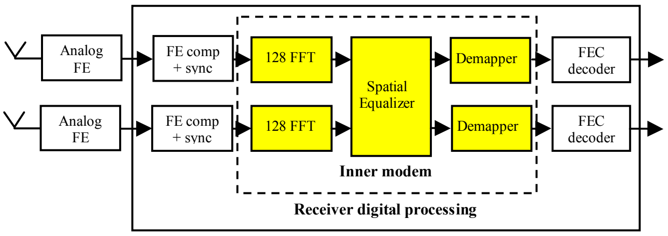


Figure 1: Block diagram of the OFDM receiver considered, with the implemented blocks highlighted.

In this work, we consider a 2-antennas SDM transceiver which combines two adjacent 20 MHz channels into a single 40 MHz one. This configuration enables data rates higher than 200 Mbps. Fig. 1 shows the block diagram of the receiver. The Fast Fourier Transform (*FFT*) together with the *Spatial Equalizer* and the *Demapper* constitute the so-called inner modem. The *FFT* block processes vectors of 128 complex elements. The *Spatial Equalizer* cancels out the channel distortion and the inter-stream interference. A Minimum Mean Square Error (MMSE) filter is implemented, consisting of 114 times (one per non-zero sub-carrier) the multiplication of a 2x2 complex matrix by a 2-elements complex vector. Finally, the *Demapper* translates the constellation symbols into bits.

This application exhibits no data-dependent execution (the data-dependent behavior of the *Demapper* can be eliminated by including a Look-Up Table). Moreover the processing is block-based, meaning that it continuously performs the same operations over blocks of 128 carriers (OFDM symbol). These two characteristics, together with a relaxed latency constraint (present in the transmission of long packets), enable block-SIMD processing. This packs carriers belonging to consecutive OFDM symbols together in a single word. Thus, the addition of a new sub-word into the original word just implies the buffering of another symbol while the control flow remains identical. This technique leads to a negligible SIMD overhead since the input buffer is typically present in current wireless architectures for synchronization purposes [11] and the data shuffling required is minimal. Consequently, by doubling the amount of sub-words packed into a word one can expect about to halve the average energy and execution time.

Before any fixed-point refinement, the selected wireless system is simulated under ideal precision conditions for the different receiver modes. This is illustrated in Table I, which shows the minimum SNR required to achieve a Bit Error Ratio (BER) of 10^{-3} for the different operation modes. The level of noise that guarantees a certain transmission performance, such as a BER below 10^{-3}, interestingly varies depending on the selected mode.

B. *Processor framework*

The processor framework considered, namely ADRES [3], consists of a templated array of interconnected functional units (FU) which have a local register-file (LRF) and configuration memory (IB). A limited subset of these units is connected to a shared multi-ported register-file (Shared-RF), enabling their operation also as standard VLIW (Very Long Instruction Word) processor.

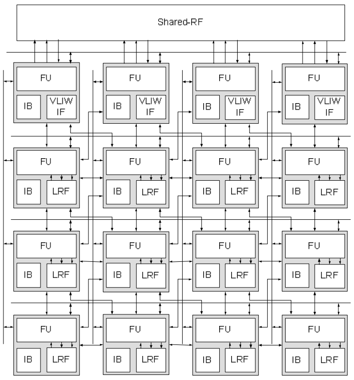


Figure 2: Diagram of the ADRES instance considered.

A retargetable C compiler named DRESC, targets both the VLIW and the array modes. DRESC transparently maps data-flow dominated loops into the array and schedules the rest of the code in the VLIW part. The compiler automatically achieves high Instruction Level Parallelism (ILP) [3]. In contrast, the Data Level Parallelism (DLP) can be handled by the programmer via intrinsic C functions.

The ADRES instance considered throughout the rest of the paper (see Fig 2), consists on a 4x4 array of 32 bit FUs. These units provide traditional DSP functionality extended with an extensive SIMD support. Table II lists some of the Instruction Set Architecture (ISA) extensions included to exploit 2-ways SIMD. Similarly, the implemented ISA also includes 4-ways and 8-ways SIMD so it is sub-word parallel. Note that we can also use other sub-word parallel processors as target. Commercial examples include the TI C6x range, NXP Trimedia range, ST 200 range, ARM Cortex and so on. But as mentioned earlier, pure vector processors would not be useful.

An RTL description of the instance was synthesized with Synopsys Design Compiler^{TM} targeting a state-of-the-art 90 nm standard cell library. The resulting gate-level netlist was used as the input for physical design with Cadence SOC encounter^{TM}. Standard cell placement is then optimized, followed by clock tree synthesis and final place&route. After parasitic extraction from the resulting layout, timing and power estimation were carried out. Timing was checked with Synopsys PrimeTime^{TM} showing an achievable operating frequency of 400 MHz. Power was estimated with Synopsys PrimePower^{TM} based on the switching activity traced during the back-annotated gate-level simulation of a 2304bytes 802.11-packet reception.

