Microprocessors and Microsystems 41 (2016) 56–66

Contents lists available at ScienceDirect

Microprocessors and Microsystems

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

FPGA based architecture of Extensive Cancellation Algorithm (ECA) for Passive Bistatic Radar (PBR)

Amin A. Jarrah*, Mohsin M. Jamali

Department of Electrical Engineering & Computer Science, The University of Toledo, Toledo, Oh 43607, USA

ARTICLE INFO

Keywords:
Auto-generation
Extensive Cancellation Algorithm (ECA)
Field-Programmable Gate Array (FPGA)
High Level Synthesis Tools (HLS Tools)
Parallel Architecture
Software Tool

ABSTRACT

Passive Bistatic Radar (PBR) exploits existing signals of opportunity from different sources such as Radio and TV signals. Extensive Cancellation Algorithm (ECA) has been proven to be a very effective way to mitigate the effects of direct signal, multipath and clutter echoes in PBR. Also, it is able to detect a moving target accurately when it comes to strong-clutter environment and long-range detection providing evidence for its robustness. However, ECA is a computationally intensive algorithm and will benefit from parallel processing and modern computational platforms such as Field Programmable Gate Arrays (FPGAs). This work involves transformation of ECA by exploring opportunities for parallel processing and elimination of any unnecessary computations and storages. ECA algorithm has been also implemented on FPGAs for high speed computation by exploiting parallel and pipelining approaches. A new software tool called Radar Signal Processing Tool (RSPT) has been developed. It allows the designer to auto-generate fully optimized VHDL representation of ECA by specifying many user input parameters through GUI. The produced VHDL code is independent of FPGA part. It is also appropriate for use with any future high performance FPGAs or ASICs to further cut down computation time. Moreover, it provides the designer a feedback on various performance parameters. This offers the designer an ability to make any adjustments to the ECA component until the desired performance of the overall System on Chip (SoC) is achieved. The computation time of our transformed/optimized algorithm has improved by a factor of 3.8. Its FPGA implementation offers a speed up of 18 over CPU.

1. Introduction

Recently, passive radar has received renewed interest and has many applications [1–8]. In [1] PBR is used to analyze its effectiveness and feasibility for WiFi applications which yielding a remarkable result. It is also used in FM radio band applications which yielding high target detection by increasing Signal to Noise Ratio (SNR) [2]. Bistatic radars have the transmitter and the receiver antennas at separate locations [9–11] as shown in Fig. 1. Conventional monostatic radars on the other hand have the transmitter and the receiver at the same location. Passive radars only have a receiver and they simply listen to transmitted data from other radars or electromagnetic emitters. Passive radars are able to utilize signals of opportunity available in the environment. These signals may be broadcast FM radio, TV signals, mobile phones, and others. They can be very useful in detecting targets without emitting any Radio Frequency (RF) signals of their own. The scattered RF signal can be received by one antenna called surveillance antenna and compared with the received signal from another antenna which is called reference antenna.

Recently, Passive Bistatic Radars (PBR) have received high interest among radar researchers. PBRs have low cost, reduce electromagnetic pollution and the interference with other necessary sources. They also do not need dedicated transmitter and frequency allocation [11]. One drawback of PBR is that transmitted signals are not under the control of the radar designer. The PBR then deals with unknown transmitted RF signals and has a variable structure of the ambiguity function. So, Passive radars do not have luxury of having appropriate ambiguity function and narrow peaks in both range and Doppler [11]. Therefore, PBR requires the use of two correlated passive antennas to collect RF signals in order to detect the desired target. A surveillance antenna steers toward the area that needs to be surveyed. The reference antenna steers toward the transmitter antenna [11]. In order to get a good signal to noise ratio, PBR requires a long integration time for surveillance since the received RF signals are continuous waveforms and may exhibit low reflected power levels [12].

PBR is based on the promise of use of unknown RF transmitted signal [13,14]. It contributes to the following:

* Corresponding author. Tel.: +4199731961.
E-mail addresses: amin.jarrah@rockets.utoledo.edu (A.A. Jarrah), mohsin.jamali@utoledo.edu (M.M. Jamali).

http://dx.doi.org/10.1016/j.micpro.2015.12.003
0141-9331/

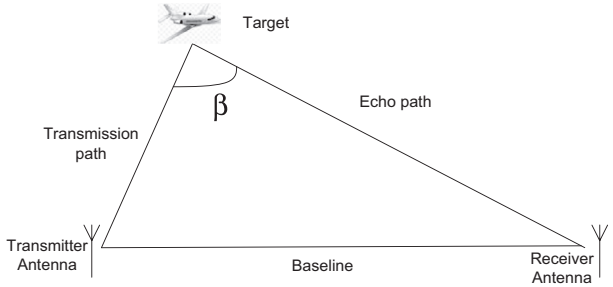

Fig. 1. Bistatic geometry for one pair of transmitter and receiver.

- Strong clutter can mask some targets.
- Small fraction of the direct signal can mask target echoes.
- Strong target echo can mask other echoes from other targets.

