Model Based Design Methodology Applied to Software Defined Radio based EW Receiver

Pedro Gonzaga L. Coutinho[1,2] Canisio Barth[1,3] Olympio L. Coutinho[1] Osamu Saotome[1]
[1]Aeronautics Institute of Technology (ITA), São José dos Campos/SP - Brazil
[2]Federal University of Itajubá (UNIFEI)- Brazil
[3]Navy Research Institute (IPqM), Rio de Janeiro/RJ - Brazil

***Abstract*— With the technology advance in analog-to-digital converter (ADC) and digital-to-analog converter (DAC) devices, associated with high speed digital signal processing (DSP) using field-programmable gate arrays (FPGA), modern Electronic Warfare Receiver (EWR) systems design have moved toward digital implementation applying software-defined radio (SDR). In this context, Model Based Design (MBD) emerges as a development methodology to integrate high-level system simulation, automatic hardware description generation, and hardware-in-the-loop (HIL) testing. This paper proposes a concept demonstration of the MBD methodology applied to an Electronic Warfare (EW) radio frequency (RF) recorder triggered by pulse envelope. MATLAB/Simulink models are initially used to simulate the system at a high level. Subsequently, the Xilinx System Generator tool generates and transfers the hardware description to a Xilinx FPGA. Finally, the hardware implementation is validated using HIL technique. Comparative measures between the simulation-only and hardware-in-the-loop versions are presented.**

***Keywords*— Electronic Warfare, Model Based Design, Software Defined Radio**

I. Introduction

Within a modern electromagnetic battlefield, Electronic Warfare Receiver (EWR) develop the important task of intercepting, detecting, and classifying radar emitters in a dense and high frequency range electromagnetic environment [1]. Radar Warning Receiver (RWR), heterodyne scan receiver, instantaneous frequency measure (IFM) and channelized receiver are frequently used to implement EWR [2]. Modern Electronic Warfare (EW) technology has been driving toward flexible and adaptive system design to achieve EW system requirements. The increasing speed of analog-to-digital converter (ADC) and digital-to-analog converter (DAC) circuits, associated to high speed digital signal processing (DSP) using field-programmable gate arrays (FPGA), have driven the modern EWR system design to digital implementation applying software-defined radio (SDR) [3]-[4]. The SDR technology allows to implement different radio receiver functionalities defined by software using a standard hardware. An analog front-end is used to down-convert the radio frequency (RF) incoming signal to feed an ADC followed by a FPGA to perform signal processing. The FPGA is configured using hardware description language (HDL) to perform the remaining receiver functions digitally, such as digital down-conversion, filtering, demodulation and detection [5]. Therefore, it gives the system design team a high degree of flexibility to prototype and test the system. However, frequent hardware changes and tests, especially during the system design phase, make the task of directly programming the FPGA using pure HDL workflows difficult and time-consuming.

Model Based Design (MBD) is a development methodology that makes it easy to perform high-level system simulation for concept validation, automatic HDL generation and hardware-in-the-loop (HIL), all in an integrated environment [6]-[7]. Xilinx Inc. is a semiconductor technology company that supplies a large family of FPGAs used in SDR standard hardware. Together with MathWorks Inc., they offer toolboxes to apply MBD for developing Xilinx FPGA-based SDR systems in a MATLAB/Simulink environment.

This paper proposes a concept demonstration of the MBD methodology to implement an EW RF recorder, triggered by pulse envelope using MATLAB/Simulink. The system is simulated at a high level, and the hardware description is generated and transferred to a Xilinx FPGA. Finally, the system is tested using HIL technique. Comparative measures between the simulation and hardware-in-the-loop implementations are presented.

In Section II, the model based design and FPGA MATLAB/Simulink support tools are introduced. In Section III, the SDR-based RF pulse recorder architecture is presented. Section IV shows the MATLAB/Simulink FPGA hardware description implementation and test, followed by Section V with the conclusions.

II. Model Based Design and FPGA MATLAB/Simulink Support Tools

The main characteristic of the MBD is that it integrates the high-level model between different development phases, as shown in Fig. 1. The method is widely adopted due to its capability of allowing correction of the model errors as they are detected in any of the existing phases.

