Hardware Implementation of Small-Sized MODE-S Signal Receivers

Vasily Meltsov[1][0000−0001−5479−9979], Alexey Lapitsky[1][0000−0003−2325−4989], and Nikolay Mustafin[2][[0000−0001−5986−7221]]

[1] Vyatka State University, Kirov, Russia
[2] ITMO University, St. Petersburg, Russia
meltsov69@mail.ru andahay@gmail.com nikolay.mustafin@gmail.com

Abstract. Aircraft identification systems plays an important role in ensuring flight safety. To date, the most widely used system operating in MODE-S mode. The MODE-S standard defines the volume, content and sequence of flight data position transmitted upon request. The paper considers the design and implementation features of small-sized, but high-speed and efficient devices for decoding and analysis of MODE-S radio signals format. The data inside the signal is encoded with a Manchester code. In addition, when processing a signal from real air, several factors arise that can interfere with the correct detection of the preamble. The main factors are the amplitude-frequency distortion of the signals. The authors proposed several ways to increase the noise immunity of the MODE-S signal receiver. Next, a block diagram of the decoder was developed. Then two versions of a specialized receiver were implemented: based on a serial microcontroller STM32 and based on the FPGA of the Spartan 6 family from Xilinx. To conduct experiments to evaluate the correct functioning of the developed devices and compare their effectiveness, air records with a high message intensity were used. An analysis of the experiments results showed that the implementation on the FPGA significantly exceeds in efficiency (the number of detected signals and the number of correctly decoded signals) the version of the receiver on the microcontroller. Using the features of the FPGA architecture, it is possible to create the necessary number of specialized nodes and blocks working in parallel.

Keywords: Radio signals · Decoder · Noise immunity · FPGA.

1 Introduction

To date, all aircraft, both civilian and military, equipped with a large amount of electronics. These are navigation systems, automatic control systems and decision support systems for pilots. Aircraft identification systems play an important role in ensuring flight safety.

Interest in the creation of such systems appeared even before the Second World War. In the beginning, there was a need to recognize aircraft only according to the basic principle of "friend or foe". Over time, this system has evolved. Frequencies of data transfer, the transmitted messages structure and the filling of transmitted information packets changed.

To date, the most common identification system has become a system with MODE-S mode [1]. The MODE-S standard determines the volume, content and sequence of flight data location transmitted upon request: special identifier of the aircraft, coordinates, speed of the aircraft, flight altitude and some other parameters [2].

The use of MODE-S allows to control air traffic, have information about which aircraft are currently in the air. The mode allows not only to control the aircraft, but also to ensure the safety of their flights. Thanks to it, dispatchers of airports and aerodromes can coordinate the time of takeoffs and landings of all aircraft and exclude the possibility of their collision in places of active air traffic. It is important to note that the receiving of signals from aircraft can be performed not only from the ground by the airport dispatcher, but also by other aircraft. Information about the location of all the surrounding aircraft helps the pilot to quickly adjust the flight path if it is necessary.

The aircraft identification system described above is closely related to Automatic Dependent Surveillance - Broadcast (ADS-B) technology. In more detail, the abbreviation ADS-B means the following:

- Automatic - works automatically, without requiring human actions;
- Dependent - Depends on the GPS system and on the flight management system;
- Surveillance - provides surveillance of the aircraft like radar systems;
- Broadcast - broadcast continuous radio data transmission to all aircraft and ground stations;

The main idea of this technology is to determine the coordinates of the aircraft using GPS navigation system [3]. Next, the coordinates and other parameters listed above are transmitted to ground centers for dispatchers and/or other aircraft. ADS-B allows all air traffic participants to see the same picture of what is happening, which greatly improves mutual understanding between them. This improves the safety and flexibility of air traffic control.

2 Problem description

Transponder uses the following frequencies of radio signals during its work in MODE-S mode. The aircraft transponder receives a request from a ground-based radar at a frequency of 1030 MHz. It confirms the receiving of the request by the radiation of a phase-pulse modulated signal at the frequency of 1090 MHz. It should be noted that regardless of the signals from the ground-based radar, approximately every second transponder transmit an extended squitter containing the coordinates of the aircraft location.

