Research of Digital-Analog Conversion Method for Reproduction of Mechanical Oscillations

Stanislav Danylenko, Oleksander Vechur, Mariya Shirokopetleva

Kharkiv National University of Radio Electronics, Nauky Ave. 14, Kharkiv, 61166, Ukraine

Abstract

This paper considers the development of software-hardware complex for an alternative method of sound reproduction – by means of air. The concept of this method, the justification of this possibility and the implementation of the software-hardware complex, which is a mechanical device with connected fans for the distribution of air flows and software for its operation. Particular attention is paid to the algorithm for converting the input digital signal to the output analog one, which is based on the sound frequency analysis, and the output is generated using a mechanical device using PWM. Experiments were conducted to study the effectiveness of this method using humans and other software system. Based on the results of experiments, conclusions were made about the possibility of its use.

Keywords

Software-hardware complex, sound characteristics, auditory perception, digital-to-analog conversion, Fourier transform, spectrum analysis, pulse-width modulation, Arduino.

1. Introduction

Hearing is one of the main ways to communicate between people and obtain information from the outside world. And these capabilities: detecting, locating and identifying sounds are impressive, given the input we receive – the eardrums detect changes in atmospheric pressure and provide the information about the continuous degree of change in sound pressure at two points near the ears to brain for analysis. It should be borne in mind that this information comes to a person in the form of a mechanical acoustic wave with a frequency of 16 Hz to 20,000 Hz, which contains a set of mixed uninsulated sounds, and this is sufficient for isolation and identification of sounds [1].

Mankind has described sound physically, introducing many terms that allow to unambiguously define certain characteristics, such as frequency, amplitude of waves, speed of sound, sound pressure. Separately described characteristics that relate only to acoustic oscillations, such as harmonics, octaves or sound level, determined how sound differs from noise [2].

However, there is a group of people around the world who have hearing problems and cannot receive sound information from the outside world in the usual way. Medicine helps them solve these problems, but it is not always possible to solve this problem with the help of hearing aids or other currently known tools.

There is a task to find other ways to solve this problem, namely the study of alternative ways to transmit and perceive sound using other human sensory systems, which is the purpose of this work. We propose to consider the method of identification of sounds transmitted by air flows generated by fans and recognized by humans by means of skin.

The research involves the use of sounds recorded in digital form, but the fans are controlled by analog voltage values, so it is needed to convert one type of signal and another and accordingly the object of this study is digital-to-analog conversion.

COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems, May 12–13, 2022, Gliwice, Poland
EMAIL: stanislav.danylenko@nure.ua (S. Danylenko); alexander.vechur@nure.ua (O. Vechur), marija.shirokopetleva@nure.ua (M. Shirokopetleva)
ORCID: 0000-0002-8142-3018 (S. Danylenko); 0000-0001-9605-1475 (O. Vechur); 0000-0002-7472-6045 (M. Shirokopetleva)

The subject of research is the processing of audio information, its reproduction and transmission in the environment.

The research is a theoretical analysis of the use of the proposed method, modeling and creation of an experimental sample, which is a software-hardware complex and conducting an experiment with its use and human participation. Also, an experiment to identify the possibility of recognizing audio information transmitted in this way by computer systems and to draw conclusions about the possible use of the studied method.

2. Related Works

No work related to such research has been found, but some work on other ways of alternative sound reproduction and perception has been considered.

A method of transmitting sound using the microwave auditory effect (Frey effect) has long been discovered. With this effect, sounds occur in the human head without a direct impact on the eardrum. Initially, this phenomenon was observed during hostilities and was considered to be auditory hallucinations, but then, after further study, this phenomenon was given the name. However, further research did not follow and in this way only managed to transmit to the brain information about the numbers from 1 to 10. This could be quite promising, but requires human testing, and because it can be dangerous to health, such experiments are banned in many countries [3].

Another alternative is to use a piezoelectric effect, the essence of which is to transmit sound to the inner ear. This technology has already been widely used for people with hearing impairments in hearing aids and modern electronics, and materials with piezo conductivity are being actively studied to improve the achieved results [4].

The considered alternative ways are based on the study of human perception of sound through the outer and inner ear and this area continues to be widely studied.

In terms of processing audio information by computer systems spectroanalysis is widely studied, which is the basis for analysis and conversion of information from sound waves. Various window functions for preprecessing or postprocessing and Fourier transforms in different representations are used for its implementation [5].

This knowledge is widely used in sound recognition systems based on neural networks, which have learned to perform their functions with high accuracy and have achieved sufficient efficiency and compactness to be placed on stand-alone boards and used in complex conditions. And they help solve important tasks of monitoring, detecting and immediate response to emergencies [6, 7].

Another research has also been conducted in a similar field, namely the use of pulse-width modulation (PWM) to generate waves and then use spectrum analysis and Fourier transform to apply filters to achieve digital-to-analog conversion. This created a digital-to-analog converter for hardware platforms such as TMS320X281X from TI company [8, 9]. Many microcontrollers do not have the ability to perform digital-to-analog conversion, although this capability is necessary for many tasks. This is a good confirmation that the approaches proposed for use in this work are relevant and their use is real.

3. Methods, Modeling and Creation of a Test Sample

The ambient sound that a person hears is continuous, analog and cannot be reproduced in the same way several times. Therefore, it is necessary to record this sound and record it in computer-readable formats. This process is called analog-to-digital conversion, and recorded sound is digital or discrete, depending on the form in which it is presented [10]. Of course, such signal capture requires a special format of data storage and pre-processing.

For the simplest way to record sound mechanical oscillations, record on the microphone can be used, and as an example of storage format is a file in mp3 format. In this case, the digitized data set must be processed for writing to a file of a certain format, this process is called encoding. Digital sound can be created directly by means of a computer generation. In this case the digitization stage is not performed, and the encoding is handled by the sound generator itself [11].