Using the flowchart in Fig. 1, through research and establishing necessary application requirements, a model design is created using Simulink Block Design. The design is simulated within the MATLAB workspace. If satisfactory results are achieved, it is converted to HDL using the Xilinx System Generator tool. In HDL, the model is implemented on a prototype FPGA for testing. Verification and error corrections are iteratively performed at each stage. Once validated, the description can be transferred to the final hardware.

The Xilinx System Generator is a design tool within the Vivado Design Suite that enables the use of MathWorks Simulink MBD environment for FPGA design. Designs are

P. Coutinho, pedro.coutinho@ga.ita.br; C. Barth, canisio@ita.br; O. Coutinho, olympio@ita.br; O. Saotome, osaotome@ita.br. This work is supported by the FINEP Brazilian Agency (Project 01.22.0581.00) and the Brazilian Air Force.

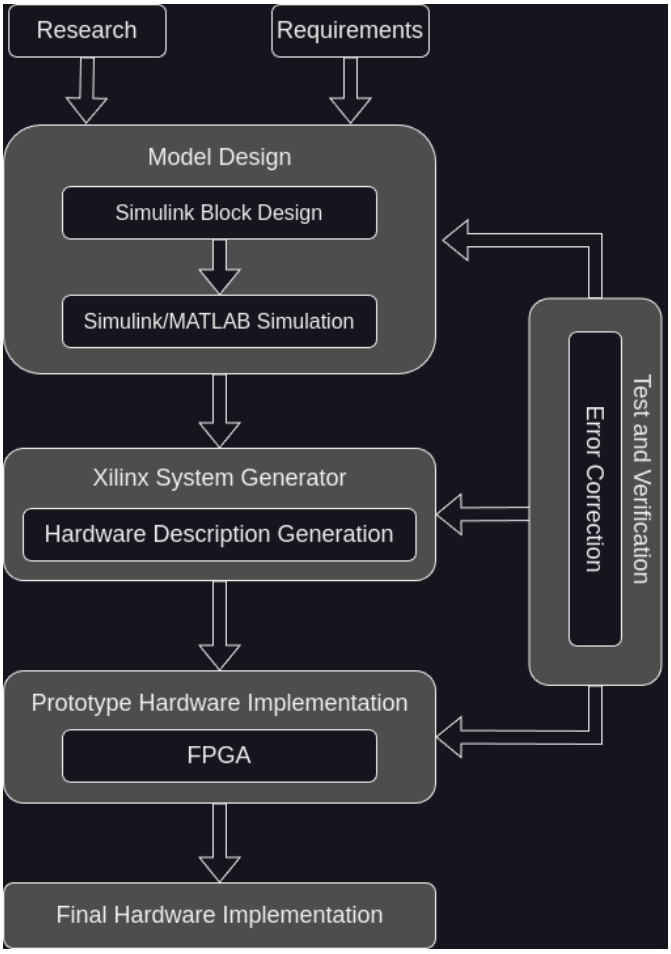

Fig. 1. Model Based Design concept applied to system development.

captured in the Simulink modeling environment using a Xilinx-specific blockset [8]-[9]. This blockset is essential for converting the design into a hardware description for a designated board. Furthermore, the design workflow allows users to interface with the FPGA and perform HIL verification between the hardware and the Simulink environment. Communication between Simulink and the FPGA board is facilitated through either JTAG or Ethernet interfaces.

For the prototype hardware implementation, this work uses the FPGA part of the ZedBoard development kit. This board is an evaluation and development device based on the Xilinx Zynq-7000 Extensible Processing Platform. The ZedBoard's robust mix of on-board peripherals and expansion capabilities makes it an ideal platform for testing and development [10].

After performing iterative error correction at each phase of the design workflow and validating the final model in the HIL simulation, the hardware description is ready to be implemented as a product in the final hardware of the EWR, which can be a ruggedized FPGA or even an application specific integrated circuit (ASIC).

III. SDR-based RF Pulse Recorder

To demonstrate the MBD method in EW applications, we propose a basic system that records RF signals, as shown in Fig 2.