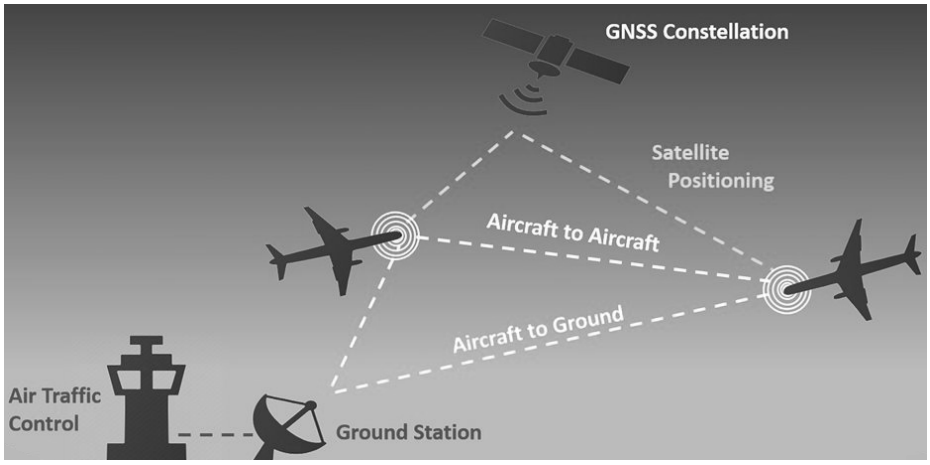

Fig. 1. Messages exchange scheme.

The signal transmitted by the aircraft in response to a request consists of two parts. The first part is the preamble. It contains 4 pulses that are separated from each other by a certain distance. The preamble serves to synchronize the receiver and transmitter of the radio signal. Its duration is 8 ns, and the duration of each four pulses is 0.5 ns. The structure of the preamble is shown in Fig. 2.

The second part of the signal transmitted by the aircraft is the information block. It can contain 56 or 112 bits of information. It depends on type of request that the aircraft received. The information inside the signal is encoded with a Manchester code, i.e. the bit value is determined by the transition of a signal from one level to another. For the "1/0" transition, the bit value will be 1, for the "0/1" transition, the bit value will be 0. Given the preamble structure (Figure 1), the preamble pulse duration, the pulse duration in the signal information part is 0.5 s. This means that it takes 1 s to transmit one bit.

An 8-bit analog-to-digital converter (ADC) with a sampling frequency of 16 MHz performs signal processing. It should be noted that Figure 1 shows the ideal signal. Processing a signal from real air arise several factors (features) that can interfere with the correct detection of the preamble.

The first feature of a real signal is that the waveform itself will not be rectangular [4]. This means that there will be no clear transition between signal levels. A transition from one value to another will take some time. During this time, approximately 7-8 values (points) will be obtained from the ADC. Thus, a rectangular signal has the corresponding "rise" and "descent" and takes on a form more reminiscent of an "elongated peak" with a duration of 14-16 points.

The second feature of working with a real signal is its varying amplitude. The maximum value that the used analog-to-digital converter can give out is 255. This is the value that must correspond to the highest signal level (i.e. 1). In fact, "logical 1" is far from always matching this maximum. It often happens that

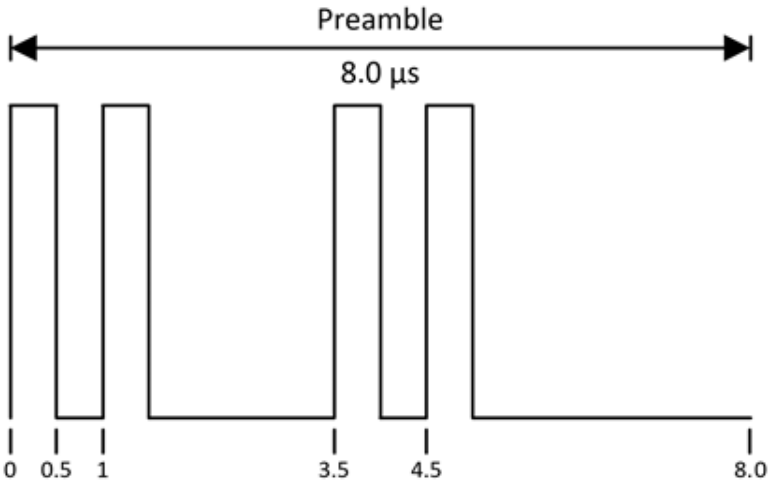

Fig. 2. Preamble structure.

the level "1" can fall even below 100 and fluctuate in the range from 50 to 100. This means that there is no single, well-defined threshold value for distinguishing between "logical 0" and "logical 1". This fact introduces great complexity into the operation of the modules responsible for identification, decryption, and signal analysis, since it becomes necessary to process signals of different amplitudes.

