Microelectronics Journal 45 (2014) 167–178

Contents lists available at ScienceDirect

Microelectronics Journal

journal homepage: www.elsevier.com/locate/mejo

Design and verification of a frequency domain equalizer

Anis Souari [a], Amjad Gawanmeh [b], Sofiène Tahar [a,*], Mohamed Lassaad Ammari [c]

[a] Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, Canada H3G 1M8
[b] Department of Electrical and Computer Engineering, Khalifa University of Science, Technology and Research, P.O. Box 573 Sharjah, United Arab Emirates
[c] École Nationale d'ingénieurs de Sousse, Sousse, Tunisia

ARTICLE INFO

Article history:
Received 20 May 2013
Received in revised form
20 October 2013
Accepted 23 October 2013
Available online 21 November 2013

Keywords:
Error analysis
Theorem proving
Design and verification

ABSTRACT

In this work we provide a methodology for the design and verification of a frequency domain equalizer. The performance analysis of the equalizer is conducted using two methods: simulation based verification in Simulink and System Generator and theorem proving techniques in Higher Order Logic. We conduct both floating-point and fixed-point error estimations for the design in Simulink and System Generator, respectively. Then, we use formal error analysis based on the theorem proving to verify an implementation of the frequency domain equalizer based on the Fast LMS algorithm. The formal error analysis and simulation based error estimation of the algorithm intend to show that, when converting from one number domain to another, the algorithm produces the same values with an accepted error margin caused by the round-off error accumulation. This work shows the efficiency of combining simulation and formal verification based methods in verifying complex systems such as the frequency domain equalizer.

1. Introduction and motivation

With the recent technological growth, electronic devices have invaded all aspects of our lives. These devices are getting more and more compact and consequently more complex. The price of this complexity is the challenge of delivering error-free devices, which require thorough testing and verification at all stages of the design flow. On the other hand, a faulty design can lead to costly delays for the time-to-market. Therefore, design verification is necessary to avoid such situations and is considered a bottleneck in the design process. In order to verify that an implementation meets its specifications, simulation is the most widely used technique in the industry, because it is straightforward and does not need any expertise. This simulation is based on the generation of test patterns, and therefore, it does not provide full coverage for the system under test. On the other hand, formal verification techniques [1] are considered complementary to simulation as they can provide full coverage for the system under test, and in addition, they can catch corner case bugs in the design. Formal verification does require a certain level of expertise to be efficiently used, which may incur a considerable human resources cost.

Equalization is an application of adaptive filtering that can eliminate the inter-symbol interference caused by the noise in the transmission environment. Uncountable adaptive algorithms are used to regulate the filter or the equalizer coefficients in order to match the output to the desired response. To decrease the filtering complexity, the equalizer can be implemented in the frequency domain using the Fast Fourier Transform (FFT) and the Inverse FFT (IFFT), where time convolution is replaced by frequency multiplication. This method offers low complexity growth in comparison with the time domain method. Data processing and filtering require dealing with data at different domains: real numbers, floating-point numbers, and fixed-point numbers. The specification of an equalizer design can be given in the floating-point domain, while the design implementation can be conducted in the fixed-point domain. This conversion generates and accumulates errors due to the different levels of accuracy provided by each number's domain. Therefore, a frequency domain multiplication based system must be tested thoroughly, and error analysis must be conducted to be sure about the correctness of its operation.

Verifying the correctness of an equalizer is very challenging because, firstly, its implementation can be based on an iterative algorithm, secondly, it can contain multiple FFT and IFFT blocks, and finally, it may contain multiple mathematical operators in different number domains. As a result, errors are naturally generated during data conversion between these domains, and can accumulate while performing various algorithmic iterations, FFT and IFFT operations. Therefore, a particular implementation of such a system must be verified in order to be sure that error accumulation is within acceptable limits.

In this paper, we will present a design and verification methodology for a frequency domain equalizer based on a combination of simulation and formal verification. This work is an extension of [2], within which we used theorem proving techniques in order to provide the error analysis for an implementation of the frequency domain equalizer based on the Fast Least Mean Square (Fast LMS) algorithm [3,4]. First, we will develop a model for the Fast LMS algorithm specification at the floating-point

* Corresponding author.
E-mail addresses: asouari@ece.concordia.ca (A. Souari), amjad.gawanmeh@kustar.ac.ae (A. Gawanmeh), tahar@ece.concordia.ca (S. Tahar), mohamed.ammari@eniso.rnu.tn (M. Lassaad Ammari).

0026-2692/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.mejo.2013.10.012

number level of abstraction in Simulink [5]. Then we will conduct simulation based verification in order to measure the floating-point signal-to-noise ratio (SNR) error for this model. Next, we will perform a multi-level formal error analysis to show that, when converting from one number domain to another, the algorithm produces the same values with an accepted error margin caused by the round-off error accumulation. We conduct formal error analysis at the floating-point, fixed-point, and real number domains using the high order logic (HOL) theorem prover [6]. Finally, we will provide an implementation for the Fast LMS algorithm in the frequency domain using the System Generator for DSP [7] at the fixed-point number level of abstraction. Overall, error estimation and analysis for both floating-point and fixed-point models are required to show that the error generated in the implementation of the algorithm conforms with the required accuracy of conversion in the equalizer design to operate properly.

