

An Approach of Generating and Matched Filtering of Modulated Radar Signals Using the FPGA

Goran Antonijević¹⁾
 Dragan Nikolić¹⁾
 Zvonko Radosavljević¹⁾
 Dejan Ivković¹⁾

The process of generating and matched filtering of three types of modulated radar signals in baseband, using the FPGA Zedboard, is described in this paper. Binary phase sequence, linear frequency modulated (LFM) sequence and Costas sequence was generated using VHDL programming language. In order to perform a comparative analysis of these processes, named sequences had equal values of time-bandwidth product, pulse duration, pulse repetition of frequency and sampling frequency. The process of realization of the matched filtering has been explained using the System Generator program. To confirm efficiency, results of matched filtering of the aforementioned sequences were presented together, from the Matlab program and obtained using the FPGA Zedboard. Percentage values of utilization of resources of the FPGA Zedboard were presented, of the each coding sequence particularly

Key words: radar signal, signal modulation, signal filtering, signal generating, digital signal processor, programming language.

Introduction

IN the past, for the implementation of digital processor for radar signals in real time, usually it was necessary to create dedicated computers, using thousands of high performance integrated circuit. Digital technology has progressed, so programming of these processors, their development and modification is much easier. The invention of Field Programmable Gate Array (FPGA) represents a revolution in the development of digital signal processing in real time. FPGAs are integrated circuits from a group of programmable logic circuits PLD (Programmable Logic Devices). The electrical circuits are usually designed in Hardware Description Language (HDL), such as VHDL (VHSIC Hardware Description Language) or Verilog. Other software tools convert this higher-level hardware description into a file which is sent to FPGA and defines how it has been configured.

In this paper, for generation and processing of radar signals, FPGA Zedboard was used. The ZedBoard is an evaluation and development board based on the Xilinx Zynq™-7000 All Programmable SoC (AP SoC) and can be targeted for broad use in many applications. For example in [1] the steganography algorithm implemented on ZedBoard using Vivado HLS tool was presented. In that paper Vivado is used to create the hardware system and software development kit (SDK) is used to create an application to verify the hardware functionality. In [2] the development of a system for automatic number plate recognition of Colombian vehicles was described. That system is based on an embedded system with mixed processing capability, specifically using the ZedBoard, which includes a Zynq 7000 Xilinx device. In [3] a model of FPGA implementation of prototype noise jammer using pulsed noise jamming techniques to jam LFM with

adequate power was introduced. Furthermore a complete LFM PC radar model was introduced for jammer evolution purpose. A design and implementation method of radar time domain waveform based on FPGA was presented in [4], in order to achieve the waveform library building, and agility in the transmitter of cognitive radar system.

In this paper the possibility for generating of binary phase sequence, linear frequency modulated (LFM) sequence and Costas sequence using the VHDL programming language on FPGA Zedboard was analyzed. Besides, the goal it was to perform matched filtering of the aforementioned sequences using the program System Generator. The efficiency of implementing of matched filtering was confirmed by comparing the results obtained using the FPGA Zedboard, with the results from the Matlab program. At the end the utilization of resources of the FPGA Zedboard was analyzed which were used for generating and matched filtering.

Radar signals processing

Choosing a particular waveform type and a signal processing technique in a radar system heavily depends on the radar's specific mission and role. Range and Doppler resolutions are directly related to the specific waveform frequency characteristics. Thus, knowledge of the power spectrum density of a waveform is very critical. In general, signals or waveforms can be analyzed using time or frequency domain techniques.

Pulse compression

It is known that the maximum range of radars, with optimally processing of receiving signals and certain spectral noise density, depends exclusively on the total energy of signal and it is independent of the shape of the waveform. This allows

¹⁾ Military Technical Institute (VTI), Ratka Resanovića 1, 11132 Belgrade, SERBIA
 Correspondence to: Goran Antonijević; e-mail: kadet.anton@gmail.com

the selecting of shape of the signal, for the given signal energy, in order to optimize another radar characteristic, for example the range resolution. On the other hand, the range resolution depends exclusively on the width of the frequency band. The average transmitted power is directly linked to the receiver, based on the Signal to Noise Ratio (SNR). It is desirable to increase the pulse width while simultaneously maintaining adequate range resolution. This can be made possible by using pulse compression techniques [5].