The third feature of real signals is that the signal may be subjected to "timing" distortions. This affects the position of the preamble peaks relative to each other. Based on the analysis of real-air signals, it was found that an increase or decrease in the pulse duration can reach 10%. Thus, it turns out that with a standard preamble duration of 8 ns, as a result of distortion, it can last from 7.2 to 8.8 ns. Fig. 3 shows a comparison of the ideal preamble with that which is being analyzed in real device processing. Despite the fact that the preamble contains only 4 peaks, an incorrect analysis of the first part, due to existing distortions, can lead to a "skip" (pass) of the entire radio signal.

The size of the information part is much larger and can be 56 or 112 bits. The transmitted information is encoded by the Manchester code: each bit is determined by the transition of the signal from 0 to 1 or from 1 to 0. When decoding data from the information part, signal distortion has a much greater influence on the decoding accuracy. The solution of the problem with decoding the signal information part will be discussed in more detail below. It is important that the problems of decoding the information part do not have any effect on the process of detecting the preamble on the air, since the preamble is transmitted before the information with the data packet.

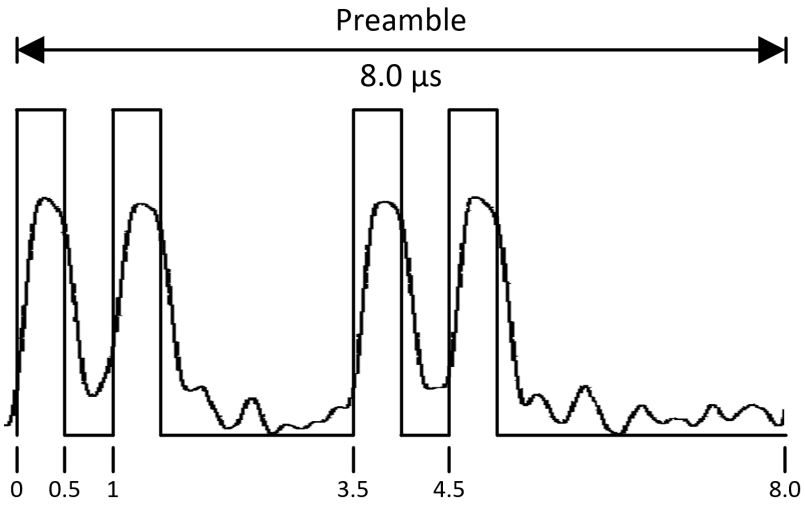

Fig. 3. Ideal and real preamble comparison.

3 Ways of decoder realization

Professional electronic equipment for avionics is a fairly complex system with a large number of independent devices that performs certain specialized tasks. Airports, private aerodromes, and air traffic monitoring centers uses such sysytems. The design and implementation of such equipment requires significant time, human and technical resources. In addition, professional systems have, significant dimensions, as a rule. For example, in one of the most compact solutions, its dimensions are comparable to the sizes of a standard system unit [5].

At the same time, for the aircraft observation and the elementary analysis of activity in airspace, coordination of aircraft movement is not required. In this case, it is sufficient to use compact receiving devices. In addition, developers of electronic components for unmanned aerial vehicles can also take simpler approaches when creating MODE-S signal receivers. There are two main options for their hardware implementation: based on serial microcontrollers (MK) and based on FPGA [6]. In both cases, the device will be placed on a board a little larger than 10x7 centimeters.

Implementing a MODE-S signal receiver on a microcontroller is a simpler task [7]. The microcontroller already has all the necessary nodes and blocks (in this case of talking about the operating device, internal memory, comparators, etc.). All processing will take place at the firmware level, which greatly simplifies not only the development process, but also the stages of debugging this system. The main disadvantage of the microcontroller implementation option is the low speed of the device, relative to the FPGA realisation. And this drawback, unfortunately, can greatly affect the quality of signal processing in the analysis of dense air traffic.