TABLE II

ISA extensions added to support 2-ways SIMD. The inputs (src1 and src2) and the output (dst) contain 2 concatenated sub-words, indicated with a final R and I.

Instr	Description	Pseudo code
cadd	complex addition	dstR = src1R + src2R dstI = src1I + src2I
csub	complex subtraction	dstR = src1R - src2R dstI = src1I - src2I
cshftr	complex right shifter	dstR = src1R >> src2 dstI = src1I >> src2
dprod	dot product	dstR = src1R * src2R dstI = src1I * src2I

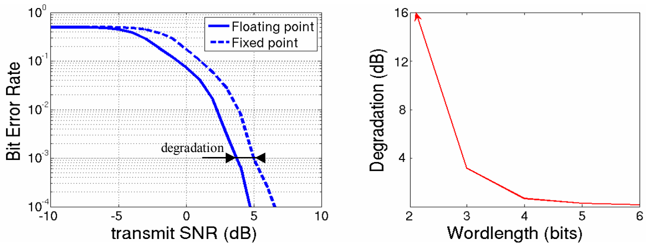


Figure 3: BER curves of the SISO BPSK ½, before (solid) and after (dashed) fixed-point refinement (left) and its *BER degradation* as function of the data wordlength (right).

C. *Fixed-point data format exploration*

A simulation-based approach is applied to cover the fixed-point data format refinement process. This can easily propagate the degradation introduced by the finite precision signals to the high-level performance metrics such as BER. In order to enable a fixed-point simulation, the signals of the initial floating-point description are instrumented. This is done by including a set of functions in the initial code which have as input the original floating-point signal and outputs a fixed-point representation. The conversion is controlled with a set of parameters. The total number of bits per signal, the number of decimal bits, the quantization mode (round or truncation) and the overflow mode (wrap-around or saturation) are the most important parameters. After giving a value to those parameters, the entire communication chain can be simulated with fixed-point precision. Consequently, the impact on the application performance of the selected fixed-point configuration (given by the set of values introduced in the instrumentation function) can be estimated. By an iterative process, one can obtain the optimal set of parameters that satisfies a desired performance while minimizing the signals' word-length. Details on this process are out of the scope of this work, but the interested reader is referred to [4]. Instead, we will concentrate on how different fixed-point configurations, associated with different receiver conditions (scenarios), can provide important energy savings while keeping degradation to the system performance under control. For convenience, we restrict the exploration space to the traditional power-of-two word-lengths, encountered in most DSP architectures. Saturation arithmetic and rounding are also assumed.

In order to properly steer the fixed-point refinement, an application performance indicator needs to be defined. The BER curve plots the ratio of erroneous bits received at different SNR conditions. Due to the finite precision effects, the BER curve experiences a shift to the right which is commonly referred to as *implementation loss* (see Fig 3a). We define the *BER degradation* as the difference in SNR between the floating-point and the fixed-point representation at which the system delivers a BER of 10^{-3} (minimum performance required for a reliable transmission). Then, the goal of the fixed-point refinement step is to reduce signal bit-width while keeping the *BER degradation* below a user-defined value. Fig. 3b represents the *BER degradation* as function of the signal word-length. The curve monotonically grows with the bit reduction up to a point where reaches infinite degradation. This point indicates that the BER curve floors before the 10^{-3}, and performance is corrupted beyond the acceptable range.

TABLE III
Word-lengths of the receiving modes for different *BER degradations*

mode	0.5dB [bits]	1.5dB [bits]	2 dB [bits]
1	8	8	4
2	8	8	8
3	8	8	8
4	8	8	8
5	8	8	8
6	8	8	8
7	16	8	8
8	16	16	8
9	8	8	8
10	8	8	8
11	8	8	8
12	8	8	8
13	16	8	8
14	16	16	16
15	16	16	16
16	16	16	16

Following this strategy, the different receiver modes/configurations were refined independently. In this work all the configurations were assumed to have the same word-length along the different processing blocks. This reduces the overhead introduced by the inter-block shuffling operations. However, it also reduces the opportunity of having smaller word-lengths. During the fixed-point refinement, different BER degradation factors were also explored. Table III shows the resulting bit-widths. Notice that with a maximum BER degradation of 0.5 dB an important number of modes can be represented with half of the bits that are used in typical implementations. Moreover, the increase of BER degradation gradually enables even shorter word-lengths.

