

UDC 519.711; 681.5; 621.382

DOI: 10.15587/1729-4061.2017.116082

# DEVELOPMENT OF A SYSTEM FOR MONITORING VIBRATION ACCELERATIONS BASED ON THE RASPBERRY PI MICROCOMPUTER AND THE ADXL345 ACCELEROMETER

**A. Holovaty**

PhD, Associate Professor

Department of Information Technologies

Ukrainian National Forestry University

Generala Chuprynky str., 103, Lviv, Ukraine, 79057

E-mail: aholovaty@ntu.edu.ua

**V. Teslyuk**

Doctor of Technical Sciences, Professor

Department of Automated Control Systems\*

E-mail: vasyli.m.teslyuk@lpnu.ua

**M. Iwaniec**

Doctor of Technical Sciences, Professor, Head of Department

Department for Automation of

Technological Processes and Production

AGH University of Science and Technology

al. Mickiewicza, 30, Krakow, Poland, 30-059

E-mail: iwaniec@agh.edu.pl

**M. Mashevska**

PhD

Department of Information Systems and Technologies\*

E-mail: marta.v.mashevska@lpnu.ua

\*Lviv Polytechnic National University

S. Bandery str., 12, Lviv, Ukraine, 79013

*Реалізовано систему моніторингу і аналізу спектру віброприскорень на одноплатному мікрокомп'ютері Raspberry Pi 3 і трьохосовому цифровому MEMS-акселерометрі ADXL345. Система включає драйвер акселерометра під Raspberry Pi та програмне забезпечення для збору і опрацювання даних від датчика прискорення. Програмне забезпечення перетворює вимірні сигнали віброприскорення з часової області в частотну використовуючи алгоритм дискретного перетворення Фур'є (ДПФ) та будує графіки віброприскорень і спектрів*

*Ключові слова: система моніторингу віброприскорень, ADXL345, Raspberry Pi, спектральний аналіз, дискретне перетворення Фур'є*

*Реализована система мониторинга и анализа спектра виброускорений на одноплатном микрокомпьютере Raspberry Pi 3 и трехосном цифровом MEMS-акселерометре ADXL345. Система включает драйвер акселерометра под Raspberry Pi и разработанное программное обеспечение для сбора и обработки данных с датчика ускорения. Программное обеспечение преобразовывает измеренные сигналы виброускорения из временной области в частотную, используя алгоритм дискретного преобразования Фурье (ДПФ) и строит графики виброускорений и спектров*

*Ключевые слова: система мониторинга виброускорений, ADXL345, Raspberry Pi, спектральный анализ, дискретное преобразование Фурье*

## 1. Introduction

Present-day manufacturing processes require continuous control over many parameters of the process equipment. The most important of them are mechanical parameters including mechanical vibrations of the object being studied. Such control is required in various fields of science and technology, for example, to control vibration of crystal-growth units in semiconductor electronics or photolithographic installations in microelectronics. This control is used to monitor vibration of metal-working machines, beat of the processed workpieces and vibration of the cutting tools in the machine-building industry or control vibration of individual car assemblies and the vehicle as a whole in the automotive industry. Vibrations are monitored to detect approach of trains on railroads, control vibration of gas turbine blades in power engineering and

vibration in gas pipelines or control turbine beat in aircraft engineering, etc. [1].

Vibration monitoring systems enable solution of numerous problems associated with vibrations that arise during operation of process equipment (mechanisms, machines, CNC machine-tools) [1–3]. Consequently, development of the systems for monitoring and analysis of vibration in technical objects is a topical issue at present.

## 2. Literature review and problem statement

Analysis of existing literary sources makes it possible to assert that there are two basic methods for measuring vibration parameters: contact methods with a constraint between the sensor and the object under study and non-contact ones, i. e., that are not connected with the object by a constraint [4–7].

Contact methods are the simplest methods for measuring vibration using piezoelectric sensors [4]. Such methods make it possible to measure with a high accuracy in the range of low frequencies and relatively large amplitudes of vibration but due to their high inertia, signal waveform is distorted which makes unfeasible measurement of high-frequency, small-amplitude vibrations. Besides, if weight of the object under study and therefore its detention lag is small, then such a sensor can significantly affect the vibration character introducing extra measurement errors.

These drawbacks are eliminated by the method of open resonator [5]. The essence of this method consists in measuring parameters of the high-frequency resonator which vary according to vibration of the object under study. The resonator has two mirrors of which one is fixed and the other is mechanically connected with the object. Offsets at small vibration amplitudes are recorded by an amplitude method proceeding from variation of the output power. This measurement method requires a steady power supply to the resonator and a high stability of the excitation frequency.

All non-contact methods for measuring vibration are based on probing the object with acoustic and electromagnetic waves. One of the latest solutions is the ultrasonic phaseometry method [6]. The basic idea is to measure the current value of the phase difference between the reference signal of ultrasonic frequency and the signal reflected from the object being studied. This method advantages include low cost and compactness of the apparatus, small measurement time, absence of restrictions to the frequency range and a high accuracy when measuring low-frequency vibrations. High ultrasound attenuation in the air, dependence on the atmospheric conditions, worsening measurement accuracy with the growth of vibration frequency constitute its disadvantages.

Methods of object probing with visible light are widely spread at present [7]. All optical methods are divided into two groups. The first group includes the methods based on recording of the Doppler effect [7]. The easiest of them is the homodyne method that enables measurement of amplitudes and phases of harmonic vibrations but it is impossible to investigate non-harmonic and large vibration amplitudes with this method. These disadvantages can be eliminated using heterodyne methods. But they require calibration and complex measuring tools. Strict requirement to surface quality of the object under study is a material disadvantage of the above methods. But this disadvantage disappears when using holographic methods that form the second group. Holographic methods have high resolution but require complex and expensive equipment. In addition, measurement is rather long-lasting. The general disadvantages of optical vibration measurement methods are their complexity, bulkiness and high cost of equipment, high energy consumption, strict requirements to surface quality of the object under study and to atmosphere (level of humidity, dustiness, etc.). In addition, laser radiation has a detrimental effect on the eyesight of the staff and requires additional precautions and protection.

Various scientific and technical papers are devoted to the development of vibration monitoring and analysis systems. In particular, a system for monitoring and analysis of vibrations occurring in electric motors was developed [8]. The system uses a piezoelectric accelerometer (ICP 603C11) and the National Instruments' NI 6009 data acquisition board. Vibration signals are taken from various parts of electric

motors and transmitted to a computer through a data acquisition board. A virtual tool that enables real-time monitoring and Fourier analysis of the signals received from the vibration sensor was implemented in the LabVIEW system [10].

A built-in system for monitoring pump unit vibrations based on the Microchip Co. microcontroller has been developed in [9]. Its data acquisition and analysis software were optimized for testing pumps with turbo-supercharging at rotational speeds of up to 2000 rpm.

The software task was limited by auto-diagnostics but it can be configured for individual and manual vibration diagnostics. The only limitation of the system is the accelerometer characteristics. The authors have carried out a large number of measurements on various turbines to determine operating conditions of pumping units using the developed system [9].

A method for determining offset and speed from acceleration signals taken from accelerometers was described in [10], and a method for monitoring reliability of bridge structures using MEMS accelerometers was developed in [11].

