Threshold-event-based Algorithm for Real-time Pump Condition Monitoring: Static and Dynamic Analysis

Nodira Alimova [1], Aziza Khaitova [2,*], Nozima Norova [3], Alisher Mukhammadiev [2], Dilrabo Sultanmuratova [3], and Makhpuza Aripova [4]

[1] Department of Mechatronics and Robotics, Tashkent State Technical University named after I.A. Karimov, Tashkent, Uzbekistan
[2] Department of Modern Information and Communication Technologies, International Islamic Academy of Uzbekistan, Tashkent, Uzbekistan
[3] Department of Economics and Management, Tashkent International University of Financial Management and Technologies, Tashkent, Uzbekistan
[4] Department of Radio Engineering Devices and Systems, Tashkent State Technical University named after I.A. Karimov, Tashkent, Uzbekistan
Email: alimova@mail.com (N.A.); xaitovaaziza45@gmail.com (A.K.); n.norova@tift.uz (N.N); alisher@mail.com (A.M.); d.sultanmuradova@tift.uz (D.S); maxpuza1963@gmail.com (M.A)
*Corresponding author

***Abstract*—This study presents a threshold-event-based algorithm for real-time condition monitoring of a centrifugal pump unit driven by a 75 kW electric motor. The proposed approach combines binary vibration-event detection, cumulative decision logic, and time-based filtering to distinguish short-duration disturbances from sustained abnormal operating conditions. Experimental investigations were conducted under different hydraulic loading regimes using an SW-420 vibration sensor connected to an Arduino Uno microcontroller. The SW-420 was used as a threshold-triggered event detector rather than as a calibrated vibration-measurement sensor. Accordingly, the algorithm evaluated the number, duration, recurrence, and persistence of vibration-trigger events within predefined observation intervals. The proposed decision logic successfully filtered isolated short-term events while identifying persistent vibration-event sequences that required warning or protective action. The estimated execution time of one processing cycle was approximately 5.3 ms, confirming the feasibility of real-time implementation on the Arduino Uno platform. The obtained results demonstrate that the proposed low-cost monitoring approach can support abnormal-condition indication, protective pump control, and maintenance decision-making in industrial pumping systems.**

***Keywords*—vibration-event detection, pump condition monitoring, SW-420 sensor, threshold logic, dynamic filtering, Arduino, protective control**

I. Introduction

Pumping units are essential components of industrial and hydraulic infrastructure, including water-supply, irrigation, wastewater-treatment, and energy systems. Their stable operation directly influences process continuity, energy efficiency, maintenance planning, and operational safety. Abnormal vibration-event patterns may indicate unstable hydraulic conditions, mechanical disturbances, improper installation, or developing equipment degradation. Therefore, continuous vibration-event monitoring is an important component of preventive maintenance and protective control systems.

Conventional industrial vibration monitoring commonly relies on calibrated accelerometers, vibrometers, and spectral-analysis equipment. These instruments provide quantitative measurements of vibration displacement, velocity, or acceleration. However, their cost, installation requirements, and signal-processing complexity may limit their use in low-cost embedded monitoring applications [1].

For applications in which the primary objective is to detect vibration presence, repeated threshold exceedance, and persistent abnormal behaviour, a binary vibration-event sensor may provide a practical alternative. The SW-420 module generates a digital switching signal when mechanical vibration exceeds an adjustable comparator threshold. Although it does not provide calibrated vibration amplitude, velocity, acceleration, or frequency-domain information, it can be used to detect vibration-trigger events and evaluate their temporal behaviour.

A major limitation of instantaneous threshold-based monitoring is its sensitivity to short-term impacts, start-up disturbances, and isolated switching events. A single threshold exceedance does not necessarily represent a

Manuscript received May 11, 2026; revised June 1, 2026; accepted August 5, 2026; published Novemeber 20, 2026.

hazardous operating condition. Therefore, a practical monitoring algorithm should evaluate not only the current binary sensor state but also the duration, recurrence, and persistence of vibration-trigger events.

This study proposes a threshold-event-based monitoring algorithm that combines binary event detection, cumulative counting, and time-based filtering. The algorithm distinguishes short-duration disturbances from sustained vibration-event sequences and generates a protective decision only when predefined persistence criteria are satisfied. The proposed method was implemented using an SW-420 vibration sensor, an Arduino Uno microcontroller, relay-based pump control, and a communication module for operator notification.