Audio recording is not a complete copy of the original audio. The most important characteristic of such a record is the sampling rate. It shows how many points from the initial signal were recorded in 1 second. Usually this value is 44100 Hz. This is due to Nyquist–Shannon sampling theorem, which states that in order to unambiguously restore the original signal, the sampling rate must be more than 2 times greater than the highest frequency in the signal spectrum. And because the person hears the maximum sounds with a frequency of 20 kHz, this value was chosen.

There are also other key features, such as sample depth, which shows the number of bits to encode each sample; the number of channels, the parallel playback of which forms the final sound signal; the number of bits used per unit time – is the bitrate. But the quality of the recorded signal depends less on them [11].

Using the characteristics described above, encoders write to the file only the information from the initial audio signal that is useful for its reproduction.

The reverse process is the process of reproducing the recorded digitized signal. These processes are called decoding and digital-to-analog conversion. And in this paper the last of them is considered and in addition processing of the decoded digital information from a source is carried out. With the help of non-standard spectrum analysis and digital-to-analog conversion by means of pulse-width modulation that it is possible to investigate the possibility of signal reproduction by means of fans and sound transmission by means of air. There will also be encoding and decoding steps for the values calculated by the algorithm so that these values can be transferred to a mechanical device that will reproduce them.

Since people distinguish sound by frequency, it will be used as a characteristic for processing by the algorithm. Fourier transform will be used for frequency analysis – an integral transformation of one complex-valued function of a real variable into another. By means of it we can get values in the frequency domain, knowing the values in the time domain. Since it is necessary to work with discrete information, a discrete Fourier transform with a window function will be used to process the finite interval and modification of the fast Fourier transform, for which it is necessary to choose a coefficient corresponding to the power of two [11]. This method is well researched and has ready-made software implementations, so it is possible to use them from free software libraries and focus on developing an algorithm for processing the data. At this stage, it is necessary to determine what values should be reproduced on each of the fans and in what form to be transmitted to the end device and how exactly to reproduce it.

Most fans require DC power for their operation, but there are also some modifications that have a separate control pin to which the analog voltage signal must be applied and thus control the speed. It is possible to achieve the supply of such a signal with a discrete signal using pulse-width modulation [9].

3.1 Modeling of a Software-hardware Complex

The idea of the studied method is to analyze the flow of audio discrete information using discrete Fourier transform and to identify the most common values in the frequency ranges, each range to match the fan and rotate it at a speed corresponding to the value found using digital-to-analog conversion by means of PWM. For this purpose, a software-hardware complex was implemented, which consists of the following hardware components: client, server and end device and the corresponding software for each node.

The client is the main component, it receives the initial signal. The signal can be obtained from an audio file, be generated by a computer or obtained from a microphone. Audio data for processing must be with a sample rate of 44.1 kHz, the chosen format of audio files for uploading for processing – mp3.

The server is an intermediate node between the client and the end device. Its functions are to establish a connection with the end device and transfer data between the client and the end device.

The end device is the Arduino platform, which can receive digital data through its interfaces and generate an analog signal for mechanical devices by applying a PWM signal.

To understand which frequency range is reproduced, a person needs to know how the fans are placed and the frequency ranges they correspond to. They are arranged in one plane in the shape of a square plate, with a 3 by 3 grid. It is shown in Figure 1.

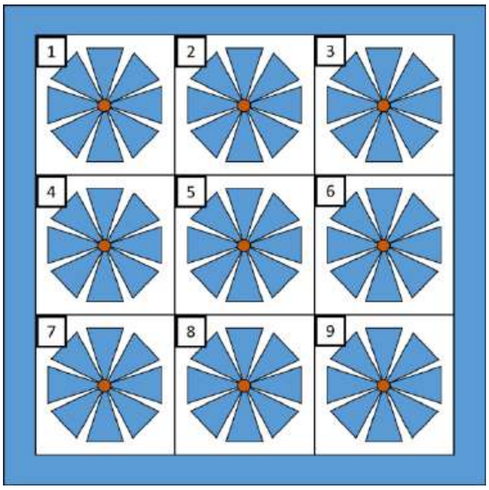

Figure 1: Scheme of fans location

Thus, the frequency ranges were divided into high (1, 2, 3), medium (4, 5, 6) and low (7, 8, 9). The correspondence between the fan and the frequency range is shown in Table 1.

Table 1
Frequency ranges that match for fans

№ of the fan	Frequency range
1	19,53 kHz – 21,96 kHz
2	17,09 kHz – 19,52 kHz
3	14,65 kHz – 17,08 kHz
4	12,3 kHz – 14,64 kHz
5	9,77 kHz – 12,2 kHz
6	7,33 kHz – 9,76 kHz
7	4,89 kHz – 7,32 kHz
8	2,45 kHz – 4,88 kHz
9	16 Hz – 2,44 kHz

The general architecture of the developed complex is shown in Figure 2.