The remainder of this paper is organized as follows: Section 2 presents an overview of the state of the art; a background on signal model and reference scenario of ECA is presented in Sections 3 and 4 provides optimized methods to parallelize ECA algorithm, the software package is discussed in Sections 5 and 6 presents synthesis results; Section 7 presents comparisons between HLS tool and VHDL programming; verification of the optimized algorithm is presented in Section 8; and conclusions are given in Section 9.

2. State of the art

A number of researchers [15–21] have used different techniques to overcome the above concerns. For example, an iterative method was used in [17] and [19] to cancel strong multipath and target echoes while remaining the strong targets. In [21], a sequential cancellation algorithm was presented which exploits a variable number of iterations adaptively identified from the measured data itself. However, these algorithms are very time consuming for real data applications and lacking to achieve all the limitations of PBR. Therefore, Colone et al. have proposed Extensive Cancellation Algorithm (ECA) [11] which is a very effective way to mitigate the direct signal, multipath, and clutter echoes in PBR. Also, it is able to detect desired target accurately for strong-clutter environment and long-range detection.

ECA is a computationally intensive algorithm and may not be able to provide target information in real-time. ECA will benefit from parallel processing for achieving real-time requirements. Parallel processing systems may utilize multi-core, Network on Chip (NOC), Field Programmable Gate Arrays (FPGAs), and Graphic Processing Units (GPUs). FPGA contains very large number of logic slices, I/O blocks (IOBs), block memory, on-chip multipliers (DSP48), reconfigurable interconnection logics and other necessary hardware resources. A logic slice is the computational workhorse of the FPGA. It consists of Look up Tables (LUTs) and a flip-flop. Programming software can configure available hardware resources to perform any desired computation. FPGAs offer memory storage in the form of LUTs and block memory. External memory can also be interfaces with FPGAs via DDR3 memory. An efficient synthesizer can choose any of these memories as storage as part of the computational process. FPGA based systems are reconfigurable platform which allow faster system exploration with a high degree of flexibility. Parallel hardware should be efficient in terms of latency, area, power consumption, and flexibility. One drawback of FPGAs is that it requires VHDL programming language. VHDL is perceived to be difficult [22].

However, the design and implementation of complex applications using low level languages such as Verilog and VHDL is challenging. HDL is difficult to learn and far from traditional high level languages. The use of high level languages is easy to learn and modify. They also do not require deep knowledge of computer architecture. Therefore, High-Level Synthesis (HLS) tools [23–26] arise as an alternative to HDLs when using FPGAs.

The main advantage of using HLS tool is to simplify the hardware implementation and accelerates the design process [27]. Altera and Xilinx provide a DSP builder and system generator respectively. They work in conjunction with the Simulink for FPGA design [28]. However, these tools lack the critical step of control logic that is needed in the hardware implementation. Control logic helps in scheduling micro-operations. They also have limited available library elements.

A number of HLS tools [23,29,30] have been developed to move the design effort to higher abstraction levels. This reduces the design time for FPGA based hardware implementation and speed up the verification process. Also, it improves the performance and the design exploration by adopting many optimization techniques. Meeus et al. [31] have compared twelve HLS tools based on many metrics and provided designers with a good overview of current HLS tools. Based on their evaluation, Vivado HLS tool [31] is chosen since it offers the best performance based on their metrics evaluation. However, all tools lack exploitation of any available opportunities of data locality and reduction of memory bandwidth requirements. Also, in all HLS tools [31], even the best ones, exploration and optimization options are still application specific which is the responsibility of the designer.

However, HLS has many limitations where it does not support dynamic memory allocations, recursion, and some restrictions on pointers. HLS does not support certain trigonometric and other mathematical functions and requires the development of customized functions. The HLS compiler is not able to extract all the possible parallelism from the sequential program. The coding style of the input program can drastically influence the end result of the generated design. The designer still needs to be aware of the underlying hardware and uses the proper coding style to arrive at an optimized architecture.

The goal of this work is to minimize computation time and storage resources of ECA by exploring opportunities of any computation and storage that could be eliminated. Parallel processing is also used to achieve the real-time constraints. Data locality has also been improved in our implementation by exploiting any opportunity of optimization. Moreover, a new software tool called Radar Signal Processing Tool (RSPT) has been developed. It will allow the designer to auto-generate fully optimized VHDL representation for any radar signal processing algorithm. This work focuses on development of FPGA based hardware for real-time execution of Extensive Cancellation Algorithm (ECA) [11].

The RSPT allows the designer to specify user input parameters through a Graphical User Interface (GUI). This offers great flexibility in designing an ECA component for a SoC without having to write a single line of VHDL code. No implementation has been reported for ECA component architectures in FPGA in the literature. Moreover, RSPT provides the designer a feedback on various performance parameters such as occupied slices, maximum frequency, and dynamic range. This offers the designer an ability to make any adjustments to the ECA component until the desired performance of the overall System on Chip (SoC) is satisfied. The tool will utilize optimization techniques such as pipelining, code in-lining, loop unrolling, loop merging, and dataflow techniques to improve throughput and latency.

3. Signal model and reference scenario of Extensive Cancelation Algorithm (ECA)

An example of a PBR geometry for detecting and locating desired target is shown in Fig. 2. ECA needs two separate antennas; the reference antenna steered toward the transmitter and the