SYMDAS – Raw Data Simulator for the Radar with Electronically Controlled Beam

Gustaw Mazurek

Abstract—**The subject of the paper is the software simulator of baseband signals received in a radar. These synthetic signals are sent to the DSP unit responsible for target detection and estimation. The software has been developed in Warsaw University of Technology as a part of R&D project, and its main purpose is to allow flexible testing and validation of the algorithms running in DSP processors of the new radar.**

Keywords—**Radar simulation, MTD, MTI, electronically controlled beam, DSP.**

I. INTRODUCTION

MODERN radar designs employ sophisticated algorithms of target detection and estimation. However, testing of such algorithms is a very complex and time consuming task. Moreover, it requires a huge amount of data to be transferred to DSP parts, like it is performed in real operating conditions.

One approach to this problem is based on field tests of the radar with artificial targets. Unfortunately it is a very expensive work and requires all the hardware to be fully functional. The other approach uses simulated data instead of real signals generated by RF subsystem. It allows to extensively test the detection and estimation algorithms a long time before the complete radar hardware is built. Obviously, the costs of this method are much lower.

As a part of R&D project, in Warsaw University of Technology we design detection and estimation algorithms for the radar demonstrator working in S-band with electronically controlled beam. These algorithms are implemented in DSP processors. For the purpose of development and verification the algorithms, a raw data simulator has been designed. The simulator runs in Matlab (or Octave) environment and generates synthetic raw data similar to that obtained in the output of the receiver baseband output. The software does not need installation of any toolboxes and has been tested both in Windows or Linux operating systems.

A. Structure of Designed Radar

Fig. 1 we can see the simplified block diagram of the radar that is designed. The antenna array consists of several elements combined in order to synthesize the required directional characteristic. In horizontal plane, the characteristic is fixed and consists of only one pencil beam. The scan in azimuth is performed by mechanical rotation of the antenna. In vertical plane, the characteristic is electronically controlled, therefore

G. Mazurek is with Warsaw University of Technology, Institute of Electronic Systems, Nowowiejska 15/19, 00-665 Warsaw, Poland (e-mail: g.mazurek@ieee.org).

no mechanical movement in this direction is required. The characteristic in vertical plane is composed of 4 transmitter beams and several receiver sub-beams. For every transmitter beam, there are three receiver sub-beams, which are processed separately in dedicated DSP blocks.

The received RF signal is amplified, filtered, downconverted and digitized. The resulting I/Q baseband signal from selected sub-beams is fed into three DSP blocks. These blocks are responsible for target detection and estimation, and they consist of several Analog Devices TigerSHARC DSP processors. The detection is performed in bank of MTD filters and, in some cases, also in MTI filters. The results of detection and estimation are transmitted via Gigabit Ethernet to the tracking computer that also provides user interface.

The radar is intended to detect airborne targets and it can operate in seven different modes, each of them optimized for different application. Therefore, the DSP algorithms, and also the simulator, must be compatible with all the working modes.

B. Features of the Simulator

The simulator SYMDAS is intended to replace all the hardware subsystems placed in the left from the dashed line in Fig. 1: the antenna, beamformer, transmitter and receiver circuits. It provides synthetic I/Q baseband signals in a form required by the DSP blocks. The main features of the simulator are:

- simulation of the moving targets,
- generation of the signals with the specified time schedule,
- generation of target echoes scaled with proper antenna characteristics,
- simulation of the AWGN noise,
- quantization of the received signals,
- arranging of the data and headers in the format required by the DSP processors.

The selection of the working mode, along with a setup of all the radar parameters, is performed in Matlab script files.
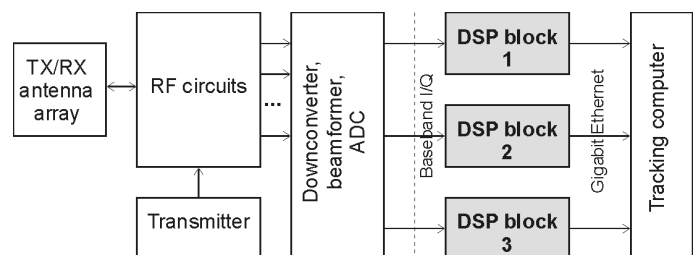

Fig. 1. Block diagram of the designed radar.