In this paper, implementation of three types of modulated radar sequences on FPGA Zedboard, will be thoroughly analyzed. Those are binary phase sequence, LFM sequence and Costas sequence.

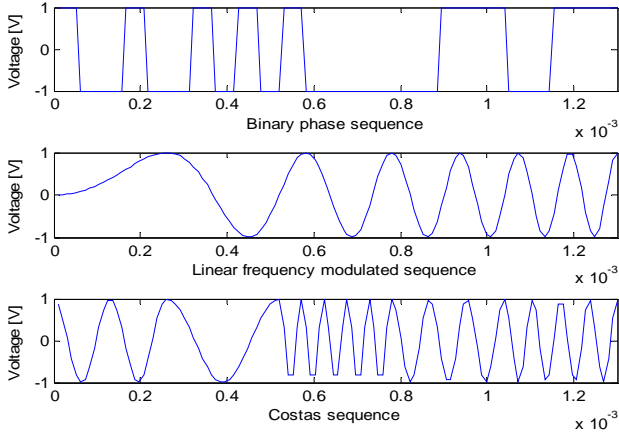


Figure 1. Used modulated radar signals in baseband

In Fig.1 modulated radar signals in baseband are presented, which are generated in this paper. The main parameters and characteristics of those signals are presented in Table 1.

Table 1. Parameters of the used sequences

Parameters of radar signal	Binary phase sequence	LFM sequence	Costas sequence
Pulse duration [ms]	1.28	1.28	1.28
Number of sub-pulses	25	/	5
TB product	25	25	25
Number of samples in one pulse	125	125	125

Based on the data in Table 1, we can conclude that using the all three types of aforementioned sequences it will be obtained the same range resolution.

Matched filtering

The impulse response of a matched filter is defined by the particular signal to which the filter is matched. Matching will result in the maximum attainable SNR at the output of the filter when signal time-inverted related to original, to which it was added white noise, are passed through it. In radar applications, SNR is of paramount importance, and matched filters are used extensively. The probability of detection is related to the SNR rather than to the exact waveform of the signal received. The input to the matched filter is the signal $s(t)$ and additive white Gaussian noise with a two-sided power spectral density of $N_0/2$. Obtaining of the impulse response $h(t)$ that will yield the maximum output SNR at a predetermined delay t_0 was explained in [6]. We obtain the impulse response $h(t)$ that will maximize SNR:

$$\left(\frac{S}{N}\right)_{out} = \frac{|s_0(t_0)|^2}{n_0^2(t)} \quad (1)$$

Let the Fourier transform of $s(t)$ be $S(\omega)$, then the output signal at t_0 is given by:

$$s_0(t_0) = \frac{1}{2\pi} \int_{-\infty}^{\infty} H(\omega) \cdot S(\omega) \exp(j\omega t_0) d\omega \quad (2)$$

The mean-squared value of the noise, which is independent of t is:

$$\overline{n_0^2(t)} = \frac{N_0}{4\pi} \int_{-\infty}^{\infty} |H(\omega)|^2 d\omega \quad (3)$$

After adding the values (3) in (2), and after the execution explained in [6], the maximum output SNR it was obtained when:

$$H(\omega) = K \cdot S^* \exp(-j\omega t_0) \quad (4)$$

In [6] it was derived that maximum SNR is $2E/N_0$. The impulse response is linearly related to the time-inverted complex-conjugate signal, which is described by autocorrelation function (ACF). When the input to the matched filter is the correct signal plus white noise, the peak output response is related to the signal's energy.

In order to emphasize peak-side lobe ratio of sequences used in this paper, it will be presented ACF in decibels.