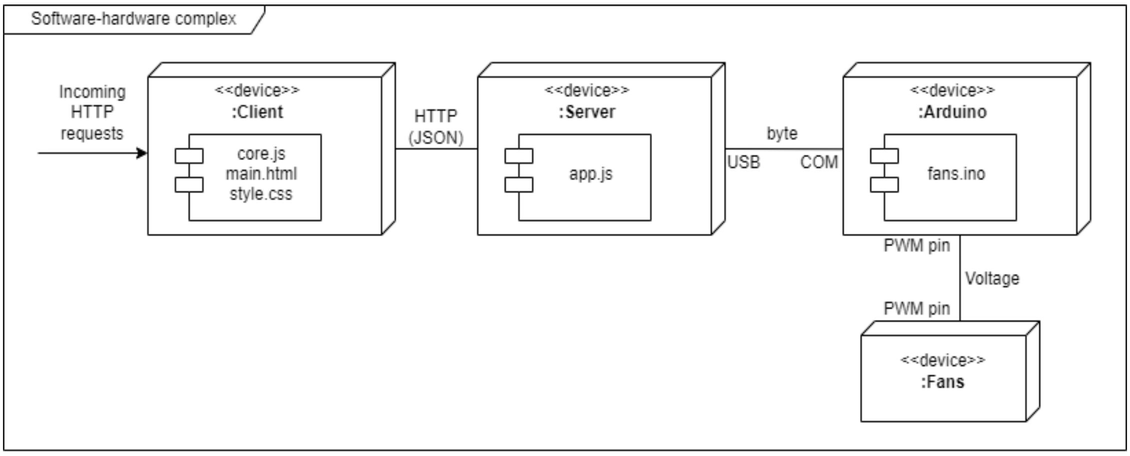

Figure 2: Deployment diagram of the software-hardware complex

Where the interfaces and data types that the nodes exchange are additionally defined on the diagram.

And the usual workflow is illustrated in the sequence diagram in Figure 3.

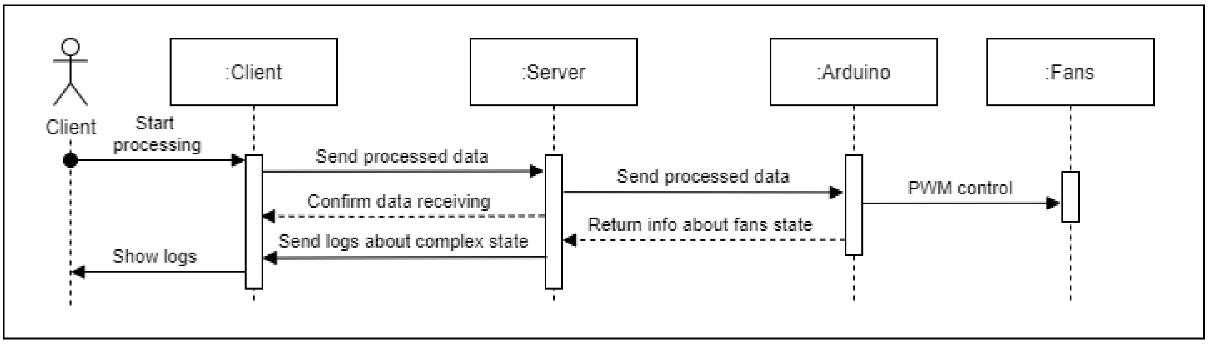

Figure 3: Sequence diagram of the simple transformation request

At the client, the signal is processed by an algorithm, recorded by an encoder in a form from which information can be converted into a format understandable to fans, delivered to the end device via a server, decoded to understandable values and reproduced by fans using PWM.

Additionally, after each step, its success is confirmed in order to detect errors or inform the user about the actions taken during the process. All operations are performed in streams, one after another, until the user completes the process.

3.2 Hardware Description

The client and server can be hosted on the same node or on separate ones. Such a node is a computer device running an operating system that can run applications implemented in the JavaScript programming language, such as Windows, Linux, or MacOS. It requires 220 V AC, at least 2 GB of RAM, 512 MB of ROM memory and a base CPU speed of at least 1 GHz.

The base of the end device is the Arduino MEGA 2560 board, as it has the required number of output PWM ports (pins). The serial port on the board side and the USB port on the server side are the interface between the nodes, and data is transmitted in byte format by wired connection. The Arduino MEGA 2560 board itself has the following characteristics listed in Table 2:

Table 2
Characteristics of Arduino MEGA 2560 [12]

Characteristic	Value
Operating voltage	5 V
Input voltage	7-12 V
Count of digital inputs / outputs	54
Count of PWM inputs / outputs	14
Constant steaming through inputs / outputs	40 mA
Flash memory	256 kB
RAM	8 kB
CPU clock speed	16 MHz

The Arduino requires a 5V DC power supply, which is implemented by connecting to a server.

Since the fans must form a square structure, as mentioned earlier, the maximum number of fans that can be is 9, because the maximum possible number of devices connected to the PWM ports is 14.

The Arduino is connected to nine Delta Electronics ASB0412MA fans with the characteristics listed in Table 3.

Table 3
Characteristics of fans Delta Electronics ASB0412MA

Characteristic	Value
Height	40 mm
Width	40 mm
Depth	10 mm
Operating voltage	12 V
Amperage	0,08 A
Noise level	20,5 dB
Power	480 mW
RPM	5000

The fans require a supply of 12 V DC, which is provided by an autonomous power supply and have 4 output pins, one of which is a contro pin, through which the control of fan speed is performed by Arduino [13].

Each Arduino PWM pin is matched to a fan control pin to perform speed control. Arduino and fans have a single zero phase. One fan pin, which is a speed sensor, remains unconnected and if necessary can be used for its intended purpose.

3.3 Software Description

The software was designed taking into account the maximum number of fans that can be connected to a specific software-hardware complex, but can be easily modified to support more or less fans.

The software implementation is performed using two programming languages – JavaScript and the C-shaped language of Arduino. HTML5 and CSS3 technologies were used to design the user interface.

Using the JavaScript programming language, a client and a server nodes were implemented, including a module with the basic algorithm, which is executed in the browser on the user's page. At the client node a module from the standard Web Audio API library was used to perform the Fourier transform [14]. Also, by means of standard components, the ability to download a file, generate or receive a signal from the microphone was implemented.

At the server node was implemented module to communicate with the end device using the SerialPort library and receive values from the client module using HTTP protocol by means of jQuery library [15].

Also, as a client extension, a software module for a classification experiment using the decision-tree library was implemented [16]. It contains values reader which intercepts values before sending them from the client and provides ability to save it to a file with csv extension. After that values can be read and used for building a prediction model.