TABLE I
SUPPORTED PARAMETERS OF THE TARGETS

Parameter name	Symbol
Position relative to radar antenna	x, y [m]$/R$ [m], β[°]
Height above the ground	h [m]
Power of the reflected signal (relative to the noise floor)	P [dB]
Type of the movement (trajectory)	M
Velocity	v [m/s]
Direction of the movement	ϕ [°]

II. SIMULATION OF THE TARGETS

The user has to provide to the simulator a table of simulated targets. The target is modeled as a single point that reflects RF signals from the radar. For each target several parameters have to be defined, as shown in Table 1.

It is possible to define target position either in Cartesian (x, y) or polar (R, β) coordinates. Current version of the software allows three movement types (M): fixed (non-moving), linear, and circular movements. In case of the target moving in circles, the position of the circle center and the circulation period also needs to be defined.

Depending on the simulator setup, the amplitude of the reflected signal can be either fixed (A) or scaled with inverse of fourth power of the target range ($\sim A/R^4$). Some additional target parameters (e.g. elevation angle θ, Doppler speed v_D and Doppler frequency f_D) are calculated on a current basis during simulation. The targets' positions are updated according to their trajectory after every Elevation Search Cycle (ESC, see Fig. 2).

The software allows to plot a map of the targets' positions during the simulation process. The position of the moving targets and the direction of the antenna are periodically refreshed on the map. It is possible to switch on the option that records a movie to AVI file, showing all the targets and antenna movement in a real time. The example map of targets is depicted in Fig. 3. In this picture we can see eight targets moving in circles and the direction of the radar antenna.

III. MODEL OF THE TRANSMITTED SIGNAL

The hierarchical structure of the transmitted signal is shown in Fig. 2. In the designed radar, the transmitter sends the
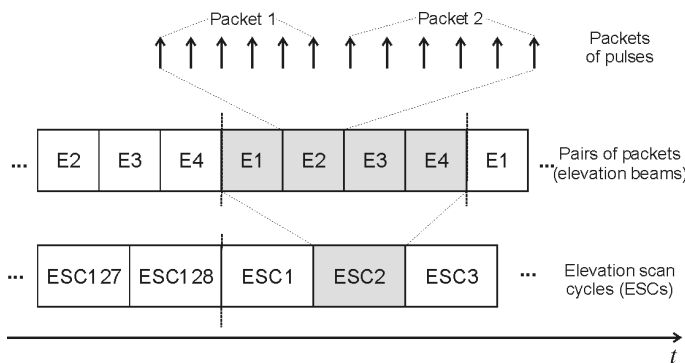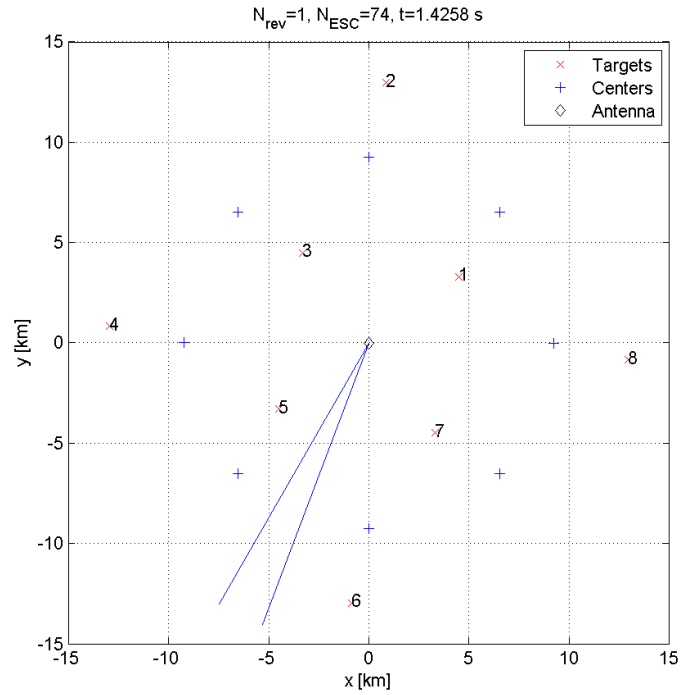