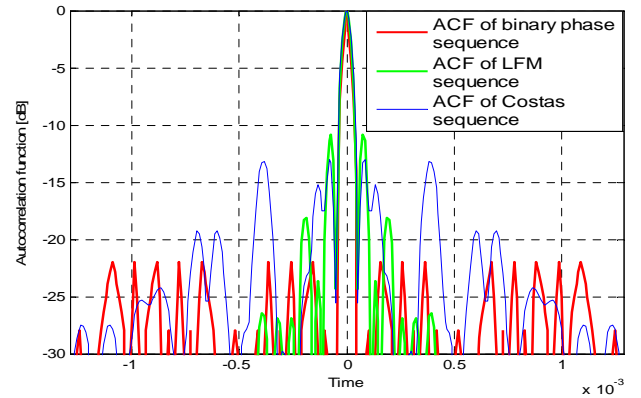


Figure 2. ACF of used radar sequences

Fig.2 shows that the best peak-sidelobe ratio is obtained using binary phase sequence with 25 elements, and amounts 22 dB. Besides, it is important to note, that binary phase sequence has the same level of all sidelobes, unlike others two sequences.

Hardware and software platform

FPGA Zedboard

The FPGA ZedBoard, which was used in this paper, represents Zynq-7000 FPGA evaluation and development kit, and it is equipped with the reconfigurable devices. Zynq system on chip (SoC) integrates a Xilinx 7-series FPGA and ARM Cortex-A9 dual core based processor system on same chip together. ZedBoard has coherent multiprocessor support, three watchdog timers, one global timer, and two triple-timer counters. In Zynq-7000 FPGA the parts containing intense computations are performed on FPGA. The control parts not containing computations can be done on the processor by using software.



Figure 3. The FPGA Zedboard

Hardware platform configuration for Zynq device consist two parts Processing System and Programmable Logic with 85000 cells. The intercommunication between two different portions of electronic circuits is described as Programmable Logic. The upper basic portion, called the processing system (PS) works like traditional processor. It is mainly formed by the ARM Cortex-A9, DDR3 controller for external DDR3_SDRAM memory and UART for serial communication. Programmable logic part contains the structures of standard FPGA [7].

The FPGA Zedboard perform signal processing in digital form, and from that reason it was necessary to perform digital to analog conversion, in order to display of output signal. For that operation two-channel 12-bit D/A converter Pmod DA2 was applied, which was connected on PMOD connector JB. It is important to note that the sampling rate of this D/A converter is 16.5 MSa, and that represents the limitation for displaying of short signals with high sampling frequency, such as radar signals.

In order to observe generating and processing of obtained signals, it was used Analog Discovery instrument, in oscilloscope mode. It was possible to record results of matched filtering and compares it with results obtained in program Matlab, using Digilent's software for this instrument.

Vivado Design Suite

Vivado Design Suite is a software suite produced by Xilinx for synthesis and analysis of HDL designs, superseding Xilinx ISE with additional features for system on a chip development and high-level synthesis. Vivado represents a ground-up rewrite and re-thinking of the entire design flow (compared to ISE). Vivado enables developers to synthesize (compile) their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. Vivado is a design environment for FPGA products from Xilinx, and is tightly-coupled to the architecture of such chips, and cannot be used with FPGA products from other vendors. Xilinx recommends Vivado Design Suite for new designs with Ultrascale, Virtex-7, Kintex-7, Artix-7, and Zynq-7000.

Binary phase sequence, LFM signal and Cotas sequence are in detail explained in Section 2. Process of generating of aforementioned radar signals in baseband is performed in VHDL programming language. Firstly the clock signal which activate D/A converter, was generated and time between two rising edges of this signal represents sampling time of output radar signal. For generation of this clock signal it was used

fabric clock signal of the frequency 100 Mhz, whose contact is on the J9 pin on the FPGA Zedboard.

For each sequence particularly ROM memory was generated, which has 125 locations of 12 bits, as described in [9]. Dimensions of the memory corresponds to the number of samples of radar signal as well as the number of bits of D/A converters. In mentioned ROM memory it was performed writing the 125 values of modulated radar signals in baseband. Data type, which have been written in ROM memory were signed, in order to proper presentation of positive and negative values of baseband signals. That was performed using the complement of two for negative values of signal.