The C-shaped language of Arduino was used to program the selected Arduino board. The functionality of connection to the workstation, data acquisition, processing and communication with mechanical fans by means of PWM was implemented.

3.4 Algorithm Description

The general structure of the algorithm is shown in Figure 4. Item 1 is obtaining of a discrete value of the acoustic wave from one of the available sources. Steps 2-5 are a discrete Fourier transform using Blackman window function and smoothing. Steps 6-12 are, in fact, a processing algorithm that has been developed and will be described in detail below. Step 13 is a call of the Arduino function to send a value to the fan and process it by means of PWM.

All operations are performed separately for each piece of data from the audio stream.

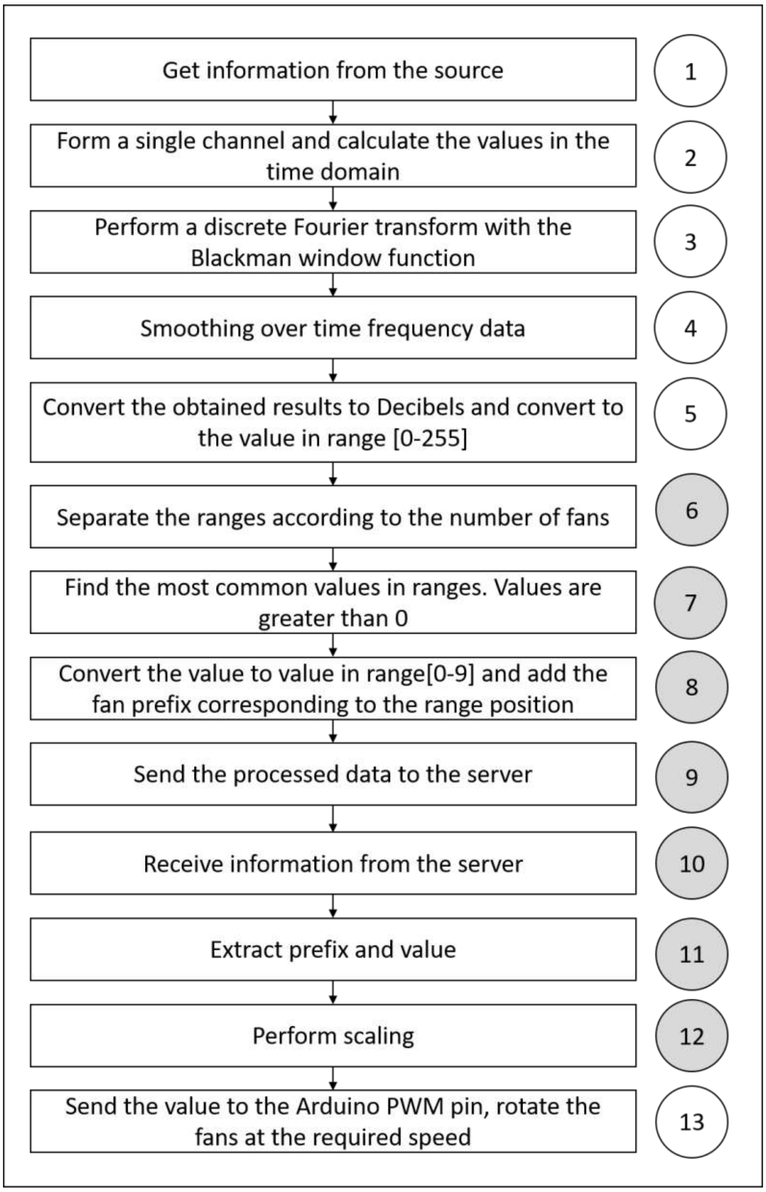

Figure 4: Algorithm

Steps 1-5 are performed using Web Audio API technology, they are typical spectrum analysis operations and are described in detail in the documentation. The following are brief explanations for it. Time domain – the area of dependence of sound pressure on time. Frequency domain – the value of the pressure of a wave of a certain frequency over time. The Blackman window is applied with the following parameters: $\alpha = 0.16$, $a_0 = \frac{1-\alpha}{2}$, $a_1 = \frac{1}{2}$, $a_2 = \frac{\alpha}{2}$ [14].

The discrete Fourier transform matches the time domain of a function to its frequency domain. A coefficient of the fast transform set to value of 2048 . This means that values for 1024 frequency points will be obtained at the output, and 21 analyzes will be performed for 1 second. However, the fan cannot change the speed so many times, so half the values will be skipped during further processing.

After spectroanalysis, we obtain an array with N bins, where N is the number of values obtained after the Fourier transform for one analysis. And the value of the element of the array b is the relative difference between the sound pressure waves of different frequencies in the range 0-255. However, the result of the Fourier transform has a mirror image and it makes sense to use only one of two parts, so the result is an array of size $N/2$, representing the force of sound pressure b at certain frequency points A, the value of which is determined by the following formula:

$$A[q] = q * B/C \quad (1)$$

where q – serial number of the bean, $q = 0,...,N/2-1$, B – sample rate, C – the number of points for which the frequency value will be calculated.

For example, for B = 44,1 kHz and C = 2048 values $A[n]$ will be as follows:

- A_0: 0 * 44100 / 2048 = 0;
- A_1: 1 * 44100 / 2048 = 21,5 Hz;
- A_2: 2 * 44100 / 2048 = 43 Hz;
- A_3: 3 * 44100 / 2048 = 64,6 Hz;
- $A_{N/2-1}$: 1024 * 44100 / 2048 = 22 kHz.

The points belong to the entire frequency range that a human can hear.

The first step is to divide the resulting data set by the number of fans I and the results will be grouped for this number and each of the fans will reproduce values only for a certain frequency range with a specific number of frequency points j:

$$j = N/2/I \quad (2)$$

where N – the total number of frequency points after the Fourier transform, I – determined by the configuration of the end device (in this case is 9).