Fig. 3. Example map with simulated targets.

pulses aligned in packets (typically 7...11 pulses per packet). The pulses are transmitted with different repetition frequencies (PRF), determined by the design of MTD filters. One pair of packets is sent in each elevation beam (E1, E2, E3). In the fourth beam (E4), only one packet of pulses is transmitted. This packet contains more pulses and uses alternating periods between the pulses, as required in MTI filtering algorithm.

Four pairs of packets cover all four elevation beams (E1...E4) for one azimuth position of the radar antenna. We call such a structure *Elevation Scan Cycle* (ESC). The complete revolution of the antenna consists of 128 ESCs.

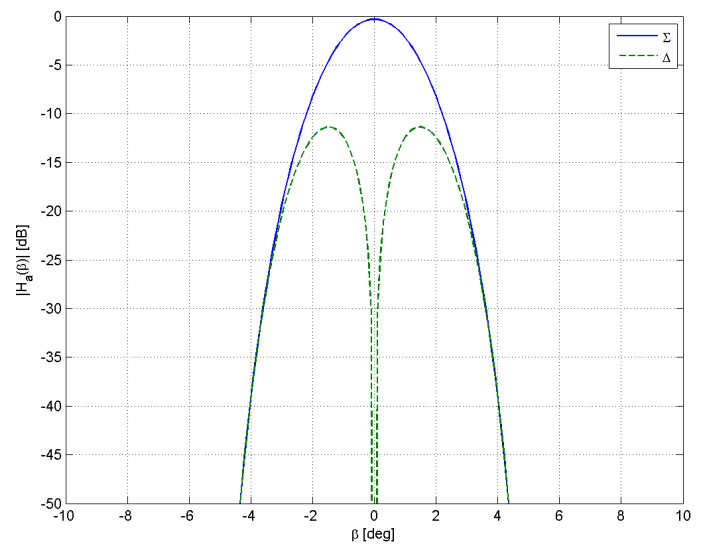
Fig. 2. Structure of the transmitted signal.

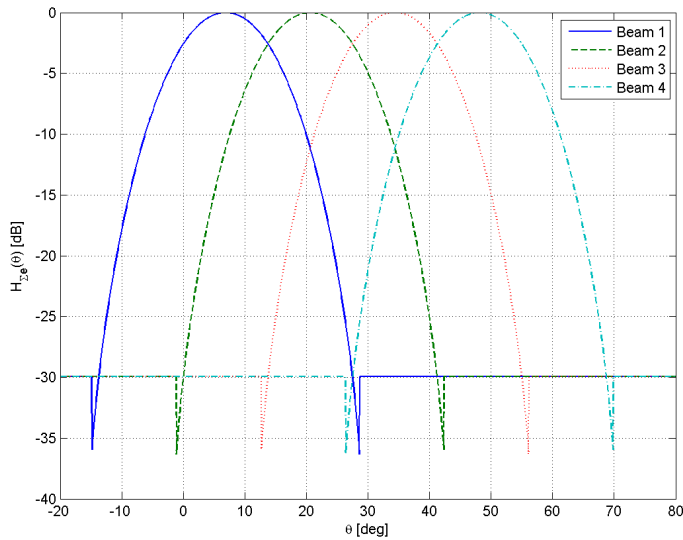Fig. 4. Amplitude of the simulated additive and differential antenna characteristics in horizontal plane.

Fig. 5. Amplitude of the simulated additive antenna characteristics in vertical plane.

Fig. 6. Simulated characteristics of the compression line.