The main contribution of this study is the development of a computationally simple embedded algorithm that evaluates vibration events in both static and dynamic forms. Static analysis describes the relationship between the current binary input state and the corresponding pump-control output. Dynamic analysis evaluates event duration, persistence, response delay, and internal counter behaviour over time. This approach reduces false switching caused by isolated disturbances and supports real-time protective control of industrial pump units [2, 3].

II. Literature Review

The reliability and operational efficiency of pumping systems are critical factors in industrial, municipal, irrigation, wastewater treatment, and energy infrastructures. Since pump units frequently operate under continuous and variable loading conditions, their technical condition has a direct impact on system availability, maintenance costs, and operational safety. Consequently, vibration-based condition monitoring has become one of the most widely adopted approaches for assessing the health status of rotating machinery.

Early studies on pump diagnostics primarily focused on conventional vibration indicators such as Root Mean Square (RMS) values, peak amplitudes, vibration velocity, and frequency-domain characteristics. Rao [4] and Wowk [5] demonstrated that vibration measurements provide valuable information regarding the mechanical condition of bearings, shafts, couplings, and rotor systems. Similarly, Randall and Antoni [6] reported that vibration signatures contain diagnostic features capable of identifying incipient defects long before catastrophic failure occurs.

With the advancement of signal-processing techniques, researchers increasingly shifted their attention toward frequency-domain and time–frequency analysis methods. Antoni [7] proposed kurtogram-based analysis for detecting transient faults in rotating machinery, while Peng and Chu [8] demonstrated the effectiveness of wavelet-transform techniques for identifying non-stationary vibration patterns. Lei *et al.* [9] further showed that Empirical Mode Decomposition (EMD) can improve fault-feature extraction in complex rotating systems operating under variable conditions.

Several investigations have specifically addressed vibration monitoring of centrifugal pumps. Gülich [10] established the relationship between hydraulic operating conditions and vibration behaviour in centrifugal pumping systems. Chen *et al.* [11] developed dynamic models of pump rotor systems and demonstrated that changes in rotational speed significantly affect vibration characteristics and spectral peak distribution. Experimental studies by Mustafin *et al.* [12] and Panfilov [13] confirmed that abnormal vibration-event states are frequently associated with hydraulic instability, cavitation phenomena, bearing degradation, and rotor imbalance.

In recent years, the rapid development of digital technologies has stimulated the integration of artificial intelligence and machine-learning methods into condition-monitoring systems. Li *et al.* [14] proposed a multimodal deep-support-vector classification approach for fault diagnosis of rotating machinery, achieving improved classification accuracy compared with conventional diagnostic methods. Tama *et al.* [15] and Sunal *et al.* [16] demonstrated that convolutional neural networks and transfer-learning algorithms can effectively detect pump faults from vibration signals under varying operating conditions. Furthermore, Zhi *et al.* [17] reported that multi-channel information fusion significantly enhances diagnostic reliability in centrifugal pump monitoring applications.

The emergence of Industrial Internet of Things (IIoT) technologies has further expanded the capabilities of vibration-based monitoring systems. Antoni [7] emphasized the role of intelligent sensor networks in water-management infrastructure, while recent studies [18, 19] highlighted the potential of integrating vibration sensors, motor-current analysis, and cloud-based data processing for continuous equipment surveillance. These developments have enabled the transition from periodic inspection strategies toward predictive and condition-based maintenance frameworks.

Despite substantial progress in vibration-based diagnostics, many advanced approaches require calibrated sensors, high-resolution data-acquisition systems, extensive signal-processing resources, or large training datasets. Such requirements may limit their implementation in low-cost embedded protection systems. In applications where the main objective is to detect repeated or persistent vibration events rather than to quantify absolute vibration severity, binary threshold sensors provide a simpler alternative. However, instantaneous binary detection is susceptible to false triggering caused by impacts, start-up transients, and short-duration disturbances. The present study addresses this limitation by introducing cumulative counters and time-based decision logic for distinguishing isolated vibration events from sustained abnormal event sequences.

Unlike calibrated accelerometer-based systems, the present study does not estimate vibration amplitude, velocity, acceleration, or spectral fault frequencies. The SW-420 module is used only to detect threshold-triggered binary vibration events and to evaluate their temporal persistence.