D. *Scenario definition and detection: Link adaptation*

The various modes of the receiver provide different compromises between raw data rate and noise robustness. Since a wireless receiver also experiences different SNR conditions depending on the specific moment, the mode that performs better under the given conditions should be selected. This selection is already done by the base station and the receiver controller just needs to identify the selected modulation mode (information included in the received preamble) and switch to the corresponding implementation at run-time.

Typically, in order to decide which mode is the most appropriate for a given SNR, the link adaptation procedure identifies the mode that achieves the highest average throughput at that SNR (Fig. 4 plots a reduced set for illustration purposes). The figure shows that the different receiver configurations have a SNR region where they outperform the others: A for the BPSK, B for the QPSK, C for the 16QAM and D for the 64QAM. We can assume that when the receiver is in a given SNR condition, the highest throughput configuration will be selected. From this we can define the envelope of all the throughput curves as the system performance indicator. This approach also enables easy scenario detection at run-time.

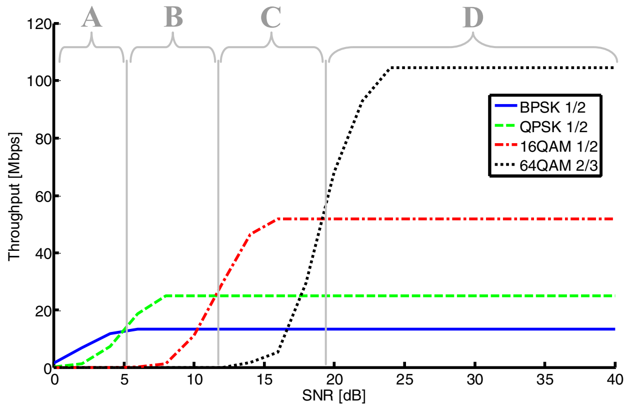


Figure 4: Throughput curves of different receiver modes.

E. *Energy-Throughput degradation trade off*

After splitting the application into the different scenarios, the inner receiver blocks introduced in Section IV.A have been implemented with the different resolutions indicated in Table III. The entire communication system has then been simulated and throughput curves extracted for the different implementations. The throughput curve results from averaging on 50 packets of 2304 bytes (largest packet size) transmitted over 50 different channels. Ideal synchronization and channel estimation were assumed. Following the technique introduced in the previous subsection, the set of scenarios (link adaptation) was defined for 3 different cases: a traditional all-modes 16 bit implementation (reference implementation since is the worst-case precision requirement) and a scenario-based data formatted implementation when allowing 0.5 and 2dB BER degradation. The throughput envelope of the 3 cases considering the SISO (Single-Input Single-Output) and the 2 antennas SDM mode are plot in Fig. 5a and Fig. 6a respectively. In addition, their corresponding energy per transmitted bit, estimated by the flow described previously, is plot in Fig. 5b and Fig. 6b. One can observe that the low rate configurations consume more energy than the high rate ones. This can be easily understood since for transmitting the same amount of information, the low rate configurations need to send more OFDM symbols (due to the lower modulation order and/or lower coding rate). Consequently the processor needs to process during a longer time consuming more energy per bit of information.

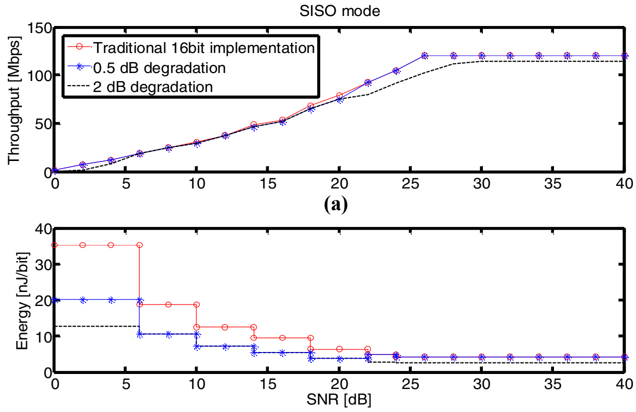
When little BER degradation is allowed (e.g. less than 0.5 dB), negligible system performance loss is observed. However the energy per bit of the lower rate configurations is considerably reduced. For instance, in the region from 0-6 dB of the SISO case (see Fig. 5) the 16bits sub-word implementation can be reduced to 8bits resulting in a 43% energy saving. Due to the leakage power the reduction is slightly less than the ideal 50%. When more BER degradation is allowed (e.g. less than 2 dB), the performance starts to suffer. As an indicator, when less than 2dB degradation is allowed and the receiver works with a 3dB SNR, the maximum throughput drops by 53%. However a 4bits implementation can now be accommodated, which reduces the energy by a 66%. In this case some performance is traded off by energy.