A system for monitoring machine tools and machining processes was developed in [12]. The vibration signal acquisition system is based on the Arduino microcontroller connected to a computer via an USB port. A specially developed software under LabVIEW reads and processes data in real time.

Suitability of micro-electromechanical (MEMS) accelerometers for monitoring state of CNC machines was studied in [13].

Tests were conducted on an actual CNC machine working in a typical industrial workshop. It was shown that MEMS sensors can be a good alternative to standard vibration sensors because they do not require heavy electrometric amplifiers. Such sensor must be chosen in accordance with the requirements of application and the suitability test result. MEMS sensors with special packaging can be used in tough operation conditions.

A number of authors use Arduino microcontrollers in a hardware-software system for measuring mechanical vibrations [14]. The ADXL335 accelerometers were used as vibration sensors. The developed system was used in studying and monitoring vibration of a vacuum pump. The studies have shown feasibility and expediency of developing built-in real-time vibration monitoring systems using non-expensive hardware and software.

Application of another type of microcontroller is described in [15]. In particular, a system for monitoring vibrations in rotary equipment and machine tools based on the PIC-18F6520 microcontroller and the ADXL322 accelerometer was proposed. The study was carried out to test the possibility of recording frequency peaks occurring in various malfunction cases of failure.

The best approaches and technical solutions among the above-described solutions for the range of tasks involved include the methods described in [8], [12] and [13-15]. The study methods were based on conducting experiments with various technical objects including measurement of vibration parameters, their processing and analysis with the help of the self-developed hardware and software systems. The developed systems are closed and expensive which does not let them to extend their functionality or modify to respective operational needs.

Thus, the analysis of existing approaches and technical solutions has led to the development of a qualitatively new,

inexpensive, open hardware and software system for monitoring vibrations in real time. Such a system was built of available and inexpensive components, free software with an ability to modify or extend its functionality in accordance with requirements of the application field.

### 3. The aim and objectives of the study

The study objective was to develop and study a system for monitoring and analysis of the vibration acceleration spectrum (MAVAS) which is characterized by its low cost and ability of rapid device modification according to the technological conditions of the measurement environment.

To achieve this objective, the following tasks had to be solved:

- develop structure and operation algorithm of the MAVAS system based on a modular principle;
- develop a physical model of the MAVAS system built with the use of general-purpose hardware components and characterized by low-cost technical solutions;
- develop specialized software for the MAVAS system which enables realization of a manifold functionality.

### 4. Development of the MAVAS system structure

The developed MAVAS system structure shown in Fig. 1 includes the following main components:

- the Raspberry Pi 3, Model B single-board microcomputer (MC) [16, 17] intended for acquisition and processing of the data taken from a vibration sensor (accelerometer);
- the Analog Devices’ triaxial digital ADXL345 accelerometer used as a vibration acceleration sensor [18, 19];
- a flash memory for installation of OS, drivers, software and data storage;
- a 7” color graphics TFT display for viewing information arriving from the Raspberry Pi 3 MC;
- wireless communication interfaces for a remote information exchange with the PC.

The developed system operation algorithm includes the following steps (Fig. 2):

*Step 1.* Initialize the ADXL345 accelerometer on the I2C bus [20].

*Step 2.* Read the value from the ADXL345\_DEVID register at 0x00 address and check the accelerometer number which should be equal to ADXL345\_ID 0xE5.

*Step 3.* Adjust the ADXL345 accelerometer.

*Step 4.* Switch the sensor to the operating mode (0x00 byte record), set the passband of 100 Hz (0x0A byte record) using the ADXL345\_SetRegisterValue(ADXL345\_BW\_RATE, (0x00|ADXL345\_RATE (0x0A))) function.

*Step 5.* Set the data format, measurement range 16g, interruption, equalization of the measured values using the ADXL345\_SetRegisterValue(ADXL345\_DATA\_FORMAT, (0x00|ADXL345\_RANGE(ADXL345\_RANGE\_PM\_16G)|ADXL345\_FULL\_RES)) function.

*Step 6.* Conduct ADXL\_NO\_SAMPLES measurements (samples) of acceleration on all three axes x, y, z at a step of  $t_s = 1e6/F_s$  (where  $F_s$  is the sampling rate) and record their values in two-byte integer 1024 size arrays.

*Step 7.* Calculate real acceleration values  $a_x, a_y, a_z$  in the time interval from 0 to  $1e6/F_s$  by the formula  $axx[i] \times g / LSB \times 9.80665$ ,  $xx$  is measurement axis x, y or z. The g/LSB ratio at the set bit FULL\_RES = 1 in the maximum resolution mode is 0.0039. Record the obtained values in arrays.

*Step 8.* Record the calculated acceleration values in  $fp\_samples\_ax, fp\_samples\_ay, fp\_samples\_az, fp\_samples\_a$  files.

*Step 9.* Call the `adxl345_spectrum_analysis(ADXL_NO_SAMPLES, FS)` function to calculate the DFT [21, 22].

*Step 10.* Record the obtained spectrum data into files and plot the functions of acceleration dependence on time and spectra.

The developed structure of the MAVAS system is based on the modular principle which makes it possible to quickly and efficiently modify the MAVAS system in the process of its improvement. Besides, the low cost of the design solution is ensured by the use of cheap general-purpose hardware components and free software.

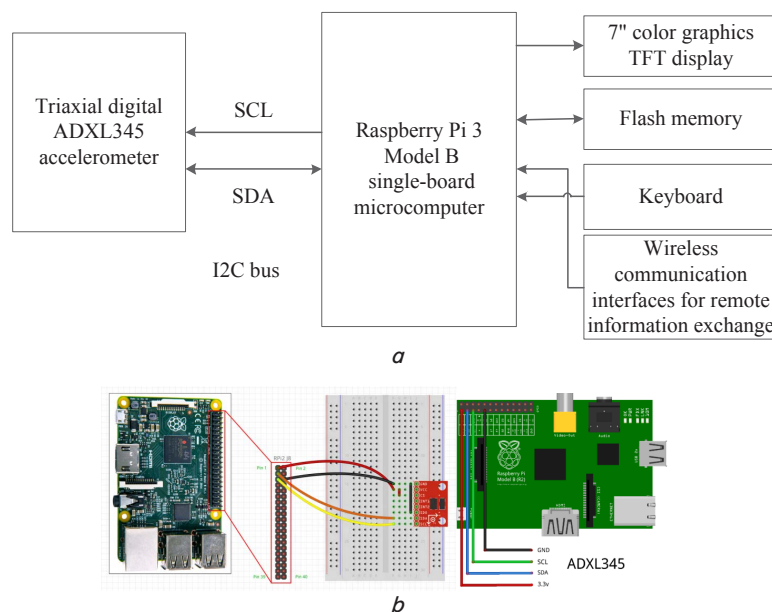


Fig. 1. Designed structure of the MAVAS system and the circuit for connecting the ADXL345 accelerometer to the Raspberry Pi MC: *a* – the MAVAS block diagram; *b* – the MAVAS physical model