Signal generator in the form of infinite loop, it was defined in VHDL programming language, with defined values of signal counter. For the values of counter signal between 0 and 124 it was performed the reading of the values of the signal from the particular addresses from ROM memory. For the values between 125 and N, value of output signal was zero, in 12-bit form, where N corresponds to values of pulse repetition interval. After that, the values of the signal counter are resets and the entire procedure is repeated.

System Generator

System Generator provides a powerful high level modeling environment for DSP systems, and consequently is widely used for algorithm development and verification on the Simulink platform. It provides a graphical environment for creating and modeling dynamical systems. System Generator consists of a Simulink library called the Xilinx Blockset and software to translate a Simulink model into a hardware realization of the model. System Generator maps system parameters defined in Simulink (e.g. as mask variables in Xilinx Blockset blocks), into entities and architectures, ports, signals, and attributes in a hardware realization. In addition, System Generator automatically produces command files for FPGA synthesis, HDL simulation, and implementation tools. According that, users can work entirely in graphical environments in going from system specification to hardware realization [8]. The realization of matched filtering was the most important step in this procedure. Matched filter was realized like typical fir filter and number of coefficients was same as number of samples of transmitted signal. The impulse response was time-inverted version of transmitted signal. On the basis of this, on the first position of matched filter coefficient, it was written the last value of the sample of transmitted signal, etc. It is important to note that for realization of matched filter it was used Xilinx blocks (adder, line delay and multiplier). For representation of true values of coefficients in Xilinx multipliers it was necessary to specify the adequate number of bits for the integer value of the coefficients, as well as for the decimal values of the coefficients. In order to analyze usage of capacity of FPGA Zedboard, minimum required number of bits it was defined, for each of the coefficients, representing the true values of coefficients in four decimal places. It is necessary to emphasize that the voltage levels of output connectors PMODS, on FPGA Zedboard, are between 0 and 3.3V. This was an aggravating factor for displaying autocorrelations of the radar signals, whose values have positive and negative values. For this reason after the matched filtration, the attenuation and the absolute value was applied, of the obtained signal.

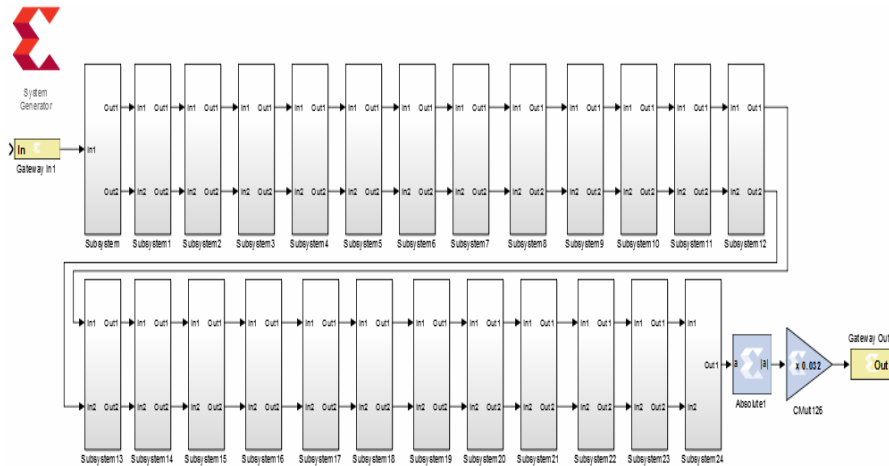


Figure 4. Block system of matched filtering in program System Generator

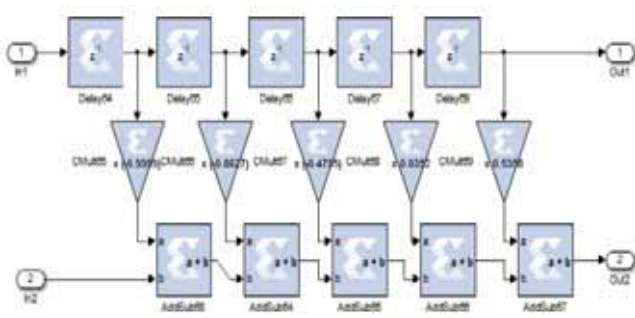


Figure 5. The content of one subsystem from previous figure

In Figures 4 and 5 the realization of the matched filtering was presented, in System Generator program using Xilinx blocks.