Using the capabilities of modern FPGAs, it is possible to design more detailed architecture of a device specialized for processing radio signals [8]. The number of gates, even in low-cost microcircuits from leading manufacturers, makes it easy to implement parallel data processing and, thereby, significantly increase the device's performance [9]. In addition, a large number of input and output pins allows you to connect all the necessary equipment for conducting experiments without special difficulties. The main disadvantage of this implementation method is the need to develop all elements and components of the device "from scratch". Also a small minus of FPGA development is the lack of a simple and quick mechanism for debugging an ongoing project.

Despite the fact that the approaches to designing devices are different, the functional structures of the devices will coincide, as they are determined by the sequence of processing steps that must be performed on the received signal. The first step is to detect the message preamble. This step is quite important, since the accuracy of determining the preamble affects the correct detection and decoding of the information part of the message that immediately following it. The message structure is shown in Fig. 4.

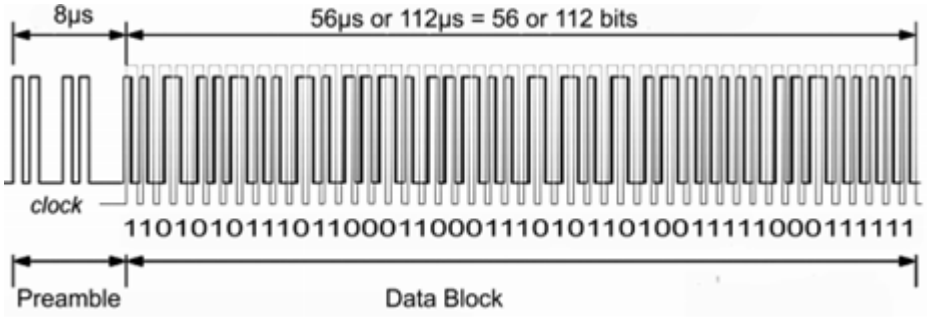

Fig. 4. MODE-S message structure.

It is very important to determine the moment from which the transmission of a special signal code begins. Even one missed value from the ADC can lead to incorrect decoding of the signal and its further loss. Since the structure of the preamble is known, it is possible can use the correlation function to detect it [10]:

$$\psi_{12}(t) = \int_{-\infty}^{+\infty} s_1(t)s_2(t-\tau)\ dt = \int_{-\infty}^{+\infty} s_1(t+\tau)s_2(t)\ dt \tag{1}$$

Also, to simplify further signal processing, it is necessary to perform a discrete Fourier transform [11]. Since the information fragment of the signal has a finite length, then the conversion will be performed accordingly:

$$x_p(n) = \frac{1}{N} \sum_{k=1}^{N-1} X_p(k) e^{j\frac{2\pi}{N}kn} \tag{2}$$

$$X_p(k) = \sum_{n=0}^{N-1} x_p(n) e^{j\frac{2\pi}{N}kn} \tag{3}$$

After the preamble has been found and the beginning of the signal information part has been determined, it is necessary to convert the manchester code into a standard binary representation of the signal [12]. In the information part of the signal, the duration of one information bit is 1 s. As was mentioned above, an ADC with a sampling frequency of 16 MHz is used for the design of the decoder, it meant that, the period of data income is 0.0625 s. This satisfies the criteria of the Kotelnikov theorem and provides sufficient accuracy for high-quality decoding of the signal. In accordance to these equipment parameters, during the processing of one information "bit" presented using manchester codes, 16 values of signal level (points) will be obtained from the ADC. Provided that each "bit" of the source information is encoded by the transition of the signal level from "1" to "0" (or vice versa), the converted "bit" will be represented by eight values of the younger "half" and eight values of the senior "half". The representation of the information bit after conversion by the ADC is shown in Fig. 5.

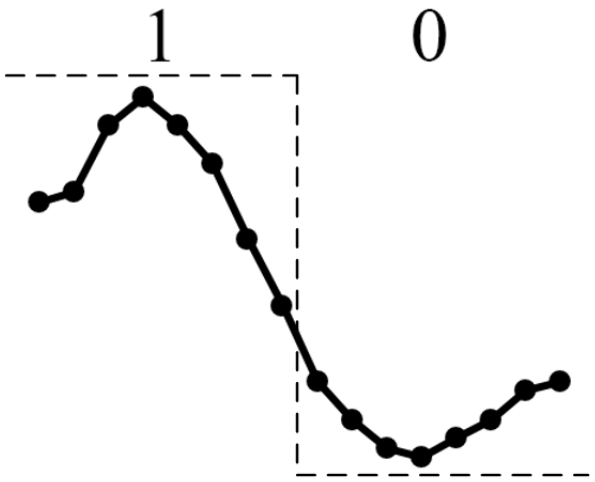

Fig. 5. Bit presentation after ADC conversion.