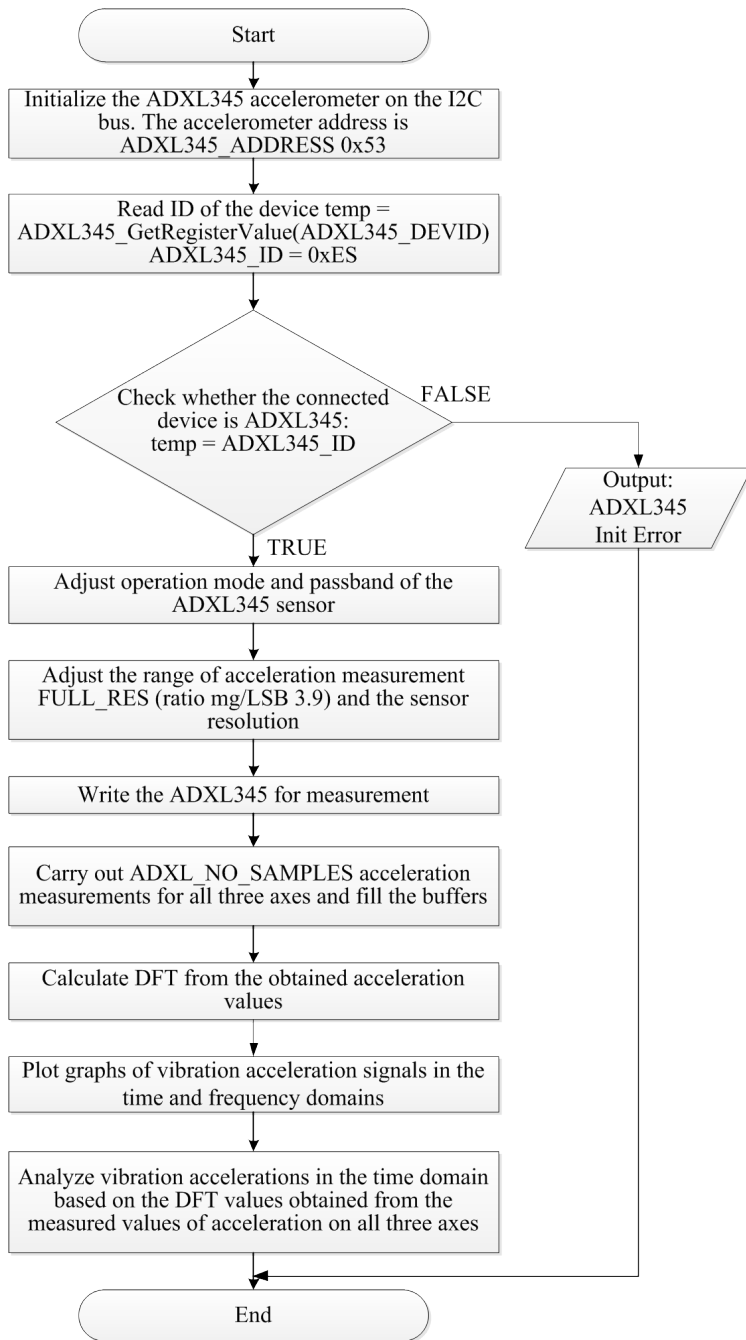


Fig. 2. Algorithm of the system operation

**5. Development of hardware for monitoring and analysis of vibration accelerations**

**5.1. Construction of the MAVAS system physical model**

The developed physical model of the MAVAS system includes the following main elements:

- a single-board Raspberry Pi 3 microcomputer;
- a triaxial digital accelerometer;
- a liquid-crystal graphic display.

The system hardware is based on the single-board Raspberry Pi 3, Model B microcomputer and the triaxial digital ADXL345 accelerometer (Fig. 3) [23, 24]. The ADXL345 accelerometer is installed on the monitored object (e. g. on a milling head of a CNC milling machine) and connected

to the I2C bus to the Raspberry Pi MC. The microcomputer takes data from the sensor and processes them. The ADXL345 accelerometer is used as a sensor for measuring vibrations. The ADXL345 is a miniature triaxial digital accelerometer from Analog Devices Co. with low power consumption, high resolution (13 bits) and a range of acceleration measurement up to ±16 g. The range of measurements may be selected from the row: ±2 g, ±4 g, ±8 g and ±16 g. The result of measurements can be read byte-by-byte via the SPI digital interface (3- or 4-wire type) or I2C in a form of 16-bit data.

The ADXL345 accelerometer belongs to the class of capacitive accelerometers with a passband of 0.05...1600 Hz. This device is ideal for measuring dynamic accelerations, low-frequency vibrations, static accelerations of gravity, motion and angles of inclination. The passband characterizes the sensor ability to sense acceleration changes occurring at high frequencies (e. g., vibration with a frequency of 1000 Hz). This characteristic is influenced by the sampling rate of the built-in ADC accelerometer which should be at least twice the passband. The maximum sampling rate for the ADXL345 accelerometer is 3200 Hz [24].

The ADXL345 [24] accelerometer is powered from a 2.0...3.6 V power source and consumes 25...130 µA in the measuring mode and 0.1 µA in the standby mode. It leaves resolution to the user's choice. Fixed resolution is 10 bits and the maximum-resolution mode is available when resolution increases with an increase in the range of acceleration (g). Maximum value is up to 13 bits when measuring acceleration ±16 g at a constant sensitivity of 4 mg/LSB in all ranges of g measurement. The sensor possesses functions of detecting single and double impulses and monitoring activity/inactivity. The free fall detection function; the SPI (3- and 4-wire types) and I2C interfaces [20]; ability to flexibly set interrupt modes with the choice of any (of 2 possible) interruption outputs; the measurement range as well as the passband can be selected by issuing a corresponding command. The sensor is operable in a wide temperature range from -40 °C to +85 °C and has a high impact resistance, up to 10,000 g.



Fig. 3. The triaxial ADXL345 MEMS- accelerometer

Such a built-in system provides for a constant real-time monitoring of vibration, e. g. vibration of the working machine, and analysis of the vibration parameters. The system

issues warning messages or stops the machine in a case of inadmissible vibrations, thus preventing possible breakdowns and accidents. The system also provides the user with sufficient information so that he can identify possible problems and take preventive measures based on the analysis of the vibration spectrum for a concrete machine.

The main advantages of such system at the level of production management are as follows: prevention of failures of the spindle and the machine tool as a whole and protection of the cutting tool and the workpiece. The system prevents damage to the especially important machine assemblies due to the use of the rapid impact response function. Such monitoring system can be integrated into a CNC system. It can warn in a timely manner on a potential malfunction which enables the user to take timely preventive measures.

Monitoring of total vibration should be carried out both in time and in frequency bands. In the time range, the system continuously monitors the machine status in real time based on a broadband measurement of vibration velocities, vibration accelerations and vibration offsets.

In the frequency range, a number of resettable margins for warning and alarm signals in various frequency ranges can be set. These ranges cover the entire spectrum of vibration and allow the user to make analysis of the vibration level in the controlled area to know whether there is an override of the entire frequency spectrum.

The system allows users to perform vibration analysis in the time range in three separate resettable frequency ranges. The first of them is a low-frequency range with an adjustable bandwidth to determine the boundary of vibrational offsets within the specified range. The second one is the frequency band with adjustable bandwidths to determine boundary of the vibration velocity within the specified range. The third high-frequency range with an adjustable bandwidth is intended for determining the vibration acceleration boundary within the specified range.

The bands can be selected based on the machine tool configuration, the operating conditions and the characteristics of expected failures.

### 5. 2. Analysis of vibrations in the frequency range

The system applies the formulation with the use of fast Fourier transform (FFT) for recording of the vibration spectrum and real-time analysis of the frequency range [21 23].