Results of signal processing

Results of matched filtering of radar signals using the FPGA Zedboard

The most important part of this paper is an application of analysis and confirmation of validity of results obtained using FPGA Zedboard. That will be realized performing comparison between results from the program Matlab, and results which were obtained by FPGA Zedboard, using instrument Analog Discovery in oscilloscope mode.

As stated in subsection 4.1, it was generated pulse train of modulated radar pulses, which were brought on matched filter. On the output is expected a series of absolute values of autocorrelation peaks. In order to highlight the shape of autocorrelations on the output, an enlarged picture of one autocorrelation will be presented.



Figure 6. The autocorrelation of binary phase sequence obtained using the FPGA Zedboard

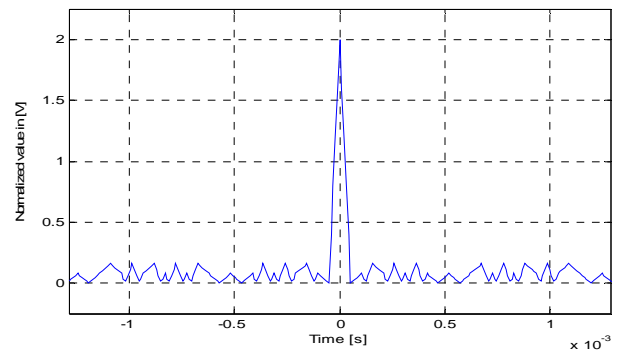


Figure 7. The autocorrelation of binary phase sequence obtained using program Matlab

In Figures 6 and 7 the autocorrelations of binary phase sequence with 25 elements were displayed.

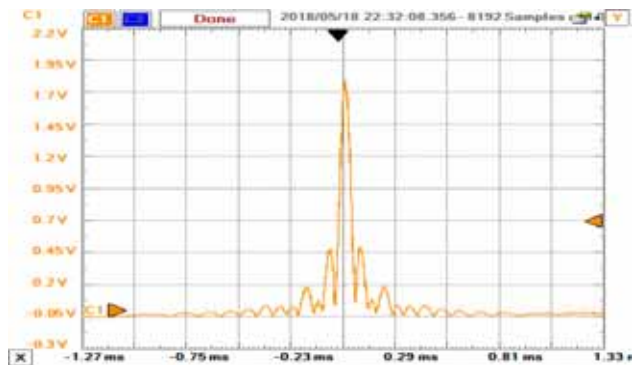


Figure 8. The autocorrelation of LFM sequence obtained using FPGA Zedboard

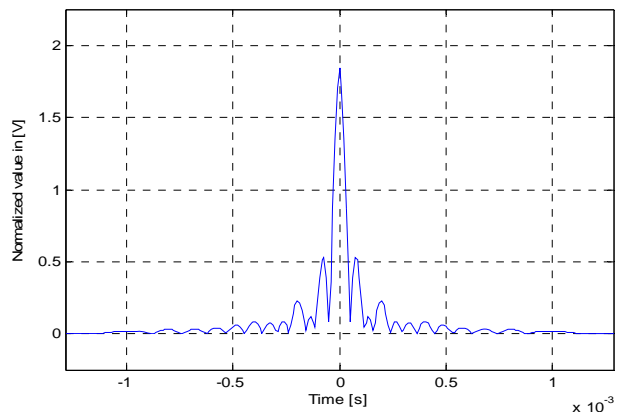


Figure 9. The autocorrelation of LFM sequence obtained using program Matlab

In Figures 8 and 9 the autocorrelations of LFM sequence were displayed.