IV. MODEL OF THE ANTENNA

The transmitting and receiving antennas are modeled by complex directional characteristics in horizontal and vertical planes.

The antenna characteristics are complex, i.e. change both amplitude and phase of the incoming signal. The receiving antenna always forms two beams: additive (Σ) for target detection, and differential (Δ) for estimation. Therefore, two antenna characteristics for each plane are necessary. In Fig. 4 and 5 we can see example antenna characteristics used during simulations.

For simplicity, in early stages of project development, in simulations we use combined transmitter receiver antenna characteristics: $H_{\Sigma a}(\beta)$, $H_{\Delta a}(\beta)$ – additive and differential characteristics in azimuth (horizontal) plane, $H_{\Sigma e}(\theta)$, $H_{\Delta e}(\theta)$ – additive and differential characteristics in elevation (vertical) plane. In the final release, these simple antenna characteristics will be divided into separate transmitter / receiver functions and replaced by much more accurate models obtained from the radar antenna array design.

In the RF and IF subsystem of the receiver, there is a very important filter matched to the pulse transmitted by the radar, known as the compression line. It allows to narrow down the received pulse width and thus to improve the resolution in range. This filter is also taken into account in the simulation and is modeled by the following characteristic:

$$H_R(r) = \begin{cases} \frac{75^2 - r^2}{75^2}, & \text{for } |r| \le 75, \\ 0, & \text{for } |r| > 75, \end{cases} \qquad (1)$$

where r is the distance between the target and the center of the range cell. The plot of characteristic (1) is shown in Fig. 6.

V. MODEL OF THE REFLECTED SIGNALS

In the simulator, we generate signals reflected only by the targets that are illuminated by the main lobe of the antenna characteristics in the horizonta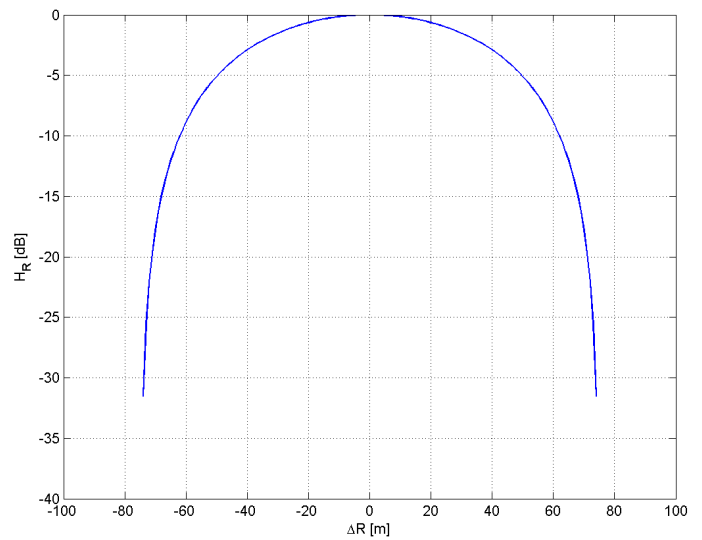l plane. Practically it means that for every antenna position (ESC) we neglect all the targets placed more than 4.6° away from the antenna direction, which allows to speed up the simulation significantly. The model of the reflected signal takes into account the following target parameters:

- distance from the radar,
- direction in horizontal and vertical planes,
- power of the reflected signal,
- Doppler speed (in case of moving targets),
- time of the pulse emission.

We generate three discrete complex signals ($s_\Sigma, s_{\Delta a}, s_{\Delta e}$) resulting from additive (Σ) and differential (Δ) complex antenna characteristics. After each k-th transmitted pulse, a vector of N_C samples ($n = 1 \ldots N_C$), corresponding N_C to range cells, is generated. Initially this vector contains only the samples of white Gaussian noise with fixed power. Subsequently, the components from currently visible targets are added to this vector, in places resulting from target distances:

$$s_\Sigma(n) = A H_{\Sigma a}(\beta_k) H_{\Sigma e}(\theta_k) H_R(r_n) \exp(j2\pi f_D T_k + \varphi_0), \qquad (2)$$