The following functions are provided in the system: the saving function, the averaging function and the function of comparative analysis. The saving function is used for recording and storing characteristic vibration spectra. The averaging function is used to find the average value of the measured spectra in order to construct a total spectrum of characteristic vibrations. The function of comparative analysis is used to detect changes in the machine working conditions.

### 5. 3. Adjusting the accelerometer

To adjust the ADXL345 accelerometer, it should be connected to the Raspberry Pi MC via the I2C interface (Fig. 4). A standard transmission rate of 100 kHz and a

high transmission rate of 400 kHz are supported for the I2C interface. The ALT\_ADDRESS line is responsible for the accelerometer address on the I2C bus. At a high logic level in the line, the accelerometer is assigned a 7-bit address with a 0011101 (0x1D) value and at a low level, the address value is 1010011 (0x53). For a proper operation of the interface, the SDA and SCL lines should be connected to the supply voltage through the Rp (4.7 kOhms) resistors. Because of limitation of the data transfer rate to 100 kHz, the maximum acceleration measurement frequency at the I2C interface should not exceed 800 Hz.

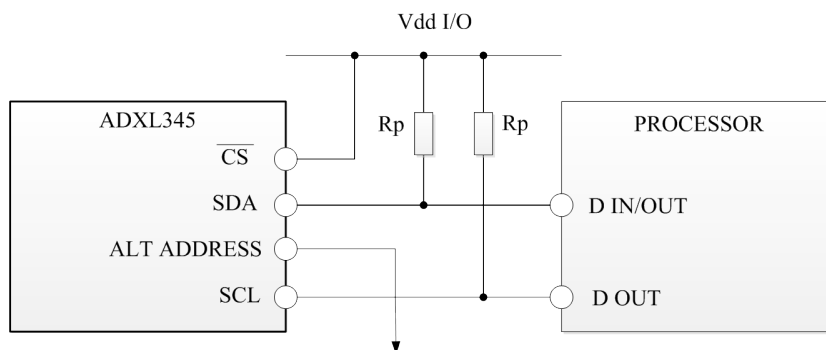


Fig. 4. Circuit for connecting the ADXL345 accelerometer to the I2C interface (address 0x53)

The information exchange with the accelerometer is carried out according to the following algorithm. First, the Raspberry Pi MC sends (via the I2C bus) the address of the device register from which data will be read or written to it. Then, data are sent for recording to this register or a specified number of bytes are read from the device.

Initialization of the ADXL345 accelerometer includes three steps:

1. Activation of the measurement mode by setting 3 bits in the POWER\_CTL register (address 0x2D). The value 0x08 is recorded to it.
2. Definition of the data format in the DATA\_FORMAT register (address 0x31).
3. Recording offsets OFSX, OFSY, OFSZ in the OFFSET registers. The OFFSET register addresses are 0x1E (OSFX), 0x1F (OSFY), 0x20 (OSFZ).

With the help of the DATA\_FORMAT register, format of the output data for the DATA\_X, DATA\_Y, DATA\_Z registers and a number of other functions are adjusted. Using the FULL\_RES bit of the DATA\_FORMAT register, resolution of the acceleration measurement is adjusted. When setting the bit, the extended resolution mode is switched on which depends on the boundaries set by the Range bits and the result of the measurement is calculated based on the factor of 3.9 mg/LSB. The zero-bit value sets the fixed resolution of 10 bits, and the measurement result along with the calculation factor will depend on the selected boundary. The Justify bit of the DATA\_FORMAT register sets the method for aligning the measurement result inside the DATA\_X, DATA\_Y, DATA\_Z registers: the bit value equal to 1 sets the left alignment while the bit value equal to 0 sets the right alignment.

The output data frequency is adjusted in the BW\_RATE register. The adjustment parameters are shown in Table 2 and Table 3. The Rate bits of the register in Table 2 are responsible for the output data frequency.

Table 1

Setting the range of acceleration, g

Bit value		The range of the measured acceleration, g
D1	D0	
0	0	±2 g
0	1	±4 g
1	0	±8 g
1	1	±16 g

Table 2

Register 0x2C – BW\_RATE (read/write)

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	LOW_POWER	Rate			

Table 3

Adjustment of the passband and the sampling frequency of the ADXL345 accelerometer, BW\_RATE register (address 0x2C)

Frequency of measuring and updating the output data (Hz)	Passband (Hz)	Binary code of D0...D3 bits of the BW_RATE register
3200	1600	1111
1600	800	1110
800	400	1101
400	200	1100
200	100	1011
100	50	1010
50	25	1001
25	12,5	1000
12.5	6.25	0111
6.25	3.13	0110
3.13	1.56	0101
1.56	0.78	0100
0.78	0.39	0011
0.39	0.20	0010
0.20	0.10	0001
0.10	0.05	0000

At the next step, the frequency range of measurement must be adjusted by setting 3 bits: D3, D1 and D0. When the FULL\_RES bit is allowed, the device operates at a maximum resolution of 3.9 mg/LSB. No matter which measurement range is defined, one bit represents acceleration of 3.9 mg. If the FULL\_RES bit is not set to 1, the ADXL345 accelerometer will work in a 10-bit mode and the bits of range D1 and D0 will determine the value of mg/LSB. The mg/LSB values for various configurations of the acceleration measurement ranges are shown in Table 4.

Table 4

The mg/LSB values for various ranges

Range	mg/LSB
Total range at the bit FULL_RES=1	3.9
±2 g, 10-bit mode	3.9
±4 g, 10-bit mode	7.8
±8 g, 10-bit mode	15.6
±16 g, 10-bit mode	31.2

By writing the value 0x0B to the DATA\_FORMAT format register, a range of ±16 g will be selected with a

maximum resolution of 13 bits. Now we have a working accelerometer without the set values of offset. Offsets to compensate for the measured acceleration values can be set for each axis in OFFSET registers (OFSX, OFSY, OFSZ) or by programming. The offsets are stored in a format of a two's complement of two bytes and automatically added to the values in the output data registers. The offsets must be calculated separately for each accelerometer. Each accelerometer is calibrated individually and has its own unique offset values. The initialization code has the follows appearance:

```

unsigned char status = 1, temp=0;
fd = wiringPiI2CSetup(ADXL345_ADDRESS); // initialization of a device on the I2C bus
temp = ADXL345_GetRegisterValue(ADXL345_DEVID);
if (temp != ADXL345_ID)
    status = 0;
printf("ADXL345 Device ID: 0x%x, Status: %u\n", temp, status);
ADXL345_SetRegisterValue(ADXL345_BW_RATE, (0x00|ADXL345_RATE(0x0A)));
ADXL345_SetRegisterValue(ADXL345_DATA_FORMAT, (0x00|ADXL345_RANGE(ADXL345_RANGE_PM_16G)|ADXL345_FULL_RES));
ADXL345_SetRegisterValue(ADXL345_FIFO_CTL, (0x00|ADXL345_FIFO_MODE(0x2)|ADXL345_SAMPLES(0x1E)));
ADXL345_SetRegisterValue(ADXL345_INT_MAP,(0x00|~ADXL345_WATERMARK));
ADXL345_SetRegisterValue(ADXL345_INT_ENABLE,(0x00|ADXL345_WATERMARK));
ADXL345_SetPowerMode(0x1);
    
```