First, a test pulsed signal is generated, which is intended to be recorded only under specific trigger conditions. To achieve

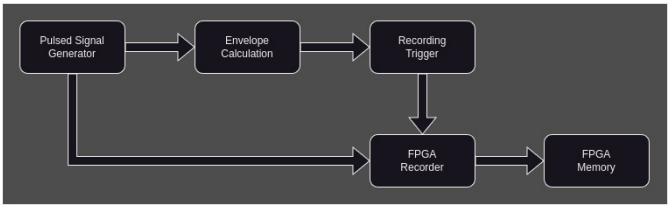

Fig. 2. RF Recorder block diagram.

this, the envelope of the RF signal is calculated from its in-phase (I) and quadrature (Q) components. This is done by summing the squares of the I and Q components and then taking the square root of the result to obtain the envelope value at each sample. Such calculation can be denoted by (1) [11]

$$A(t) = \sqrt{v_i^2(t) + v_q^2(t)}, \tag{1}$$

where $A(t)$ is the envelope calculated in the time instant t, $v_i(t)$ is the in-phase component and $v_q(t)$ is the quadrature counterpart. With this approach, we can trace the envelope value $A(t)$ of the RF signal in a given time window. When this value crosses the predetermined recording trigger threshold within the time window, the FPGA recorder is activated. Recorded values are subsequently stored in the FPGA block memory using a first in, first out (FIFO) scheme. These stored signals are then readily available for further analysis.

IV. MATLAB/Simulink FPGA Hardware Description Implementation and Test

Based on the concepts explained in Sections II and III, a Simulink system was designed to implement the necessary application on the prototype hardware, ZedBoard FPGA. This design utilizes a combination of regular Simulink blocks and System Generator blocks specifically created for FPGA implementation. After validating the system at this level, the tool consolidates it in a single block for use in the HIL analysis. Fig. 3 shows the top-level system where both the Simulink and HIL models can run in parallel for comparison.

Fig. 3 shows an RF signal generator at its top-left corner, generating a pulsed signal with an amplitude of 1 and a period of 0.2, ms. The output of the pulse generator is separated into its I and Q components, which serve as inputs to both the RF recorder Simulink simulation block and the HIL simulation block.

The trigger block represents the current trigger level. For this application, a constant value of 0.8 is used, even though it can be an arbitrary value. The output of the trigger block is set as input for both signal processing blocks, allowing for parallel simulation and comparison of all parameters. Therefore, with these values, at every $0.2\,ms$ a pulse will cross the trigger threshold and activate the RF recorder, recording the input signal in-phase and quadrature parts and finally storing it in their respective FIFO.

Further analyzing the Simulink simulation block, if the recorder is activated, the Xilinx System Generator FIFO blocks will start recording the input signal and store it during the simulation. The recorded can only be visualized if a simulated switch command is given during simulation period. This means that the contents stored inside the FIFO are