III. Materials and Methodology

A. Sensor Selection and Measurement Principle

Modern vibration monitoring systems employ various sensing technologies, including resistive, inductive, capacitive, piezoelectric, and microelectromechanical systems based sensors. The selection of a sensor depends on the monitoring objective, required measurement accuracy, cost, and computational complexity. For industrial condition monitoring, calibrated accelerometers are generally used when quantitative vibration parameters such as acceleration, velocity, or displacement are required. However, in low-cost embedded monitoring systems intended for event detection and protective control, threshold-based sensors provide a practical alternative because of their simplicity, low power consumption, and ease of integration. In addition, Nigmatov *et al.* [20] proposed an intelligent system for the effective protection of pump units, demonstrating the practical importance of automated monitoring and protective control in enhancing the operational reliability of pumping equipment.

In this study, vibration events were detected using the SW-420 vibration sensor (Fig. 1). Unlike industrial vibrometers, the SW-420 is a comparator-based vibration switch that generates a digital output whenever the vibration intensity exceeds an adjustable threshold. Therefore, the sensor was employed exclusively for binary vibration-event detection rather than quantitative vibration measurement.

The sensor was connected to an Arduino Uno microcontroller through its digital output pin. During operation, the Arduino continuously sampled the binary output state of the SW-420. The digital output was converted in software into a binary event variable, where $v(k) = 1$ denoted a detected threshold-trigger event and $v(k) = 0$ denoted the absence of such an event. The proposed monitoring algorithm processed only this binary event information to evaluate the duration, recurrence, and persistence of vibration-trigger events.

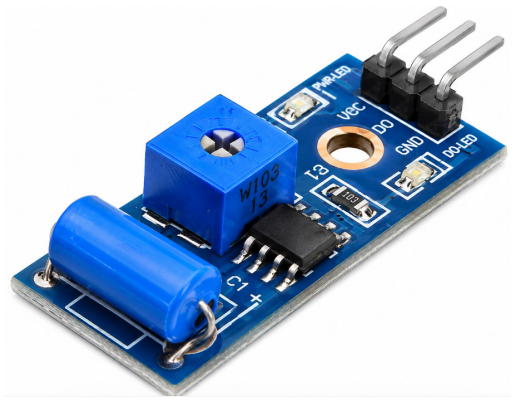


Fig. 1. External appearance of the SW-420 vibration sensor module.

Since the SW-420 does not provide calibrated measurements of vibration-trigger frequency, displacement, or acceleration, no quantitative vibration values expressed in mm/s or g were used in the proposed decision-making algorithm. Instead, abnormal operating conditions were identified using cumulative event counting and time-based persistence analysis. This approach considerably reduces hardware cost while maintaining reliable abnormal-condition detection for real-time embedded pump monitoring.

The SW-420 sensor module was rigidly mounted on the external casing of the centrifugal pump to ensure reliable transmission of mechanical vibration to the sensing element (Fig. 2). Experimental observations were carried out over a one-year period under actual industrial operating conditions. During this period, vibration-event data were collected during normal operation as well as under abnormal operating conditions identified at the enterprise. The sensor sensitivity was adjusted using the onboard potentiometer, and the threshold values used in the proposed algorithm were selected empirically after analysing the accumulated vibration-event sequences. The selected thresholds ($k_{min} = 400$, $x_{crit} = 800$, and $i_{max} = 2000$ controller cycles) provided reliable discrimination between isolated short-duration disturbances and persistent abnormal vibration-event patterns. Throughout all experiments, the sensor mounting position and sensitivity settings remained unchanged to ensure the consistency and comparability of the obtained results.

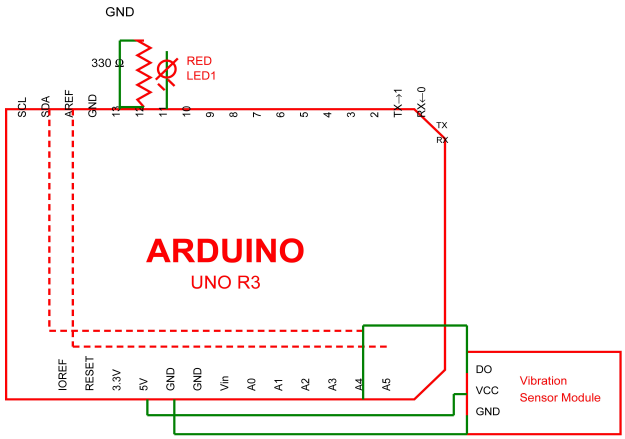


Fig. 2. Connection of the SW-420 sensor module to the Arduino Uno microcontroller.