Now, data can be read from the accelerometer. The raw values of the measured accelerations for each of the X, Y, Z axes are stored in the DATA0, DATA1, DATAY0, DATAY1, DATAZ1 and DATAZ1 registers which are read-only registers. The DATA0, DATA1, DATAY0, DATAY1, DATAZ0, DATAZ1 registers are eight-bit data registers with addresses 0x32 to 0x37. The acceleration value for each axis is represented by two bytes: DATAx0 contains a low-order byte of data, and DATAx1 contains a high-order data byte where x is the X/Y/Z measurement axis. The DATA0 (0x32) and DATA1 (0x33) registers contain the source data along the X-axis. The DATAY0 (0x34) and DATAY1 (0x35) registers contain the Y-axis output data. The DATAZ0 (0x36) and DATAZ1 (0x37) registers contain Z-axis output data. It is recommended that all 6 bytes be read from the registers in one session in order to prevent possible data loss since the contents of the registers may vary between separate operations. The low-order and the high-order bytes of data read from the registers are converted to the 16-bit format of the acceleration values for each axis. The data reading function for each axis is as follows:

```

void ADXL345_GetXyz(int16_t* x, int16_t* y, int16_t* z)
{
    *x = ADXL345_GetRegisterValue(ADXL345_DATAX1) << 8; // read the high-order byte of the X axis data
    *x += ADXL345_GetRegisterValue(ADXL345_DATAX0); // read the low-order byte of the X axis data
}
    
```

```

*y = ADXL345_GetRegisterValue(ADXL345_
DATAY1) << 8; // read the high-order byte of the Y axis
data
*y += ADXL345_GetRegisterValue(ADXL345_
DATAY0); // read the low-order byte of the Y axis data
*z = ADXL345_GetRegisterValue(ADXL345_
DATAZ1) << 8; // read the high-order byte of the Z axis
data
*z += ADXL345_GetRegisterValue(ADXL345_
DATAZ0); // read the low-order byte of the Z axis data
}

```

At the next stage, data in a 16-bit format must be converted to g values. To do this, the data should be multiplied by a defined constant coefficient which varies depending on the range of measurement and the accelerometer resolution. In this work, a mode with a maximum resolution of 13 bits was used which has a coefficient equal to 3.9 mg/LSB. Consequently, this means that the data must be multiplied by 0.0039 to convert them into acceleration values in g units ( $1\text{ g}=9.80665\text{ m/s}^2$ ). An example of software implementation of this operation is given below.

```

void ADXL345_GetAccXYZ(double* acc_x, double*
acc_y, double* acc_z)
{
    int16_t x, y, z;
    ADXL345_GetXYZ(&x, &y, &z);
    // +/-16g 13 bit mode 3.9 mg/LSB
    *acc_x = (double) x * ADXL345_MG_LSB_
FULL_RES / 1000 * 9.80665;
    *acc_y = (double) y * ADXL345_MG_LSB_
FULL_RES / 1000 * 9.80665;
    *acc_z = (double) z * ADXL345_MG_LSB_
FULL_RES / 1000 * 9.80665;
}

```

#### 5.4. The features of accelerometer calibration and setting of OFFSET values

Accelerometers are mechanical structures with freely moving elements. These moving elements can be very sensitive to mechanical influences (shocks, shakes), much more sensitive than the electronics itself. Offset at 0 g is an important metrics of an accelerometer since it defines a threshold for measuring real acceleration. Additional measurement errors occur when mounting the system with an accelerometer. These errors can be caused by stresses in the printed circuit board appearing during mounting, usage of various compounds to the components. Therefore, calibration is recommended following the system assembly to compensate for the impact of above factors.

The easiest way to calibrate the accelerometer consists in averaging the values of measurements (samples) using the so-called scheme with a single point of calibration. In the scheme with a single point of calibration, the system with an accelerometer is oriented so that one axis, as a rule, the Z axis, is located in the 1g gravitation field and the X and Y axes is located in the 0g field. It is recommended that at least 10 samples be taken at intervals of 0.1 s at a frequency of 100 Hz. These values are stored as  $X_{0g}$ ,  $Y_{0g}$ , and  $Z_{+1g}$  for 0 g measurements in X, Y axes and 1 g measurement in the Z axis.

The values measured for  $X_{0g}$  and  $Y_{0g}$  are offsets along the X and Y axes. Compensation is done by subtraction of these

values from the accelerometer output data values to get a real acceleration:

$$X_{ACTUAL}=X_{MEAS}-X_{0g}, \quad (1)$$

$$Y_{ACTUAL}=Y_{MEAS}-Y_{0g}, \quad (2)$$

where  $X_{ACTUAL}$ ,  $Y_{ACTUAL}$  are the real values of acceleration;  $X_{MEAS}$ ,  $Y_{MEAS}$  are the measured acceleration values;  $X_{0g}$ ,  $Y_{0g}$  are offsets (the values in the absence of rotation around any axis).

Since the Z-axis measurements are carried out in a gravitational field at +1 g and the calibration scheme assumes an ideal sensitivity  $S_Z$  for the Z axis, this value is subtracted from  $Z_{+1g}$  to obtain offset along the Z axis which is then subtracted from the following measured values to obtain real acceleration values:

$$Z_{0g}=Z_{+1g}-S_Z, \quad (3)$$

$$Z_{ACTUAL}=Z_{MEAS}-Z_{0g}. \quad (4)$$

The ADXL345 accelerometer can automatically compensate the output using the values recorded in the special OFFSET registers. The registers OFSX (address: 0x1E), OFSY (address: 0x1F), OFSZ (address: 0x20) are intended to represent offsets along the X, Y and Z axes, respectively. The contents of each register are added to the measured acceleration value for the corresponding axis and the result is placed in the DATA registers. The register has a scale factor of 15.6 mg/LSB and is independent of the selected acceleration measurement range. The acceleration of 15.6 mg (calculation factor of 15.6 mg/LSB) corresponds to the low-order significant bit taking into account the sign. For the value 0x7F, approximately +2g is obtained and -2 g for 0x80, respectively. Since the value placed in the registers is added it is necessary to place a negative value in them in order to discard the value of positive offset and vice versa for a negative offset. For example, let us assume that the ADXL345 accelerometer is adjusted with a maximum resolution of 13 bits and a sensitivity of 256 LSB/g ( $\pm 2\text{ g}$ ). The system is oriented so that the Z axis is located in the gravitational field and the measured initial values of acceleration on the X, Y, and Z axes are +10 LSB, -13 LSB and +9 LSB, respectively. Using the previous equations,  $X_{0g}$  is +10 LSB,  $Y_{0g}$  is -13 LSB, and  $Z_{0g}$  is +9 LSB. Each low-order significant bit (LSB) of the output at a maximum resolution is 3.9 mg or one fourth of the low-order bit of the offset register. Since the value in the register is added, the values are taken with an opposite sign and rounded to the nearest value of the low-order meaningful bit of the offset register.

$$X_{OFFSET}=-\text{Round}(10/4)=-3\text{ LSB}, \quad (5)$$

$$Y_{OFFSET}=-\text{Round}(-13/4)=3\text{ LSB}, \quad (6)$$

$$Z_{OFFSET}=-\text{Round}(9/4)=-2\text{ LSB}. \quad (7)$$

These values are programmed into OFSX, OFSY and OFXZ registers, respectively, as 0xFD, 0x03 and 0xFE. Like all registers in the accelerometer, the ADXL345, OFFSET registers do not store the values recorded to them when power is turned off. Turn off and repeated turn on of the accelerometer feed return the OFFSET registers to 0x00

default values. Since the method of absence of rotation, or the single-point method, assumes the perfect sensitivity along the Z axis, there are no errors caused by sensitivity in presence of the offset error. For example, if the actual sensitivity is 250 LSB/g, then the offset will be 15 LSB but not 9 LSB. To reduce this error, additional measurement points with an axis Z located in the 0 g field can be used and the 0 g measurement can be used in the  $Z_{ACTUAL}$  equation.