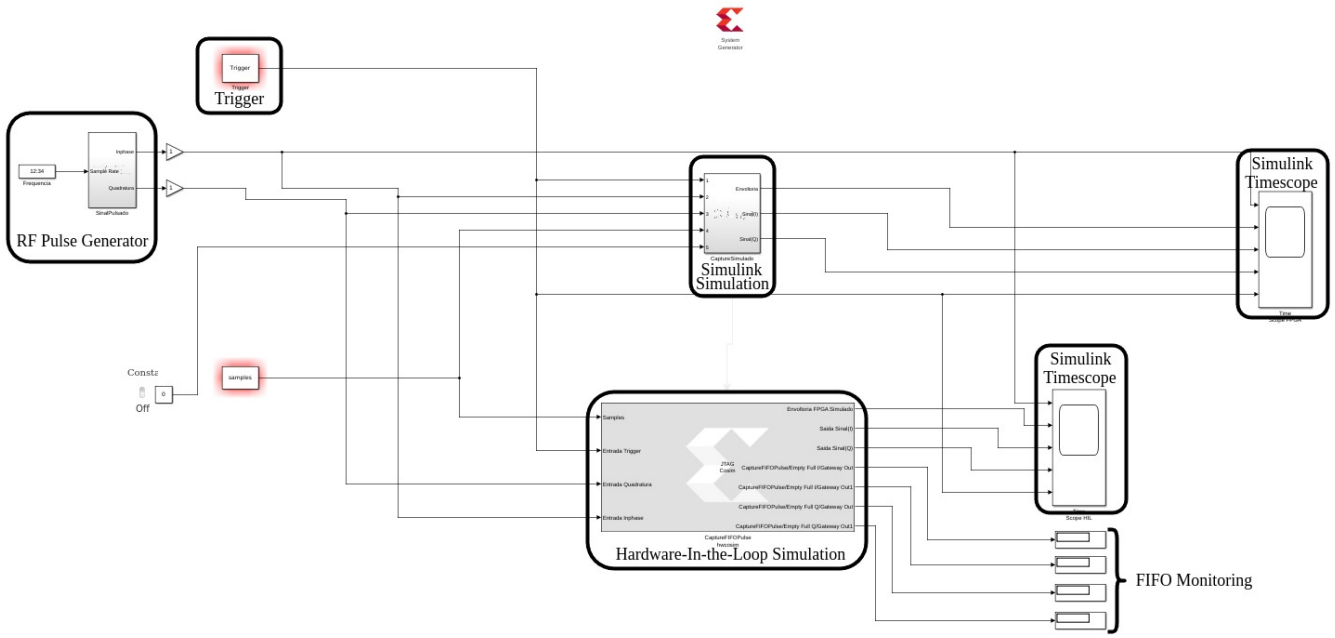

Fig. 3. Simulink Top-level Block diagram of RF Pulse Recorder.

available to be sent to the output of the simulation block and can be shown using a timescope block, as visualized in Fig. 4.

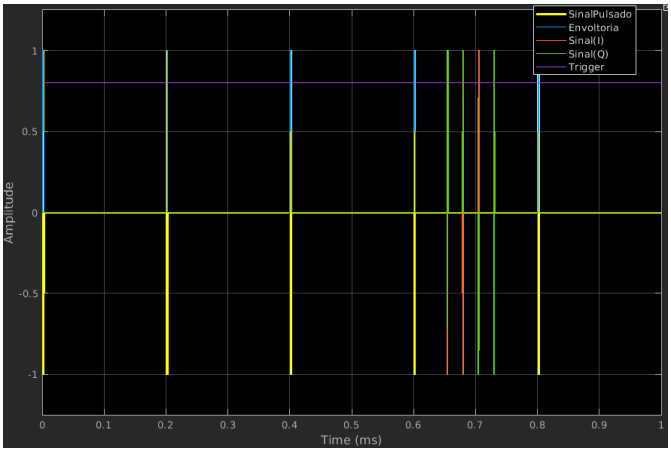

Fig. 4. Simulated RF Pulse Recorder.

Regarding the HIL simulation version, the same concept explained previously can be applied. However, in this case, instead of the Xilinx System Generator blocks operating to produce the resulting data, the Zedboard FPGA is responsible for all the recorder signal processing. In this context, the pulsed signal I and Q components are received by the hardware along with the trigger value. Thus, the FPGA is able to calculate the envelope of a given signal by implementing Eq. 1 in hardware, and subsequently comparing it with the trigger value. If a received pulse crosses the trigger threshold, an onboard FIFO records the received signal, storing it within different simulations during runtime. This enables different modeling setups to be consecutively run without restarting the system, simulating the recording of signals arriving from various sources. To account for the recorder control environment, the values stored are only read from the FIFO if a switch on the board is activated, meaning that the FIFO is allowed to be emptied, sending its stored values back to the Simulink workspace, from where it is possible to view the results on a timescope block, as shown in Fig. 5.