The rest of the paper is organized as follows: Section 2 discusses related work. Section 3 describes the frequency domain equalizer implementation based on the Fast LMS algorithms. Section 4 presents our design and verification methodology. In Section 5 we present the Simulink model for the equalizer and simulation based error estimation. In Section 6, the error analysis of the frequency domain equalizer is formalized in HOL. Section 7 presents the System Generator based model of the equalizer and its error estimation. Finally, Section 8 concludes the paper and presents suggestions for future work directions.

2. Related work

The design and implementation of frequency domain equalizers is considered vital since equalization is a fundamental process in modern communication systems. Wang et al. [8] presented an iterative frequency–time domain equalizer for Advanced Television Systems Committee (ATSC). In this approach, the multipath distortion in the signal is first compensated with a frequency domain equalizer on a block-by-block basis. Then, a time domain interference cancelation algorithm is used to eliminate the inter-block and intra-block interference. These steps are repeated until the desired receiver performance, in terms of SNR and symbol error rate, is achieved. In another work, Luzio et al. [9] proposed a pragmatic iterative and non-iterative Frequency-Domain Equalization design for offset modulation methods in order to optimize the design for low-complexity transmitters and efficient power amplification. Dinis et al. [10] designed a frequency-domain equalizer for the receiver system that is optimized for Offset Quaternary Phase Shift Keying. Both Luzio et al. [9] and Dinis et al. [10] adopted bit error rate (BER) as reference for performance evaluation.

Recently, Sobaihi et al. [11] discussed the performance of an orthogonal frequency division multiple access transceiver with Frequency Domain Equalization based on time domain channel estimation. They also provided an implementation on FPGA platform. The authors measured the received signal constellation before and after equalization and used the bit error rate with SNR for their performance evaluation. Mori et al. [12] presented a method to estimate the average block error rate performance of star 32/64QAM schemes employing a frequency domain equalizer in that is designed for orthogonal frequency division multiple access systems. In addition, Komatsu et al. [13] presented an ASIC hardware implementation of a frequency domain equalizer and measured power consumption and BER for their design using simulation. Mehana and Nosratinia [14,15] provided an analysis of a single-carrier frequency domain equalizer for cyclic delay diversity and Alamouti signaling schemes in order to obtain a threshold rate for the full spatial–temporal diversity. Finally, Li et al. [16] presented a sliding window frequency domain equalizer for multi-mode systems which operates on the time-domain received signal.

The equalizer can be used for equalization in multi-mode systems when different waveforms are supported. This technique shows that equalizers are being developed and enhanced at a fast pace, hence, new testing and verification techniques must also be introduced in order to be sure about their correctness. In addition, the design and the applications of the frequency domain equalizer discussed above were concerned with performance parameters of the equalization process in the frequency domain such as SNR, symbol error rate, and block bit error rate. These parameters were estimated using the simulation environment of MATLAB. However, in all aforementioned methods, the error resulting from data conversion between different number domains is never measured or considered in the analysis. This error is due to handling the design and implementation of the equalizer at different levels of abstraction. Regardless of how small the error is, it can be amplified when introduced into an algorithm with repetitive and accumulative nature, such as frequency domain equalizer algorithms. In this work, we consider the frequency domain equalizer from this perspective and handle errors resulted from data conversions for equalizer models at different levels of abstraction.

On the other hand, the use of formal methods in the analysis of errors resulted from manipulating numbers at different levels of abstraction is not new. Harisson [17] used the HOL-Light theorem prover to approximate floating-point algorithms to their mathematical counterparts. He mainly proved that the floating-point exponential function has a correct overflow behavior and when this overflow is absent, the result is linked to a precise error value. In the analysis done by Harisson, the error represented as an independent random variable, is calculated depending on the arithmetic type and the rounding mode. Then, the mean square error is given after performing the error analysis.

Akbarpour [18] continued the work of [17] and proposed an error analysis framework based on theorem proving and dedicated specially to DSP algorithms. The methodology is based on the idea of representing the system in the three domains; the real, the floating-point and the fixed-point. Then, he calculated the error in the transitions from real to floating-point and real to fixed-point. Finally, the error in the transition from floating-point to fixed-point is driven by doing a subtraction between the two types of error calculated before. To show the feasibility of his methodology, Akbarpour applied his technique on digital filters [19] as well as on a 16 point radix 2 FFT [20]. Abu Nasser [21,22] adopted the methodology of Akbarpour [20] to study the error analysis of an FFT–IFFT which is a combination of a 64 point radix 4 FFT and IFFT blocks. Our work is also considered as an application of the formal verification framework developed in [18] since it is dealing with the error analysis of a frequency domain equalizer.

The application we verify in this work is considered more complex and error prone than the design in [22], where there is only a single combination of FFT and IFFT blocks, whereas our system is composed of three FFTs and two IFFTs. In addition, the frequency equalizer is based on arithmetic operations that use numbers of real, floating-point and fixed-point types. Hence, the formalization of error expressions and error analysis we intend to perform on the design is based on a theorem for complex numbers of the above different types. Finally, error analysis for the equalizer requires formalizing input vectors to be able to store various symbols in each iteration of the Fast LMS algorithm.

In summary, the contributions of this work compared to the state of the art are summarized as follows: first, the use of formal methods in the verification process of equalizers, in particular, theorem proving technique. Second, the combination of both simulation and theorem proving in the design and verification of frequency domain equalizers. Third, the ability to handle a frequency domain equalizer at multiple levels of abstraction, starting from the algorithmic level, then the time domain level, then the frequency domain level, and finally the FPGA