Thus, the designed hardware is based on the use of a single-board Raspberry Pi MC, digital triaxial ADXL345 accelerometers, a flash memory and a liquid-crystal graphic display. All this ensures execution of all functions at a low cost.

---

## 6. Development of software for acquisition and processing of data from the ADXL345 accelerometer

---

To work with the ADXL345 acceleration sensor (accelerometer), a driver and special software have been elaborated. The driver reads data from the accelerometer, processes them and performs adjustment of the sensor in accordance with the user's needs. The acceleration values obtained for each axis are written to the ax.dat, ay.dat, and az.dat files, respectively. The accelerometer driver was developed in the C language with the use of the wiringPiI2C library and the gcc compiler [25].

To convert the vibration acceleration signal obtained from the sensor from the time domain to the frequency domain, a special software was developed that uses the discrete Fourier transform (DFT) algorithm [21, 22]. The software for calculating the DFT uses the FFTW software library [24]. The spectral data obtained from the DFT signals of vibration acceleration for each axis are recorded in the spectrum\_ax.dat, spectrum\_ay.dat and spectrum\_az.dat files, respectively. Shell scripts accel\_plot.sh, spectrum\_plot.sh, spectrum\_plot\_all.sh were worked out to execute graph plotting. The software was developed for the Raspbian Jessie operating system. Raspbian is a free OS built on a Unix-like Debian operating system optimized for the hardware of the single-board Raspberry Pi MC. The developed software provides the user with system interface, data acquisition and analysis, data storage in files of Micro SD memory card and visual representation of data as well as communication with the PC in the net. The developed software makes it possible to directly compare relative intensities of each frequency component. The software can handle incoming information from multiple accelerometers connected to the Raspberry Pi MC which enables a multi-channel measurement and analysis of their results. The software records and stores vibration information from each channel for its further analysis and comparison. This feature allows the user to display intensity of vibrations in real time as well as viewing contents of the previously recorded files.

Some features of the developed ADXL345 accelerometer driver and special software are given below:

```

/*! Reading the value (byte) from the register */
unsigned char ADXL345_GetRegisterValue(unsigned
char registerAddress);
/*! Writing the data (byte) to the register */
void ADXL345_SetRegisterValue(unsigned char regis-
terAddress, unsigned char registerValue);

```

```

/*! Initializing the I2C accelerometer and checking pres-
ence of the device */
unsigned char ADXL345_Init(char commProtocol);
/*! Transferring the device to a standby/measure mode */
void ADXL345_SetPowerMode(unsigned char pwr-
Mode);
/*! Reading the output data for each axis */
void ADXL345_GetXYZ(int16_t * x, int16_t * y, in-
t16_t * z);
/*! Reading of the output data for each axis of the accel-
erometer */
void ADXL345_GetAccXYZ(double* acc_x, double*
acc_y, double* acc_z);
/*! Enabling/disabling activity detection */
void ADXL345_SetActivityDetection(unsigned char
actOnOff, unsigned char actAxes, unsigned char actAcDc,
unsigned char actThresh, unsigned char actInt);
/*! Setting offset for each axis (Offset calibration) */
void ADXL345_SetOffset(unsigned char xOffset, un-
signed char yOffset, unsigned char zOffset);
/* Calculating the DFT of vibration acceleration */
void adxl345_spectrum_analysis(unsigned int no_sam-
ples, unsigned int fs);
void delay_microseconds(int delay_us).

```

The developed software is based on the use of a free software, namely: the Raspbian Jessie operating system; the gcc compiler; the program module driver for working with the ADXL345 accelerometer via the I2C interface and the FFTW software library (for calculating the DFT). Moreover, the built-in software enables real-time study of the vibration effects on an object, determination of amplitudes and frequencies, plotting graphs of variation of vibration acceleration over time, calculation of the DFT and obtaining spectra. From the spectra obtained, it is possible to determine at which frequencies the amplitudes of vibrations are maximal which will make it possible to identify vibration sources and take measures for their attenuation or reduction.

---

## 7. Discussion of the results obtained in the study of the physical model of the vibration acceleration monitoring system

---

The results of the study are graphically presented in Fig. 5, 6. Fig. 5 shows the change in vibration accelerations obtained from the accelerometer along the X, Y, Z axes and the total value of vibration acceleration along all three axes, respectively. The graphs in Fig. 6 show vibration acceleration spectra for each axis (X, Y and Z) and the total spectrum for all three axes.

Analysis of the obtained results makes it possible to state that the largest peak of the vibration acceleration amplitude falls on the Y axis in a range of frequencies from 10 to 40 Hz. Since the experiment was performed with vibrations at a frequency of 20 Hz, the result of the spectral analysis is quite reliable. In the future, it is necessary to calibrate the accelerometer and carry out experiments with several acceleration sensors by mounting them on the CNC milling machine head.

The results obtained in the test study make it possible to state correctness of the developed software and feasibility of the designed system for monitoring vibration accelerations.