The simplest and most well-known method of decoding a converted signal is a weighted comparison. This method involves the following steps:

1. Determine the sum of the values of the first eight points of the decoded bit.

2. Determine the sum of the values of the second eight points of the decoded bit.
3. Subtract the second from the first amount.
4. If the result of this subtraction is positive, then 1 was encoded. If the result is negative, then, respectively, a logical 0.

$$V = \sum_{i=0}^{n} a_i + (-1)a_{i+\frac{n}{2}} \tag{4}$$

It is possible to modify this method by comparing the sums of the values of the first and second half of the bit on the comparator. This will reduce the time of comparison and will let immediately proceed to the analysis of the information part of the signal:

$$bit = \sum_{i=0}^{\frac{n}{2}} a_i > \sum_{j=\frac{n}{2}}^{n} a_j \tag{5}$$

Most transmitted messages of the MODE-S format contain a checksum (CRC) to increase reliability and noise immunity [13]. After receiving and decoding the preamble of the received message, it is necessary to calculate the checksum and compare it with the reference value stored in a special table. If the comparison was successful, then the information part is sent for processing on the computer, and the processing results will be visualized on the screen. Taking into account the specifics of the above steps, a block diagram of the decoder was developed (Fig. 6).

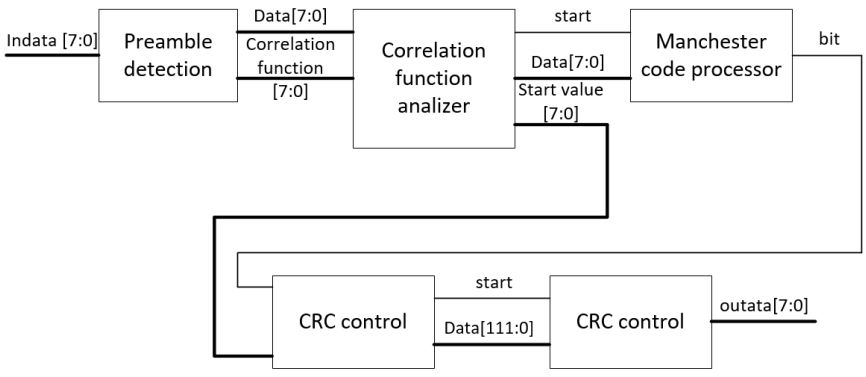

Fig. 6. Decoder structure scheme.

4 Experiments

To conduct comparative tests, both versions of the receivers were developed - based on the microcontroller STM32 and based on the FPGA of the Spartan 6 family from Xilinx. The same ADCs and additional equipment were connected to both devices to create the same working conditions.

FPGA design differs in that by "programming" the device there is ability to create an architecture from basic logic elements. Using this feature, it is possible to create the required number of specialized nodes and blocks working in parallel, each of which performs one of the above-described steps for processing a message. The volume of gates (logic gates) in modern FPGAs allows you to synthesize a device with a very high speed. This advantage becomes extremely important while processing a large number of transmitted signals in areas of air traffic with very dense traffic (for example, in areas of airports). The presence of FPGAs I/O nodes (I/O Cell) allows you to connect the required number of sensors and auxiliary equipment.

To conduct experiments to evaluate the correct functioning of the developed devices and compare their effectiveness, an air record with a high message intensity was used. In the experimental setup, a modulator was used as a virtual transmitter, which transmits the recorded signal through the SMA antenna cable in analog form. This is necessary to create conditions similar to the actual functioning of the transmitter device. Record of air contains 94358 signals. The experimental results are presented in table 1.

Table 1. Test results.