A specific fan is denoted as i:

$$i = 0,...,8 \quad (3)$$

And the boundary values A for the range are calculated as follows:

$$A_{min(i)} = A[j * i]\ ;\ A_{max(i)} = A[j * (i + 1)] \quad (4)$$

where A from (1), j from (2), i from (3).

And for a specific fan, the range of frequency points is denoted as m:

$$m[i] = (A_{min(i)} \ldots A_{max(i)}] \quad (5)$$

where i from (3), $A_{min(i)}$ and $A_{max(i)}$ from (4).

And accordingly, the frequency points A will be assigned to one of the range that will correspond to a particular fan, and the final value in the range is calculated based on their respective values b. This is the most common value and is greater than 0. Denote as v:

$$v[i] = Mo(b[m[i]]) \quad (6)$$

where i from (3), Mo – mode of numerical values for the range, $b[m]$ – set of values b from the range m, which satisfy the condition, m from (5).

Then the values v from (6) translated from the scale 0-255 to scalse 0-9 with the rules of mathematical rounding and is denoted as V:

$$V[i] = v[i] / 255 * 9 \quad (7)$$

where i from (3), $v[i]$ from (6).

Then a prefix corresponding to the serial number of the fan is added to the value. The number should have an unsigned byte format [0-255], so there is a limit of 24 fans and the corresponding byte value of 249, but in this case for 9 fans the maximum value can be equals to 99. Then the value obtained is denoted as u:

$$u = (i + 1) * 100 + V[i] \quad (8)$$

where i from (3), $V[i]$ from (7).

Then in steps 9 and 10 the value of u is sent to the end device.

After obtaining the numerical value at the end device, it must perform the inverse transformations – separate the prefix p by dividing without remainder by 100:

$$p = u / 100 \quad (9)$$

where u from (8), p is a value from (3).

And separate the value of U by finding the remainder of the division by 10:

$$U[p] = u \% 10 \quad (10)$$

where p from (9), u from (8).

The penultimate step is to convert the value to the range 0-255 using mathematical operations, because the value in this format is used by Arduino for PWM conversion. It is necessary to scale the value depending on the minimum value of L from which the particular fan starts running. Denote the final value as *F*:

$$F[i] = L[i] + U[p]/9 * (255 - L[i]) \quad (11)$$

where *i* from (3), *p* from (9), with *i = p*, *L[i]* – the minimum value from which the fan starts running, *U[p]* from (10).

The last step is to call the Arduino interface function, which gets the generated value and converts it to an analog voltage value using PWM conversion.

4 Experiment and Results

The effectiveness of the described method is determined by the number of correct sound identifications. This number will show the percentage value. Let's denote it as *Y*:

$$Y = \frac{n}{N} * 100\% \quad (12)$$

where *n* – the number of sounds that were identified correctly, *N* – the total number of sounds that were tested.

Sets with different numbers of sounds and different sound durations will be used in the experiment. Therefore, we denote this value as *Y(ct)*:

$$Y(c_i t_j) = \frac{n(c_i t_j)}{N(c_i t_j)} * 100\% \quad (13)$$

where *i* – ordinal number of the element from the set of the number of sounds, c_i – the count of sounds from the set (4, 9, 14), *j* – serial number from the set of duration of sounds, t_j – duration of sounds in seconds from the set (2, 5, 10).

With the help of these obtained values it will be possible to conclude that it is possible to use such an alternative method of reproducing sound waves as a whole and for a specific set of characteristics of the sound duration and number of sounds. To determine the positive result, we will introduce a threshold level of 75%, taking into account all external factors.

Two types of experiments were performed using the studied method – one aimed at recognizing sounds using computer systems, and the other – recognizing by people.

4.1 Computer Experiment

The previously mentioned software module for testing was developed for the classification of sounds from the information received from the client using the decision tree and the ID3 algorithm. Because the values for the experiment were recorded before sending them to the end device, it was claimed that another software and hardware complex recorded these values without loss, in the same form in which they were sent to the end device and could be reproduced. The results obtained by this way do not include errors or the action of external factors, so they are slightly better than those that could be obtained in real life. On the other hand, computer systems do not always need data from produced air flows, in some cases they may expect coded values that can be obtained in one of the available ways and identify the sound from the data.

Decision trees were used because they are a classic tool for classification and have ready-made software implementations, because writing our own is out of scope of this work [17].

During the experiment the complex worked in the usual mode, alternately played audio files for 5 classes of sounds: shot, sounds from the chicken coop, the work of the car's internal combustion engine, trimmer work, playing a musical instrument – ukulele. The data after processing by the algorithm and before sending to the end device were written to files with CSV extension and used for sampling.

The training sample had 20 items for each class. The sounds lasted 5 seconds, the values were recorded every 100 ms. Thus, 100 training items were formed, each of which represents a set of 450 numerical values in a range from 0 to 9. The test sample consisted of 5 items for each class.

After the samples were formed, an experiment was performed.

After the testing, we got a result, where none of the attempts were successful and hence the percentage of correctly predicted classes is 0%. So it can be concluded that by means of the decision tree using the ID3 learning algorithm it is impossible to teach the model and identify sounds by class, obtaining information from the implemented software-hardware complex. The decision tree cannot find the correct relationships between 450 features that are similar to each other

In addition, it was tested whether the model will be able to correctly recognize the sound, the data of which were in the training sample, but in this case, a negative result was also obtained. Evaluation of the obtained model by means of the software library also showed the unsuitability of the model for use.

We made an assumption that it is possible to achieve this goal with more sophisticated approaches, such as the previously mentioned neural networks.

The data obtained with the module for this experiment can be used to compare the reproduced sounds using a graphical representation, which will be illustrated below.