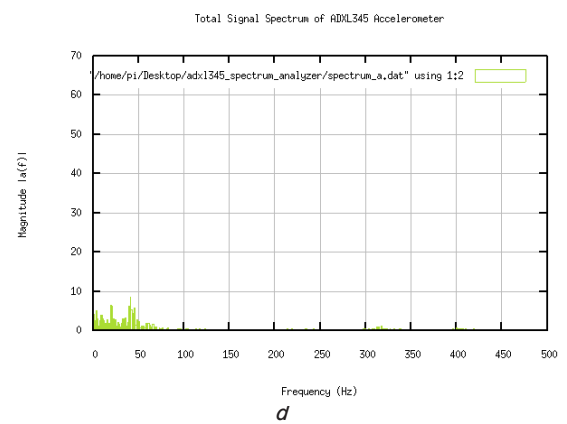
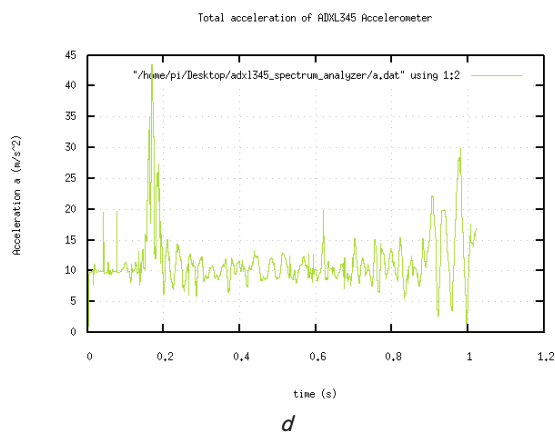
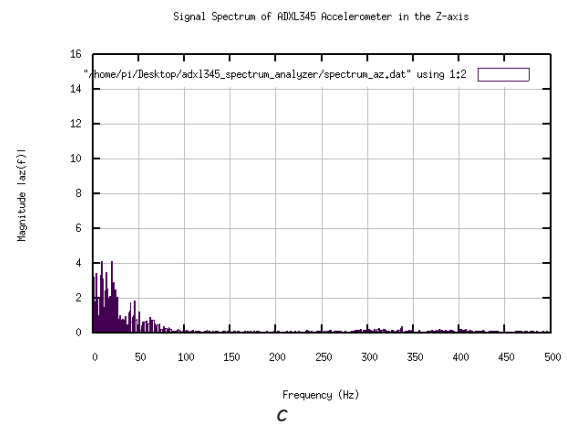
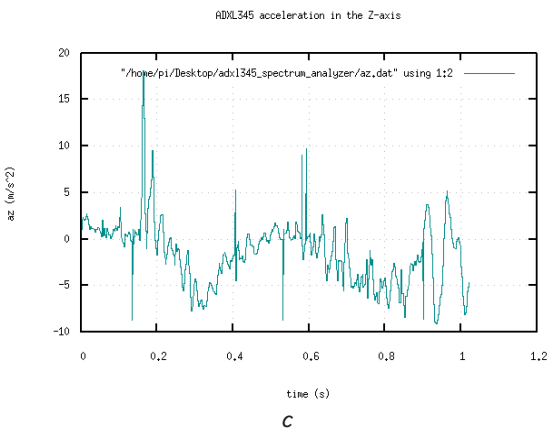
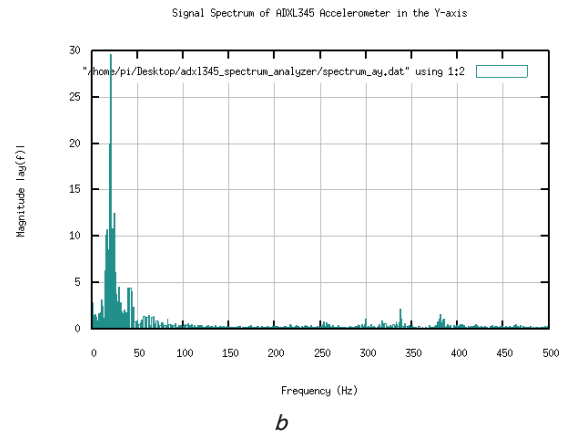
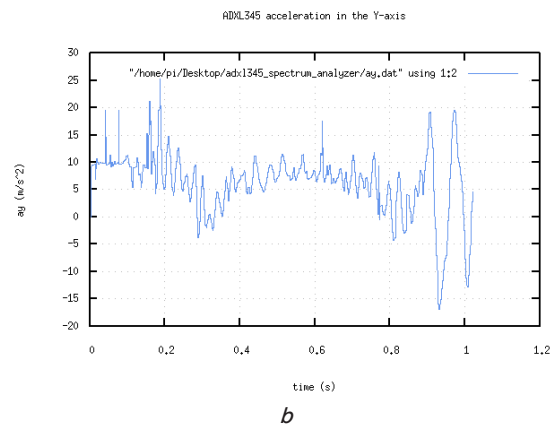
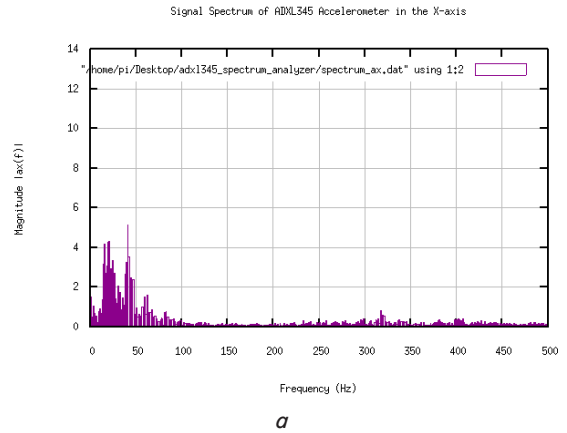
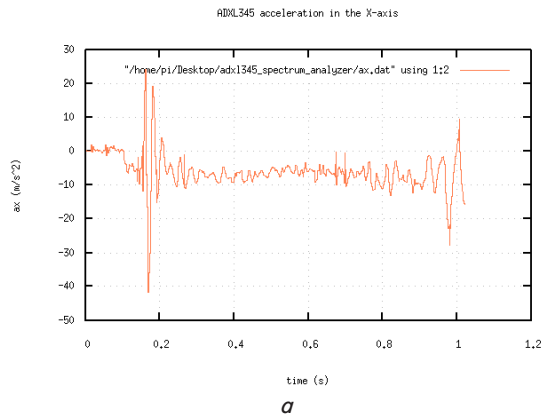


Fig. 5. Change of the output signal of vibration acceleration: *a* – along the X axis; *b* – along the Y axis; *c* – along the Z axis; *d* – total value of vibration acceleration for all three axes

Fig. 6. Spectrum of the vibration acceleration signal: *a* – along the X axis; *b* – along the Y axis; *c* – along the Z axis; *d* – total spectrum of the vibration acceleration signal

Thanks to the use of a digital acceleration sensor with I2C and SPI interfaces, it is possible to connect several sensors with various passbands to the bus and monitor vibrations in various locations of the object under study in real time. Measurements from several accelerometers on the I2C bus are possible when connecting their Alt Address outputs to the corresponding Raspberry Pi connections with a programmed control. The Alt Address pin of the addressed accelerometer should be set to a high level (the Alt Address pins of other accelerometers should be set to a low level) and only this accelerometer will respond to the 7-bit address 0x1D of the device.

In future, the software developed for acquisition and processing of vibration acceleration data can be easily adapted using an appropriate driver for working with other vibration sensors having different frequency ranges of vibration recording.

The advantages of this study compared with its counterparts consist in the use of inexpensive digital sensors to measure vibration accelerations and the single-board Raspberry Pi MC with a small power consumption and the ability of autonomous and remote data exchange via wireless communication means.

In this system, the accelerometer is connected to the Raspberry Pi MC on the I2C bus. The ADXL345 accelerometer supports two modes of data exchange via the I2C bus: the standard mode at 100 kHz and the fast mode at 400 kHz. In the mode of 100 kHz, the sampling rate of the vibration acceleration signal should not exceed 200 Hz, that is, the passband of the input signal of vibration acceleration is from 0 to 100 Hz. In the mode of fast exchange (400 kHz), via the I2C bus, the maximum sampling rate is 800 Hz and the passband is up to 400 Hz, respectively. Disadvantage of this study is that when connecting the sensor to the I2C bus, it is possible to measure vibrations only in the frequency band from 0 to 400 Hz. Therefore, to measure vibration accelerations in the band above 400 Hz and up to 1.6 kHz, the accelerometer should be connected to a high-speed SPI bus that will enable data exchange at a maximum frequency of 5 MHz.

The use of sampling frequencies of 3200 Hz and 1600 Hz is recommended with an SPI bus at a communication frequency greater than or equal to 2 MHz. The input signal sampling rate of 800 Hz is only recommended for communication speeds greater than or equal to 400 kHz. For example, the minimum recommended exchange rate at a sampling frequency of 200 Hz is 100 kHz. Operation at the output data frequencies above the recommended maximum can result in undesirable effects associated with acceleration data including data loss or excess noise. The ADXL345 sensor passband is from 0.5 Hz to 1.6 kHz (with sampling frequencies from 0.10 to 3200 Hz, respectively).