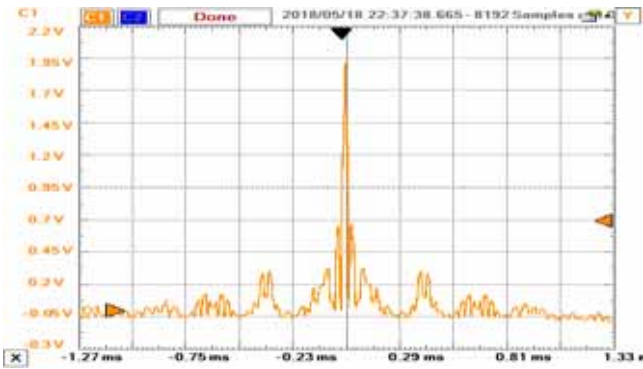


Figure 10. The autocorrelation of Costas sequence obtained using FPGA Zedboard

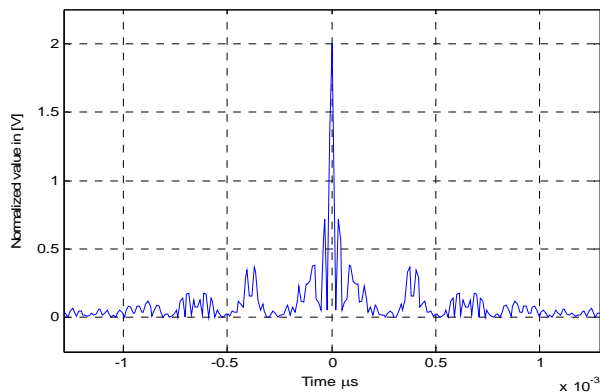


Figure 11. The autocorrelation of Costas sequence obtained using program Matlab

In Figures 10 and 11 the autocorrelations of Costas sequence with 5 elements were displayed.

Influence of frequency of clock signal on displaying of output results

It is important to note that pulse duration and sampling period, in the previous examples, were adjusted to sampling rate of D/A converter which was 16.5 MSa. This value of sampling rate is limiting factor in this paper, and significantly influences on output results. Sampling period used in previous examples was 512/50MHz, and if we take into account that it was 125 samples in transmitted pulse, it can be easily calculated that pulse duration it was 1.28 ms.

$$f_{\text{sample rate of DAC}} \gg 2f_{\text{sampling frequency of the signal}} \quad (5)$$

The condition (5) it was satisfied so we could to faithfully display the ACF.

It is known that the duration of true radar pulses is usually much shorter. In order to obtain a shorter pulse, it is necessary to set a shorter sampling period, because the number of samples is fixed. Setting the higher sampling rate, we are approaching the boundary value for particular D/A converter, and that increases the probability for degradation of correctness of shape of autocorrelation function.

An analysis for three different values of the sampling period it was performed, for LFM sequence. Output results are compared.

It can be noted that the values of the sampling period represents integer multiplication of two, because this process was realized in VHDL programming language using binary numbers. In Figure 12 it can be seen that with a decreasing of

values of sampling period of transmitted pulse, a quality of obtained signal is reduced, because of limited value of sampling rate of used D/A converter.

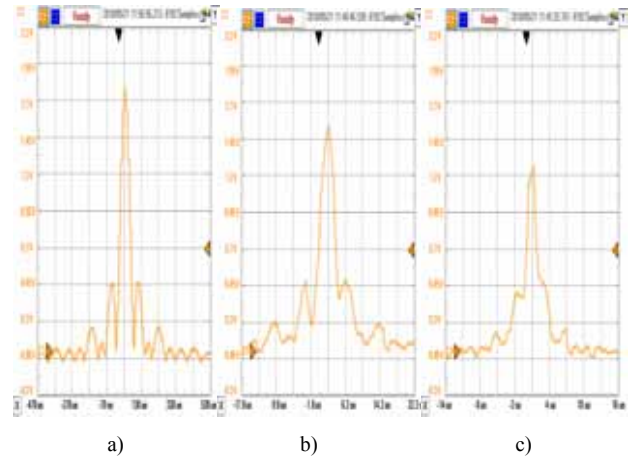


Figure 12. The ACF of LFM sequence obtained using FPGA Zedboard for values of sampling period a) 512/50MHz, b) 32/50MHz, c) 16/50MHz

Analysis of utilization of resources of the FPGA Zedboard

After completed analysis and displaying results of matched filtering of three types of radar signal in baseband, it was necessary to analyze utilization of resources of FPGA Zedboard. Namely, binary phase sequence, LFM sequence and Costas sequence represents different types of radar signal in baseband and their implementation requires different resources.