4.2 Human Experiment

Five people took a part in the experiments. Each person was independently instructed and passed the training. Each participant had 2 attempts at each stage of the experiment. Training at each stage, at the request of the candidate, was performed with or without headphones turned on. However, it was found that the effectiveness of training without headphones is better.

During the initial training, participants had to remember which fan corresponds to a specific frequency range. To do this, we used the functionality to generate sounds of a certain frequency. Thus, only one fan worked at maximum speed. The anamometer was used to record compliance with the maximum speed at a distance of 2 cm – 2 m/s.

The person placed his face in front of the fans and memorized the correspondence. At the same stage, the optimal distance at which a person needs to be in order to be able to recognize the source of air flow was determined – 2-7 cm when directed at the person's face. Shorter distances are sometimes impossible and generally dangerous to health, but at longer distances airflows are mixed and make them almost impossible to recognize. This option of using the complex in real life will also be used to train and explain the possibilities of the complex.

At this point, the initial training was completed, candidates memorized the matching of fans and frequency bands. An experiment was performedand and the results of airflow source identifications were generated, which are given in Table 4. Where abbreviation “P1-1” and similar ones means: the first digit – serial number of the participant, the second digit – his first or second try.

Table 4
Results of generated sounds experiment

	Fan 1	Fan 2	Fan 3	Fan 4	Fan 5	Fan 6	Fan 7	Fan 8	Fan 9
P1-1	+	+	+	+	+	+	+	+	+
P1-2	+	+	+	+	+	+	-	-	+
P2-1	+	+	+	-	-	+	+	+	+
P2-2	+	+	+	+	+	+	-	+	-
P3-1	+	+	+	-	-	+	+	+	+
P3-2	+	+	-	+	+	-	+	-	+
P4-1	+	+	+	+	+	+	+	+	+
P4-2	+	+	+	+	+	-	+	+	+
P5-1	+	-	-	+	+	+	+	+	+
P5-2	-	+	+	+	+	-	+	+	+

In most cases, the accuracy is 80-90%, which is a positive result.

Testing using a streaming signal from the microphone was not performed, despite the fact that the developed complex supports this possibility, because the ambient sounds are constantly changing and form a background noize. However, it is not possible to produce the same sound twice, which makes the experiment non-objective. We can use the recorded sounds and consider the result identical. This possibility is provided by the complex and the results of its testing are given below.

Then there was a stage of testing with the reproduction of sounds from audio files in mp3 format. In total, the experiment had 14 sounds: playing the piano, evening field, playing table tennis, electric trimmer, internal combustion engine, gunfire, a cappella female vocals, sounds from the chicken coop, pouring milk in a glass jar, game on the ukulele, traffic jam, kettle whistle, train movement, ambulance siren. Each file had 3 versions of different durations: 2 seconds, 5 and 10 seconds.

The training was conducted in the same way as in the previous stage, only now it was necessary to remember the operation of several fans. Testing was performed for each pair of characteristics of sound duration and number of sounds. The participant had to correctly identify the sound being played.

For each experiment, the results were calculated and summarized in the resulting table (Table 5), which shows the percentage of correct identifications for the characteristics sets.

Table 5

Summary result of the experiment for different sound count and sound duration

Count	Duration			Average value
	2s	5s	10s	
4	72,50%	82,50%	72,50%	75,83%
9	56,67%	67,78%	60,00%	61,48%
14	25,71%	38,57%	27,86%	30,71%

According to the results, only one configuration exceeded the required threshold – 4 files with a duration of 5 seconds (82.5%). And based on average result, set of 4 files with any duration can also be used (75.83%), the results for the duration of sounds with 2s duration and 10s are close to positive (72.5%).

Sets of other characteristics cannot be correctly identified by people with the required accuracy. In addition, the results were analyzed depending on the sound and the specific participant to determine whether there are additional dependencies. Table 6 represents summarized results with the accuracy of the definition of sounds by sound class and, accordingly, its content.

Table 6

The results of the accuracy of determining the sound by sound type

Name of the sound	4 files			9 files			14 files		
	2 sec	5 sec	10 sec	2 sec	5 sec	10 sec	2 sec	5 sec	10 sec
Car	-	-	-	40%	60%	30%	20%	20%	30%
Shot	-	-	-	70%	90%	80%	50%	70%	60%
Vocal	-	-	-	70%	30%	60%	20%	40%	20%
Chicken	-	-	-	20%	80%	60%	10%	20%	10%
Milk	-	-	-	50%	40%	30%	10%	50%	20%
Piano	50%	80%	60%	30%	50%	40%	30%	40%	20%
Field	60%	70%	50%	80%	70%	80%	40%	60%	50%
Tennis	80%	90%	90%	70%	90%	70%	40%	70%	50%
Trimmer	100%	90%	90%	80%	100%	90%	40%	40%	30%
Ukulele	-	-	-	-	-	-	20%	20%	20%
Jam	-	-	-	-	-	-	10%	20%	30%
Kettle	-	-	-	-	-	-	20%	20%	10%
Train	-	-	-	-	-	-	10%	20%	10%
Siren	-	-	-	-	-	-	40%	50%	30%

Some sounds, such as pianos and fields of 2 and 10 seconds duration, show low results with a small number of sounds – 4. The same situation occurs with more sounds – 9 for the sounds of cars, vocals, chickens, milk and piano – they also show low correct identification results.

There are sounds that show a high percentage of correct identifications, even with a large number of sounds – these are the sounds of tennis and shots. They differ from others in that they have intervals with the maximum value and attenuation during playback.

And some sounds showed high results before adding some other sounds to the experiment. For example, the sound of the trimmer began to show a bad result after adding the sounds of traffic jams, kettle, train, ukulele and siren. This is due to the fact that some sounds are not similar to each other in real life, but are similar when reproduced with a complex.