---

## 8. Conclusions

---

1. What has been elaborated in this study is the structure and the algorithm of functioning of the system for monitoring and analysis of the vibration acceleration spectrum which operates in a real-time mode and is based on a modular principle that enables fast improvement and modification of the device. The designed system makes it possible to analyze vibration parameters in order to predict and prevent possible accidents, thereby reducing the costs associated with the failure of cutting tools, expensive components and assemblies of CNC machines.

2. The MAVAS system physical model which includes the Raspberry Pi 3, Model B single-board MC, the triaxial digital ADXL345 accelerometer and the liquid-crystal display and is characterized by a low-cost technical solution has been designed and implemented.

3. Specialized software for the system has been developed and implemented. It includes the driver for adjusting, acquisition and processing of accelerometer data and the appropriate software for plotting the vibration acceleration signals in time and frequency domains. The developed software allows the user to realize wide functionality and is free of charge.

---

## References

1. Marne, N. S. Vibration Measurement System with Accelerometer Sensor Based on ARM [Text] / N. S. Marne, M. S. Nagmode, R. D. Komati // *International Journal of Emerging Technology and Advanced Engineering*. – 2014. – Vol. 4, Issue 4. – P. 760–764.
2. Iwaniec, J. Output-Only Identification of Vibratory Machine Suspension Parameters under Exploitation Conditions [Text] / J. Iwaniec, M. Iwaniec // *Solid State Phenomena*. – 2016. – Vol. 248. – P. 175–185. doi: 10.4028/www.scientific.net/ssp.248.175
3. Kharchenko, L. Fluctuation of multi-section aboveground pipeline region under the influence of moving diagnostic piston [Text] / L. Kharchenko, Ye. Kharchenko // *Vibration in Physical Systems*. – 2014. – Vol. 26. – P. 105–112.
4. Tykhan, M. New type of piezoresistive pressure sensors for environments with rapidly changing temperature [Text] / M. Tykhan, O. Ivakhiv, V. Teslyuk // *Metrology and Measurement Systems*. – 2017. – Vol. 24, Issue 1. – P. 185–192. doi: 10.1515/mms-2017-0010
5. Du, W. Y. Resistive, capacitive, inductive, and magnetic sensor technologies [Text] / W. Y. Du. – Boca Raton: CRC Press, 2014. – 408 p. doi: 10.1201/b17685
6. Kazys, R. Ultrasonic technique for Vibration Measurements [Text] / R. Kazys, R. Sliteris, L. Mazeika // *15th World Conference on Non-Destructive Testing*. – Rome, 2000.
7. Li, T. Study on the non-contact FBG vibration sensor and its application [Text] / T. Li, Y. Tan, Z. Zhou, L. Cai, S. Liu, Z. He, K. Zheng // *Photonic Sensors*. – 2015. – Vol. 5, Issue 2. – P. 128–136. doi: 10.1007/s13320-015-0220-9
8. Rață, G. System for Monitoring and Analysis of Vibrations at Electric Motors [Text] / G. Rață, M. Rață // *Intern. Journal of Emerging Technology and Advanced Engineering*. – 2014. – Vol. XXI, Issue 3. – P. 97–104.

9. Milovančević, M. Embedded Systems for Vibration Monitoring [Text] / M. Milovančević, A. Veg, A. Makedonski, J. Stefanović Marinović // Facta Universitatis. Series: Mechanical Engineering. – 2014. – Vol. 12, Issue 2. – P. 171–181.
10. Rocha, S. M. S. Method to Measure Displacement and Velocity from Acceleration Signals [Text] / S. M. S. Rocha, J. F. S. Feiteira, P. S. N. Mendes, U. P. B. Da Silva, R. F. Pereira // Intern. Journal of Engineering Research and Applications. – 2016. – Vol. 6, Issue 6. – P. 52–59.
11. Sekiya, H. Technique for Determining Bridge Displacement Response Using MEMS Accelerometers [Text] / H. Sekiya, K. Kimura, C. Miki // Sensors. – 2016. – Vol. 16, Issue 2. – P. 257. doi: 10.3390/s16020257
12. Goyal, D. Development of non-contact structural health monitoring system for machine tools [Text] / D. Goyal, B. S. Pabla // Journal of Applied Research and Technology. – 2016. – Vol. 14, Issue 4. – P. 245–258. doi: 10.1016/j.jart.2016.06.003
13. Albarbar, A. Suitability of MEMS Accelerometers for Condition Monitoring: An experimental study [Text] / A. Albarbar, S. Mekid, A. Starr, R. Pietruszkiewicz // Sensors. – 2008. – Vol. 8, Issue 2. – P. 784–799. doi: 10.3390/s8020784
14. Hjort, A. Measuring mechanical vibrations using an Arduino as a slave I/O to an EPICS control system [Text] / A. Hjort, M. Holmberg. – Uppsala University, 2015. – 25 p.
15. Chaudary, S. B. Vibration Monitoring of Rotating Machines Using MEMS Accelerometer [Text] / S. B. Chaudary, M. Sengupta, K. Mukherjee // Intern. Journal of Scientific Engineering and Research. – 2014. – Vol. 2, Issue 9.
16. Karvinen, T. Make sensors projects touch-based devices Arduino and Raspberry Pi [Text] / T. Karvinen, K. Karvinen, B. Valtakari. – Moscow: OOO “I. D. Williams”, 2015. – 432 p.
17. Boreiko, O. Developing a controller for registering passenger flow of public transport for the «smart» city system [Text] / O. Boreiko, V. Teslyuk // Eastern-European Journal of Enterprise Technologies. – 2016. – Vol. 6, Issue 3 (84). – P. 40–46. doi: 10.15587/1729-4061.2016.84143
18. 3-Axis,  $\pm 2$  g/ $\pm 4$  g/ $\pm 8$  g/ $\pm 16$  g Digital Accelerometer. Data Sheet ADXL345 [Electronic resource]. – Available at: <http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf>
19. ADXL345 Quick Start Guide. AN-1077 [Electronic resource]. – Available at: <http://www.analog.com/media/en/technical-documentation/application-notes/AN-1077.pdf>
20. AN10658. Sending I2C-bus signals via long communications cables [Electronic resource]. – Available at: [http://www.nxp.com/documents/application\\_note/AN10658.pdf](http://www.nxp.com/documents/application_note/AN10658.pdf)
21. Prots'ko, I. Algorithm of efficient computation DSTI-IV using cyclic convolutions [Text] / I. Prots'ko, V. Teslyuk // WSEAS Transactions on Signal Processing. – 2014. – Vol. 10. – P. 278–288.
22. Blahut, R. E. Fast algorithms for signal processing [Text] / R. E. Blahut. – Cambridge University Press, 2010. doi: 10.1017/cbo9780511760921
23. Prots'ko, I. Analysis algebraic system of arguments for prime size DHT [Text] / I. Prots'ko // Modern Problems of Radio Engineering. Telecommunications and Computer Science – Proceedings of the 11th International Conference, TCSET'2012. – 2012. – P. 428.
24. Library DTF FFTW [Electronic resource]. – Available at: <http://www.fftw.org/>
25. Library wiringPi for Raspberry Pi [Electronic resource]. – Available at: <http://wiringpi.com/reference/i2c-library/>