FPGA Zedboard consist some types of hardware capacities, which have different role in signal processing (Look Up Tables, DSP blocks, Registers, Memories etc). Every of aforementioned types of modulating signals are specific and takes a different amount of those capacities.

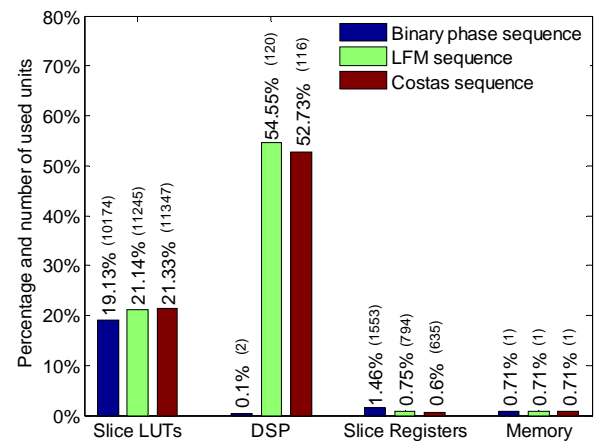


Figure 13. Overview of utilization of resources of the FPGA Zedboard for each used sequence particularly

It was important to analyze displayed percentage values of utilization of resources of the FPGA Zedboard. Every of aforementioned sequences has the same number of samples in one pulse as well as elements in matched filter. From that reason there are similar values for utilization of Slice LUTs, Slice Registers and Memories. Small differences between those values exist because different number of bits was used for faithfully presentation of values of coefficients, for each sequence particularly. The largest utilization of resources and the biggest difference between mentioned sequences it was at DSP blocks. These blocks were used for calculation of output values of matched filtration. FPGA Zedboard contains in total

220 DSP blocks. The main reason for this difference is because the binary phase signal in baseband consist only two values (-1 and 1), and other two mentioned sequences consist 2^{12} values in range between -1 and 1. For presentation values 1 is required only one bit, but for presentation some values of LFM or Costas sequence sometimes it was required even 23 bits. Calculation with one-bit numbers were not demanding, unlike computing with a larger number of bits, and from that reason there are small utilization of DSP blocks at binary phase sequence.

Conclusion

In this paper we confirmed the possibility of generating and matched filtering of modulating radar signals in baseband, using the FPGA Zedboard and D/A converter DA2, but with certain limitations. From all mentioned sequences, which give a same range resolution, it stands out binary phase sequence, which has the best peak-sidelobe ratio. The implementation of binary phase sequence takes up considerably less capacity in relation to LFM sequence and Costas sequence. In the future researches for generating and processing of radar signals, it is necessary to use FPGA boards which have symmetric output values on PMODS connectors and with higher hardware capacity (more DSP blocks). Beside that, it is necessary to use D/A converter with considerably higher sampling rate, in order to preserve the shape of the output signal.

References

- [1] KAPGATE, A.R., AGRAWAL, S.S.: *Edge Based Data Embedding Steganography Algorithm Using Zedboard*, Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017 2nd IEEE International Conference on, Bangalore, 2017.
- [2] CALDERON, J.A.F., VARGAS, J.S., PÉREZ-RUIZ, A.: *License plate recognition for Colombian private vehicle based on an embedded system using the ZedBoard*, Robotics and Automation (CCRA), IEEE Colombian Conference on, Bogota, COLOMBIA, 2016.
- [3] SAMY, T.M., ABDEL-LATIF, M.S., ELGAMEL, S.A., AHMED, F.M.: *FPGA Implementation of Pulsed Noise Interference against LFM Radar*, Computer Engineering and Systems (ICCES), 12th International Conference on Cairo, EGIPAT, 2017.
- [4] LEI, Z., YONG, L., WEI, C.: *Design of Radar Variable Transmit Waveform Based on FPGA*, Signal Processing, Communications and Computing (ICSPCC), 2017 IEEE International Conference on, Xiamen, 2017.
- [5] MAHAFZA, B.R.: *Radar Systems Analysis and Design Using MATLAB*, chapter 3, Chapman & Hall, New York, USA, 2000.
- [6] LEVANON, N., MOZESON, E.: *Radar signals*, chapter 7, John Wiley & Sons, New Jersey, USA, 2004.
- [7] ZedBoard Booting and Configuration Guide, Avnet, USA, April 2012.
- [8] System Generator for DSP Getting Started Guide, Xilinx, USA, UG639, 2 December 2009.
- [9] BROWN, S., VRANESIC, Z.: *Fundamentals of Digital Logic with VHDL Design*, Third edition, Chapter 10, McGraw-Hill, New York, USA, 2009.