For example, using the values obtained by the classification module, Figure 5 shows a graph showing the differences between the sounds of the trimmer, kettle and shot.

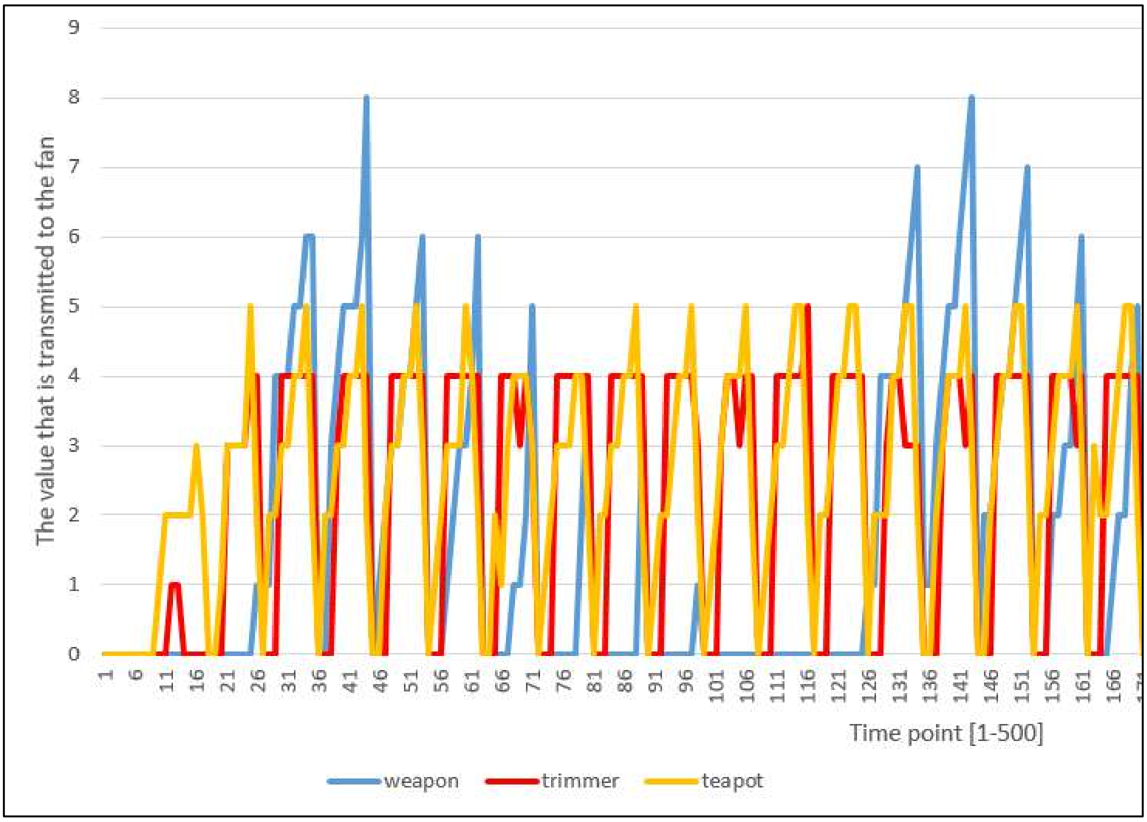

Figure 5: Comparison of the obtained values for different sounds

The figure on the abscissa shows a fragment of 450 generated values for the fans. The values are recorded sequentially on the axis for 9 fans and then cyclically for each study for 5 seconds, and on the ordinate axis recorded the corresponding value for each fan.

The shot sound graph has interval peaks and attenuation, as discussed earlier, and this helps to identify it among others, but for good identification of interval sounds, they must differ in value and intervals. The trimmer and kettle sound graphs are similar, differing by only one on two fans and overlapping over most of the time. In this case, the sounds themselves in their original form are quite different.

Analyzing the data on which the graph is based, we see that the values for the first two fans corresponding to the frequency ranges of 19.53-21.96 kHz and 17.09-19.52 kHz are almost always zero, this is the expected result, because such sounds provide a lot of pressure on a human auditory system and can cause discomfort and are not common in real life.

Table 7 shows the results of correct identifications depending on the participant of the experiment. Where abbreviation “P1-1” and similar ones means the same as in Table 4.

Table 7
Accuracy of sound determination by participants

Test by person	4 files			9 files			14 files		
	2s	5s	10s	2s	5s	10s	2s	5s	10s
P1-1	100,00%	75,00%	100,00%	67,00%	55,56%	66,67%	21,43%	28,57%	28,57%
P1-2	50,00%	75,00%	50,00%	33,00%	66,67%	33,33%	28,57%	35,71%	21,43%
P2-1	50,00%	50,00%	50,00%	44,00%	77,78%	44,44%	28,57%	50,00%	28,57%
P2-2	75,00%	100,00%	75,00%	78,00%	66,67%	77,78%	28,57%	42,86%	28,57%
P3-1	75,00%	100,00%	75,00%	67,00%	44,44%	66,67%	21,43%	35,71%	35,71%
P3-2	75,00%	100,00%	75,00%	78,00%	88,89%	77,78%	21,43%	35,71%	28,57%
P4-1	50,00%	75,00%	50,00%	56,00%	66,67%	55,56%	28,57%	42,86%	28,57%
P4-2	75,00%	75,00%	75,00%	56,00%	88,89%	55,56%	28,57%	35,71%	28,57%
P5-1	75,00%	100,00%	75,00%	56,00%	55,56%	55,56%	21,43%	42,86%	21,43%
P5-2	100,00%	75,00%	100,00%	67,00%	66,67%	66,67%	28,57%	35,71%	28,57%

From these results it was not possible to distinguish a clear relationship. In most cases, the second attempt is slightly better than the first. At the time of the third part of the experiment with 14 sounds, experiments with fewer sounds had already been conducted, but this did not help to better recognize the sounds that were better studied during previous experiments.