However, this is not always the case, in the 2 antennas SDM case (see Fig. 6) an unexpected effect appears in the SNR region from 8-10 dB: more BER degradation comes with higher energy consumption. This happens because by enabling more degradation the throughput curves are modified and consequently the link adaptation is affected. This SNR region that before degradation was covered by a 16 bits QPSK modulation, is now implemented by a 8 bits BPSK modulation. Even though the BPSK is implemented with fewer bits, it still consumes more energy per bit of information than the QPSK. Under these particular circumstances, the implementation with higher accuracy offers both better performance and a more energy efficient execution and therefore must be selected as optimal.

V. Conclusions

Software Defined Radios enable, on a single platform, the coexistence of many different implementation options without the high overhead in area and power occurred in previous ASIC-based solutions. This new feature of radio platforms has to be propagated throughout the different design steps in order to fully utilize its potential. In this work, we particularly motivate the move from the traditional design-time fixed-point data format assignment to a scenario-based approach. This is illustrated with the implementation of the inner modem of a 2 antenna 200Mbps+ OFDM receiver on a C-programmable CGA processor with extensive SIMD support. When some fixed-point degradation is allowed (e.g. less than 0.5 dB *BER degradation*), important gains in energy are achieved at negligible system performance loss. If more degradation is permitted (e.g. less than 2 dB *BER degradation*), dramatic energy savings are achieved, however, the system performance starts to be seriously compromised. As an example we report a 3-fold energy reduction at expenses of a 53% loss in average throughput.

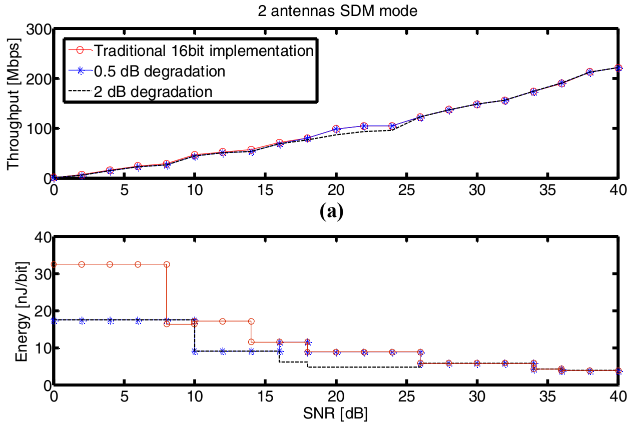
Finally, the introduction of finite precision noise can modify the region where a given receiver mode performs optimally. This can result in situations where higher fixed-point degradation does not mean less energy consumption. Under these specific situations, the implementation with less degradation must be selected.



(a)

(b)

Figure 5: Curves of the SISO throughput performance (a) and energy consumption (b).



(a)

(b)

Figure 6: Curves of the SDM throughput performance (a) and energy consumption (b).

References

[1] H. Meyr: System-on-chip for communications: The dawn of ASIPs and the dusk of ASICs. Proc. SIPS'03. Seoul, Korea (2003).

[2] Min Li et Al.:Quality-Cost Scalable Chip Level Equalizer in HSDPA Receiver. Globecom '06. San Francisco (2006).

[3] Novo, D. et Al.: Energy-Performance Exploration Of a CGA-based SDR Processor. SDR Forum '06. Orlando (2006).

[4] Gaffar, A et Al.: Unifying bit-width optimisation for fixed-point and floating-point designs. FCCM'04. Napa Valley, USA (2004). 79–88.

[5] A. Nayak et Al.: Precision and error analysis of MATLAB applications during automated hardware synthesis for FPGA's. DATE'06, Munich, Germany (2001). 722–728.

[6] Menard, D. Sentieys, O.: DSP Code Generation with Optimized Data Word-Length Selection. LNCS 2004 issue 3199. 214-228.

[7] Yoshizawa, S.; Miyanaga, Y. Tunable Wordlength Architecture for a Low Power Wireless OFDM Demodulator. ISCAS '06, Kos, Greece (2006).

[8] J.M. Carroll (Ed.) Scenario-based Design: Envisioning Work and Technology in System Development. Wiley, New York, NY, 1995.

[9] Wireless LAN MAC and PHY Specifications: Further Higher Data Rate Extension in the 2.4 GHz Band. IEEE Standard 802.11g/D8.2 (2003).

[10] B. Widrow, I. Kollar, and M.-C. Liu, Statistical theory of quantization, IEEE Trans. on Instrumentation and Measurement, vol. 45, no. 6, pp. 353--61, 1995.

[11] L Harju, J Nurmi: A programmable baseband receiver platform for WCDMA/OFDM mobile terminals. IEEE Wireless Comm. and Networking Conf. New Orleans, USA (2005).

[12] M.F. Tariq et Al.: Development of an OFDM based high speed wireless LAN platform using the TI C6x DSP. ICC'02. New York, USA (2002).