Compare parameter	MCU	FPGA	FPGA advantage
Number of detected signals	22783	75354	3,3 times more
Number of correctly decoded signals	12923	61243	4,7 times more
Average signal processing time (length - 56 bits)	80 μs	58 μs	38%
Average signal processing time (length - 112 bits)	145 μs	114 μs	27%

An analysis of the results of the experiments showed that the implementation on the FPGA significantly exceeds in efficiency (the number of detected signals and the number of correctly decoded signals) the version of the receiver on the microcontroller. This can be explained by the fact that FPGA spends less time on processing of incoming data. Firstly, the functional organization of the device on the FPGA is specialized for solving a specific problem of decoding messages in the MODE-S format. For developers on microcontrollers, only a fixed set of solutions and tools inherent in the existing chip is available. Retreat from the architecture of MK does not work in any way. Secondly, the high performance of the FPGA device is explained by the possibility of parallel processing of incoming data. It is possible to use pre-calculations of the necessary parameters when implementing on the microcontroller only for some steps of the algorithm. For most processing steps, it is not possible to calculate the coefficients and

average values in advance, and the sequential implementation of these stages of the algorithm takes too much time. While the microcontroller is processing the already detected signal, it's already necessary to receive a new message. The lack of performance leads to the omission of the preamble and, consequently, to the loss of some of the available signals on the air. Thus, the design of a receiving and decoding device based on FPGAs provides higher speed and efficiency of identification systems for aircraft and other aircraft.

5 Features of preamble detection on the FPGA

As mentioned above, due to the known structure of the preamble, the autocorrelation function (1) is used to detect it. In general, this function can be used when there is a signal standard, or a fragment of a signal with which a new one needs to be compared. The autocorrelation function shows how similar the two signals are to each other. Despite the fact that the use of this formula gives good results, it has a number of significant drawbacks. The main one is the difficulty of implementing FPGA calculations according to this formula. Not always, these calculations are integer, as a result of which their execution time also increases. Obviously, it is necessary to find an algorithm based on a simpler calculation mechanism, but at the same time allowing preserving the quality of detection of the preamble. It was decided to apply an algorithm using integer arithmetic. In this case, in addition to the ease of implementation, it is possible to significantly reduce the calculation time, in comparison with the known options.

Due to the fact that the reference form of the preamble is known in advance, it was decided to make a semblance of a "template" and check whether the received real signal is comparable with the standard of the preamble. From the standard it is known that the duration of the preamble is 8 ns. The sampling frequency of the ADC is 16 MHz. From this we can conclude that after converting the preamble by the ADC, 128 values (points) of the digitized signal will be received. The duration of one pulse in the preamble is 0.5 s, which corresponds to 8 received points from the ADC, which will correspond to the logical "1" value.

Based on the foregoing, it was decided to use a queue of 128 memory elements, which will contain the first part of the signal from the transponder, that is, the preamble. Each new value (point) from the ADC will be written to the beginning of the queue, and the rest of the values will be shifted by one memory element in the direction of older values, displacing the oldest.

Using the amplitude values at the points of the digitized signal, we will calculate a function showing presence of the preamble in the saved fragment. This can be done by knowing the reference form of the preamble. Thanks to the use of the queue, it is possible to precisely determine in which memory elements each of the impulses will be contained (ie logical level "1"):

- 1 pulse - interval 0 ns - 0.5 ns. Registers 0 through 7 correspond;
- 2 pulses - interval 1 ns - 1.5 ns. Registers from 16 to 23 correspond;
- 3 pulses - interval 3.5 ns - 4 ns. Registers 56 to 63 correspond;

– 4 pulses - interval 4.5 ns - 5 ns. Registers 72 to 79 correspond;

The remaining registers will contain a logical level of 0. Based on the values contained in the above registers (whether they exceed, and by how much, the values in the registers with logical 0), you can determine the presence or absence of a preamble in the received fragment of the ether. To process the digitized data, it was decided to use the following equitation:

$$F = \frac{\sum_{i=0}^{7} a_i + ... + \sum_{i=72}^{79} a_i}{32} + \frac{\sum_{i=8}^{15} a_i + ... + \sum_{i=80}^{127} a_i}{96} \quad (6)$$

According to this formula, calculations are performed when each new value is received. It is necessary for not to miss the moment when the calculated function reaches a local maximum. Then it will be possible to conclude that the registers contain a preamble. The theoretical calculation time using the formula is 45-55 ns, which is significantly less than the solutions considered above. Also, this time is less than the time interval before the appearance of a new value with the ADC equal to 62.5 ns.

6 Results and Discussion

To verify the effectiveness of the implementation on the FPGA of the proposed integer algorithms, it is necessary to conduct tests similar to those that were used previously. The results of repeated experiments are presented in table 2.