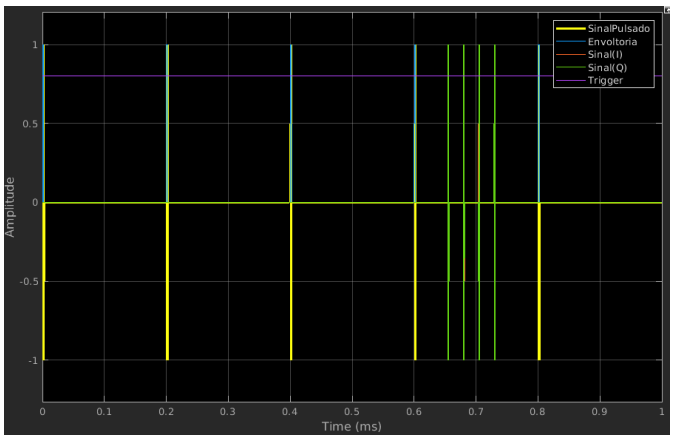

Fig. 5. HIL RF Pulse Recorder.

Both Fig. 4 and Fig. 5 show four pulses, each separated by a period of $0.2\,ms$ and at each one, the trigger threshold is crossed, meaning that in both cases the recorder is activated, capturing the input signal I and Q samples. Furthermore, as seen in the figures, between $0.6\,ms$ and $0.8\,ms$ simulation time, the switch that controls the output of the FIFOs is activated, demonstrating that the recorded signald was available through FIFO reaading when needed.

Another useful feature provided by the Xilinx System Generator is access to post-design products. As the tool creates a project within the Vivado Design Suite, it enables access to post-implementation and post-synthesis reports, similar to what the regular (non-MBD) workflow offers. Using these results, we can analyze various implementation parameters,

such as on-chip power requirements for the application to work effectively and the percentage of logical cells of the FPGA used in the implemented design.

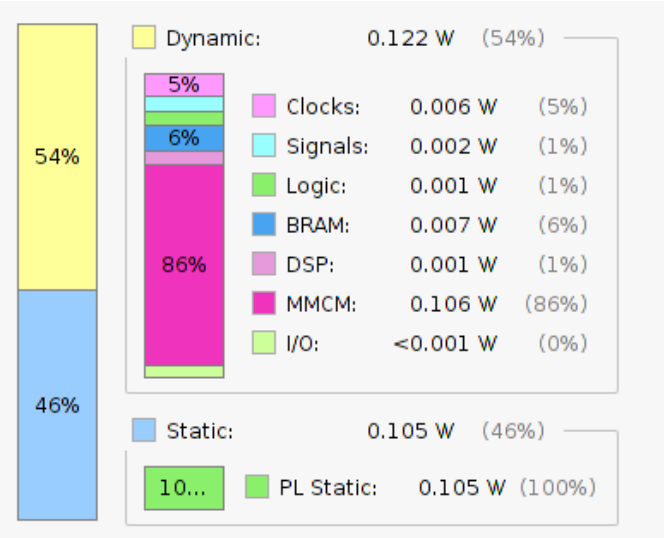

Fig. 6. Power on-chip used for RF Pulse Recorder.

For example, the graph shown in Fig. 6 represents the total on-chip power used for the RF recorder application explained on section IV. The first parameter *Design Dynamic* referred in the figure as *Dynamic* represents the power utilization for the logic parts and switching activity. The second parameter shown as *Static* represents the leakage power when the device is turned on and the power used when the device is first configured. These values show that the onboard power utilization, for the given application, works with a maximum of $0.228\,W$ in total.

Another useful statistic obtained from the post-design products is the Utilization Report, depicted in Fig. 7. Apart from clock management and buffers (*MMCM* and *BUFG*), it is worth noting that the design utilizes a significant amount of *BRAM*, which is the primitive used to implement the FIFOs. look-up table (LUT) and DSP primitives are also heavily utilized, as they serve as the building blocks for logic and arithmetic operations, respectively.