Received: 08.05.2018.

Accepted: 12.10.2018.

Jedan pristup generisanja i prilagodene filtracije moduliranih radarskih signala upotrebom FPGA

U ovom radu je objašnjen proces generisanja i prilagodene filtracije tri tipa moduliranih radarskih impulsa upotrebom FPGA ploče. Bifazna kodovana sekvenca, Linearno frekvencijski kodovana sekvenca i Kostasova sekvenca su generisane u programskom jeziku VHDL. U cilju primene komparativne analize ovog procesa, navedene sekvence su imale jednake vrednosti TB proizvoda, trajanja impulsa, periode ponavljanja impulsa i frekvencije semplovanja. Objasnjen je proces prilagodene filtracije u programu System Generator. U cilju potvrde efikasnosti, rezultati prilagodene filtracije, ranije navedenih sekvenci dobijeni u programu Matlab i dobijeni pomoću FPGA Zedboard, su prikazani zajedno. Prikazane su i procentualne vrednosti utrošenih resursa na FPGA Zedboard za svaku kodovanu sekvencu pojedinačno.

Ključne reči: radarski signal, modulacija signala, filtriranje signala, generisanje signala, digitalni procesor signala, programski jezik.

Один подход к генерации и настройке фильтрации модулированных радиолокационных сигналов с использованием FPGA

В этой работе объясняется процесс генерации и настройки фильтрации трёх типов модулированных радиолокационных импульсов с использованием платы FPGA. Двухфазная кодированная последовательность, последовательность кода линейной частоты и последовательность Костаса генерируются на языке программирования VHDL. Чтобы применить сравнительный анализ этого процесса, указанные последовательности имели одинаковые значения ТБ-продуктов, длительности импульсов, периодов повторения импульсов и частоты дискретизации. Объясняется и процесс применения пользовательской фильтрации в программе System Generator. Для подтверждения эффективности результаты пользовательской фильтрации, более ранние последовательности, полученные в программе Matlab и полученные с помощью FPGA Zedboard, показаны вместе. Процентные значения затраченных ресурсов на FPGA Zedboard для каждой кодированной последовательности показаны отдельно.

Ключевые слова: радиолокационный сигнал, модуляция сигнала, фильтрация сигнала, генерация сигнала, процессор цифровых сигналов, язык программирования.

Une approche à la création et au filtrage adoptés des signaux radar modulés par l'emploi de FPGA

Le processus de la création et du filtrage adoptés des trois types des impulsions radar modulées par l'emploi de la plaque FPGA est expliqué dans ce travail. La séquence biphasée codée, la séquence codée linéairement ainsi que la séquence Kostas ont été générées dans la langue de programmation VHDL. Pour appliquer l'analyse comparative de ce processus les fréquences citées étaient de même valeur que les produits TV, la durée des impulsions, les périodes de répétition des impulsions et des fréquences des sembles. On a expliqué le processus de l'emploi du filtrage adopté dans le programme « System generator ». Afin de confirmer l'efficacité, les résultats du filtrage adopté, les séquences déjà citées obtenus par le logiciel Matlab et par FPGA Zedboard ont été présentés ensemble. On a présenté aussi les valeurs en pourcents des ressources utilisées dans le FPGA Zedboard pour chaque séquence codée.

Mots clés: signal radar, modulation de signal, filtrage de signal, création de signal, processus digital de signal, langue de programmation.