Table 2. Test results.

Number of signals	Integer algorithm results	Integer algorithm percentage	Correlation algorithm results	Correlation algorithm percentage
12536	8965	72%	9345	74%
54315	35563	66%	34657	63%
123879	86474	69%	80474	64%
236097	159321	68%	111654	47%

From the test results it can be seen that the proposed algorithm shows a fairly good efficiency. This can be explained by the fact that the parameters of the preamble that it analyzes give the necessary accuracy of its detection. On average, the algorithm detects 68-69% of the signals from the broadcast record.

7 Conclusions

The results (the number of detected signals), which show integer algorithms, are of course inferior to the powerful industrial systems present on the avionics market today. Some known methods for detecting a signal transmitted by a transponder based on preamble decoding show very good results. They detect

almost all the preambles that are in the test sample. A significant drawback is their algorithmic complexity. As a result of this, the performed calculations require significant time costs, which can lead to the loss (skipping) of the information part of the signal immediately following the preamble. It is also worth considering that the use of floating-point arithmetic leads to a complication of the system architecture, and a significant increase in its dimensions.

The proposed integer algorithms, on the contrary, have a relatively short processing time for the preamble and fit into the interval of data receiving from the ADC, which is their main advantage. Secondly, the architecture of the detection and decoding device based on these algorithms will be much simpler. Thirdly, the error in determining signals from real air by integer algorithms is insignificant. And fourthly, the device for detecting and decoding signals occupies 25% of the area used in the FPGA tests of the Spartan 6 family from Xilinx, which allows you to bump into the implementation of all other transponder modules within a single chip.

Based on this, it can be concluded that the proposed solutions for the design and implementation of a specialized FPGA decoder will be more effective in terms of performance, size and detection quality of MODE-S format signals transmitted from transponders of aircraft and other aircraft.

References

1. Chang, E.: The story of MODE-S. MIT Lincoln's library, 19–34 (2000)
2. ICAO: Technical Provisions for Mode S Services and Extended Squitter. ICAO., Location (2012)
3. Yavuz, E.: An Introduction to GNSS: GPS, GLONASS, Galileo and Other Global Navigation Satellite Systems. 2nd edn. NovAtel Inc., Location (2017)
4. Lesnikov V., Naumovich T., Chastikov A.: Sensitivity Analysis of The Equivalent Direct Form of IIR Digital Filters. 2018 IEEE East-West Design and Test Symposium, IEEE, vol. 1. (2018). https://doi.org/10.1109/EWDTS.2018.8524745
5. AX680: ADS-B (Automatic Dependant Surveillance – Broadcast), https://www.thalesgroup.com/en/ax680-ads-b-automatic-dependant-surveillance-broadcast. (Last accessed 30 Oct 2019)
6. Kuvaes A., Meltsov V., Lesnikov V.: Features Of The Design Operating Unit Inference Engine and Its Implementation on FPGA. 2019 IEEE ElConRus, IEEE, vol. 1, pp. 110–115. (2019). https://doi.org/10.1109/EIConRus.2019.8657151
7. Noviello C.: Mastering STM32. Leanpub., Location (2018)
8. Meltsov, V., Lapitsky, A., Lesnikov V.: Features of Decoding Transponder Signal of an Aircraft Using FPGA. 2019 IEEE ElConRus, IEEE, vol. 1, pp. 120–124. (2019). https://doi.org/10.1109/EIConRus.2019.8656910
9. Breaking Down Barriers in FPGA Engineering Speeds up Development, https://spectrum.ieee.org/computing/networks/breaking-down-barriers-in-fpga-engineering-speeds-up-development. (Last accessed 30 Jan 2020)
10. Correlation function - Wikipedia, https://en.wikipedia.org/wiki/Correlation_function. (Last accessed 30 Jan 2020)
11. Discrete Fourier transform - Wikipedia, en.wikipedia.org/wiki/Discrete Fourier_transform. (Last accessed 30 Jan 2020)

12. Manchester Encoding in Computer Network - GeeksforGeeks, https://www.geeksforgeeks.org/manchester-encoding-in-computer-network/. (Last accessed 30 Jan 2020)
13. Cyclic redundancy check - Wikipedia, https://en.wikipedia.org/wiki/Cyclic_redundancy_check. (Last accessed 30 Jan 2020)