$$s_{\Delta a}(n) = A H_{\Delta a}(\beta_k) H_{\Sigma e}(\theta_k) H_R(r_n) \exp(j2\pi f_D T_k + \varphi_0), \qquad (3)$$

$$s_{\Delta e}(n) = A H_{\Sigma a}(\beta_k) H_{\Delta e}(\theta_k) H_R(r_n) \exp(j2\pi f_D T_k + \varphi_0), \qquad (4)$$

where:
A is the amplitude of the signal, resulting from the target parameter P (see Table 1),
β_k – angle between the target azimuth and antenna direction during emission of k-th pulse,
θ_k – elevation of the target during emission of k-th pulse,
T_k – time of the emission of k-th pulse,
φ_0 – random initial phase.
After including all the components reflected from the targets, the vector is stored in a file with a proper header. Subsequently,
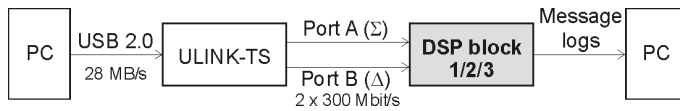
Fig. 7.   Block diagram of the connection with DSP block.

a main loop of the simulator begins processing of the next ESC.

VI. SIMULATION RESULTS

A. Output Data

The generated discrete baseband signals are quantized into 16-bit words and, after adding the required headers, organized into data blocks (packets) readable by the DSP processors. The simulation lasts for the number of antenna revolutions declared in the setup. The resulting data for each revolution is stored in two binary files (separately for Σ and Δ streams) on the hard drive. The typical single file length is 39 MB. The typical simulation time for one revolution is 9 s, obtained on a computer with Intel Core2 Duo 2.4 GHz CPU, Windows XP and Matlab R2009a software. The simulation, although much slower, is also possible in Octave, without any changes in the source code.

B. Transmission to DSP Processors

The data resulting from the simulation has to be transmitted to the DSP processors via Link Port interfaces [1], typical for the selected DSP family. This task is performed by a separate program (ULINK-TX) running in Windows environment. The diagram of a connection is depicted in Fig. 6. The hardware interface is provided by the ULINK-TS device [2], which acts as a converter between Hi-Speed USB 2.0 and TigerSHARC TS201 Link Port.

Two output ports of ULINK-TS converter are dedicated to transfer Σ and Δ data streams separately to detecting and tracking DSPs of one selected DSP block. Data is send via single-bit Link Port channels with LVDS DDR transmission (two bits in every clock cycle). The bandwidth of ULINK-TS device is ca. 28 MB/s and is divided between both Link Ports. Such a transfer rate allows sending the data for detection and estimation algorithms running in DSP close to the real time speed that will be generated in the working radar.

The message logs, generated by the detection and estimation algorithms running in DSP, can be compared with the target positions given in simulator setup. This way the performance of the algorithms can be determined.

VII. CONCLUSION

In order to develop the detection and estimation algorithms in the designed radar, the simulation software has been prepared. The simulator, named SYMDAS, works in Matlab (or Octave) environment and generates synthetic raw data. The simulation process is based on the given parameters of the targets and time schedule of the selected working mode. The generated signals are stored in binary files on the hard drive, two files for each antenna revolution. These signals can be subsequently transmitted to the Link Port inputs of the DSP processors with speed close to the real time.

The SYMDAS simulator allows our team to test, optimize and validate radar DSP algorithms which are developed in our project. With using this software, all the working modes of the radar have been checked, and the estimation accuracy in range, azimuth and elevation has been determined. The main advantage of this tool is that it allows to perform extensive laboratory tests, even before the RF part of the radar is ready, a long time before the first tests in the field can be carried out.

REFERENCES

[1] *ADSP-TS201 TigerSHARC Processor Hardware Reference*.   Analog Devices, 2009, ch. 6: Link ports.
[2] ULINK-TS converter description, http://www.progstar.com.pl/ulink-ts/ index_en.php, 2011.