5 Conclusions and Discussions

This paper investigated an alternative method for the reproduction and transmission of sound waves – by means of air. For this purpose, a software-hardware complex was developed, which consists of an Arduino MEGA 2560 board, 9 Delta Electronics ASB0412MA fans connected to it.

An algorithm has been developed that uses spectrum analysis based on fast Fourier transform, processes the obtained results, distributes frequencies by fans and calculates the final value. Then this value by means of PWM is converted into an analog signal of voltage supply to the fan, the speed of which is regulated and the air flow interacts with human skin. However, a software implementation of the described algorithm was created using the JavaScript programming language using existing software libraries.

An experiment was performed on the possible use of the obtained air flows with the help of other computer systems. To do this, a decision tree model was created using the ID3 algorithm. However, the classification of sounds thus showed negative results and for this task it is necessary to use more sophisticated approaches.

An experiment with humans has shown that reproducible sounds can be identified if their number is close to 5, the duration is about 3-8 seconds, or the number and duration may be slightly bigger/longer, but the sounds must be very different when played by software-hardware complex or sounds should have intervals.

An important factor for proper identification is the location of the fan in front of the person, the distance to the fans, their size and power. A separate study is required to select the optimal size, power and number of fans. The subjective factor for clogging is the sensitivity of human sensory systems. The effectiveness of identifications can be negatively affected by external factors such as wind.

This method can be used to transfer and identify audio information subject to the limitations outlined in this paper. After additional experiments, this method can be used in the commercial segment, and used in a variety of cinemas that affect a large number of human receptors, live concerts, augmented or virtual reality games or transferring sound information, when it is not possible to play sound due to various factors.

The research method can also be used to convey important information to people with hearing impairments during emergencies such as war. When people are in a bomb shelter or in their own home, the lack of information about what is happening around them can cost them their lives. To study this, it is necessary to conduct additional experiments with military-themed sounds: bombing, sirens, half-hearted military aircraft and helicopters, and so on.

6 References

[1] A. J. Oxenham. How We Hear: The Perception and Neural Coding of Sound, Annual Review of Psychology 69 (2018) 27-50. doi: 10.1146/annurev-psych-122216-011635.

[2] D. Ziobroski, C. Powers. Acoustic Terms, Definitions and General Information, 2005. URL: https://www.ge.com/content/dam/gepower-new/global/en_US/downloads/gas-new-site/resources/reference/ger-4248-acoustic-terms-definitions-general-information.pdf.

[3] A.H. Frey. Human auditory systems response to modulated electromagnetic energy, J. Appl. Physiol (1962). doi: 10.1152/jappl.1962.17.4.689.

[4] A. Pérez-Bellido, K. A. Barnez, L. E. Crommett, J. M. Yau. Auditory Frequency Representations in Human Somatosensory Cortex, Cereb Cortex 28 (2017) 1–14. doi:10.1093/cercor/bhx255.

[5] Z. Lin, C. Di, X. Chen, Y. Hou. Acoustic recognition method in low SNR based on human ear bionics, Applied Acoustics 182 (2021). doi:10.1016/j.apacoust.2021.10821.

[6] J. Vandendriessche, N. Wouters, B. da Silva, M. Lamrini , M. Yassin Chkouri, A. Touhafi. Environmental Sound Recognition on Embedded Systems: From FPGAs to TPUs, Electronics 10 (2021). doi:10.3390/electronics10212622.

[7] G. Proniuk, N. Geseleva, I. Kyrychenko, G. Tereshchenko. Spatial Interpretation of the Notion of Relation and Its Application in the System of Artificial Intelligence. COLINS-2019: 3rd International Conference on Computational Linguistics and Intelligent Systems, Kharkiv: CEUR Workshop Proceedings, 2019, Vol. 2362. – pp. 266-276.

[8] D. M. Alter. Using PWM Output as a Digital-to-Analog Converter on a TMS320F280x Digital Signal Controller, DSP Applications – Semiconductor Group (2008).

[9] Y. Wang, F. Cheng, Y. Zhou, J. Xu. Analysis of double-T filter used for PWM circuit to D/A converter, IEEE 2012 24th Chinese Control and Decision Conference (CCDC) (2012) 2752–2756. doi:10.1109/CCDC.2012.6244437.

[10] K. Smelyakov, S. Smelyakov, A. Chupryna. Advances in Spatio-Temporal Segmentation of Visual Data. Chapter 1. Adaptive Edge Detection Models and Algorithms. Series Studies in Computational Intelligence (SCI), volume 876, Publisher Springer, Cham, 2020, pp. 1–51. doi: 10.1007/978-3-030-35480-0.

[11] L. Tan, J. Jiang, Digital Signal Processing (Third Edition), Elsevier, 2019.

[12] Arduino Mega 2560 Rev3 Specification. URL: https://store.arduino.cc/products/arduino-mega-2560-rev3.

[13] Delta Electronics ASB0412MA-A Specification. URL: https://ru.mouser.com/ProductDetail/Delta-Electronics/ASB0412MA-A?qs=lYGu3FyN48cieCF4OvMBgg%3D%3D.

[14] C. Rogers, C. Wilson, R. Toy, P. Adenot, H. Choi. Web Audio API. (2021). URL: https://www.w3.org/TR/2021/REC-webaudio-20210617/.

[15] Node SerialPort Documentation. URL: https://serialport.io/.

[16] Decision Tree for Node.js. URL: https://www.npmjs.com/package/decision-tree.

[17] D. Cherezov, N. Tukachev. Obzor osnovnyh metodov klassifikacii i klasterizacii dannyh [Overview of the main methods for classifying and clustering data], 2009. URL: http://www.vestnik.vsu.ru/pdf/analiz/2009/02/2009-02-05.pdf.