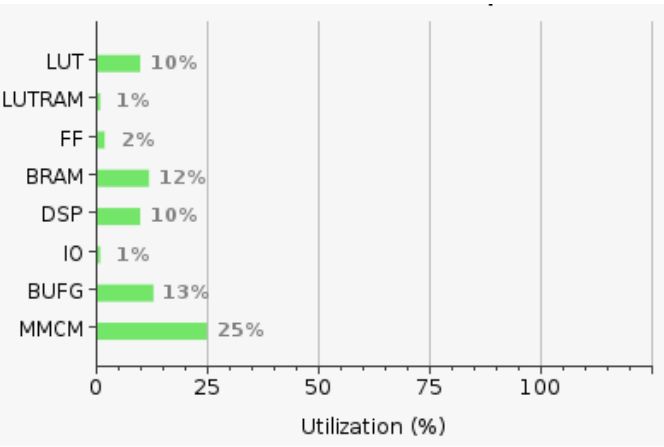

Fig. 7. Chip Utilization RF Pulse Recorder.

V. Discussion and Conclusion

In this article, we presented an analysis of how the Model Based Design (MBD) methodology can be applied to SDR in EW applications. Thus, by following the steps dictated by the MBD, detailed in section II, it is possible to apply the fast design modeling in the context of an EWR. More specifically, in a parameterized RF recorder application. Moreover, in gathering both the concepts of SDR and MBD, satisfactory results were achieved using the support tools available, specifically the Xilinx System Generator and Mathworks tools.

With minimal effort, we were able to generate a hardware description that replicates the simulated conceptualized model on the FPGA. Furthermore, as the tools enabled HIL simulation via JTAG cable communication between the board and Simulink, we were able to compare real-time results from both instances; that is, the Simulink-only and its dedicated hardware version.

Therefore, upon analyzing Fig. 4 and Fig. 5, it is evident that both implementations exhibit a high degree of similarity. This validates the hardware implementation of the control and DSP functions, signaling readiness for the next steps in the MBD flow.

Additionally, as previously explained, these tools not only facilitate rapid development of an SDR-based RF recorder but also offer valuable insights into power consumption and device utilization. This capability enables further optimization of critical applications, such as those found in EW systems.

In the EW context, future work can leverage these tools for various applications, including deinterleaving and direction finding. This enables engineers with relatively limited knowledge in HDL to accelerate their applications using hardware platforms, while keeping problem-solving and error correction contained within the modeling workflow.

References

[1] D. Adamy, *EW 101: A First Course in Electronic Warfare*, ser. Artech House radar library. Artech House, 2001.

[2] S. Robertson, *Practical ESM Analysis*, ser. Artech House electronic warfare library. Artech House, 2019.

[3] R. Poisel, *Electronic Warfare Receivers and Receiving Systems*, ser. Artech House electronic warfare library. Artech House, 2015.

[4] D. Allan, E. Atimati *et al.*, *Software Defined Radio with Zynq Ultrascale+ RFSoC*, 1st ed., L. Crockett, D. Northcote, and R. Stewart, Eds. Strathclyde Academic Media, Jan. 2023.

[5] A. Wyglinski, R. Getz, T. Collins, and D. Pu, *Software-Defined Radio for Engineers*, ser. Artech House mobile communications series. Artech House, 2018.

[6] G. Nicolescu and P. Mosterman, *Model-Based Design for Embedded Systems*, ser. Computational Analysis, Synthesis, and Design of Dynamic Systems. CRC Press, 2018.

[7] A. Romanov and S. Bogdan, "Open source tools for model-based fpga design," in *2015 International Siberian Conference on Control and Communications (SIBCON)*, 2015, pp. 1–6.

[8] Xilinx Inc. (2020) Model Composer and System Generator User Guide UG1483. [Online] Available: https://docs.amd.com/r/2020.2-English/ug1483-model-composer-sys-gen-user-guide. Accessed: Jul 6, 2024.

[9] M. Mekhfioui, A. Satif, O. Mouhib, R. Elgouri, A. Hadjoudja, and L. Hlou, "Hardware implementation of blind source separation algorithm using zybo z7 xilinx system generator," in *2020 5th International Conference on Renewable Energies for Developing Countries (REDEC)*, 2020, pp. 1–5.

[10] Digilent, Inc., *ZedBoard Zynq Evaluation and Development: Hardware User's Guide*, 2012, acesso em: 7 jul. 2024. [Online]. Available: file:///mnt/data/DataSheet.pdf

[11] A. Carlson and P. Crilly, *Communication Systems*. McGraw-Hill Higher Education, 2010.