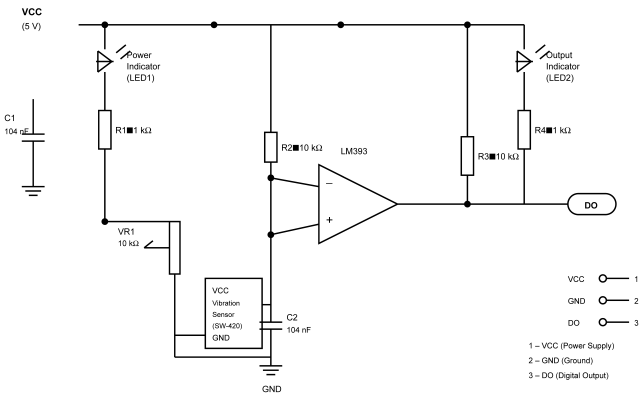


Fig. 3. Comparator-based circuit configuration of the SW-420 vibration sensor module.

Owing to its compact structure, operational stability, low cost, and ease of implementation, the SW-420 sensor is well suited for embedded condition-monitoring applications where the primary objective is the reliable

detection of persistent abnormal vibration events rather than precise vibration metrology [21].

The circuit shown in Fig. 3 illustrates the SW-420 sensing element, the LM393 comparator, the adjustable sensitivity potentiometer, and the digital output terminal. Mechanical vibration changes the state of the sensing element, while the comparator converts this response into a binary output signal. The switching threshold is adjusted by the onboard potentiometer. Thus, the module provides threshold-event information rather than calibrated vibration amplitude, velocity, or acceleration values [22, 23].

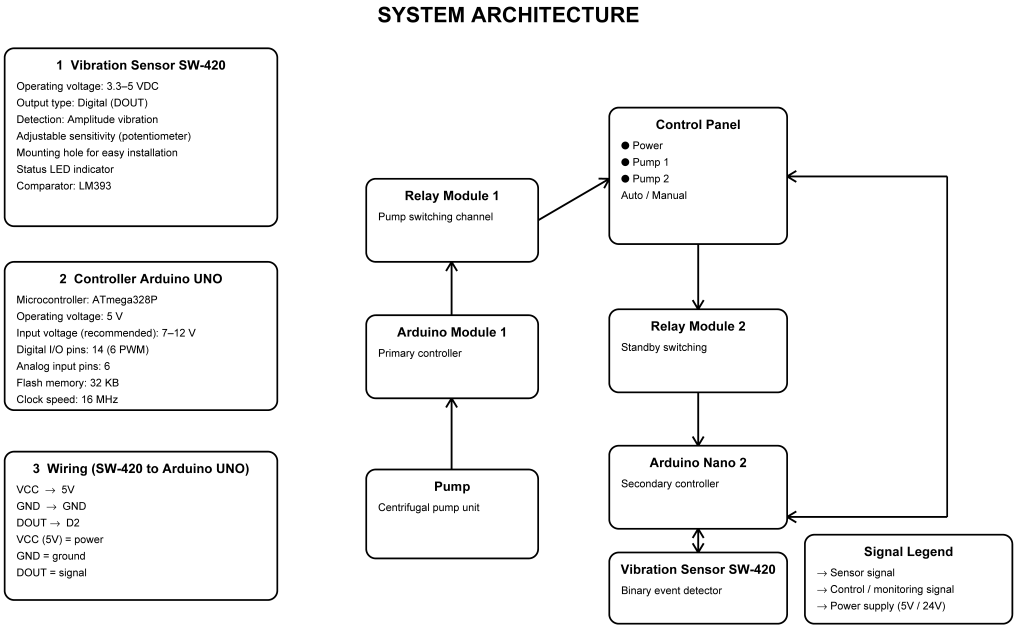


Fig. 4. Functional architecture of the proposed pump-monitoring and protective-control system.

The functional architecture shown in Fig. 4 consists of an SW-420 vibration sensor, an Arduino Uno microcontroller, relay modules, two pump-control channels, and an operator interface. The sensor produces a binary vibration-event signal, which is acquired and processed by the microcontroller. The controller evaluates event duration and persistence and subsequently issues a warning, maintains normal operation, switches to the standby pump, or stops both pumps depending on the decision state. The architecture is intended for low-cost protective monitoring and does not perform calibrated vibration metrology [18, 24, 25].

The proposed system architecture provides continuous monitoring of threshold-triggered vibration events, supports the assessment of abnormal operating conditions, and contributes to the prevention of emergency failures. Furthermore, its modular structure may support future Condition-Based Maintenance (CBM) applications when combined with calibrated vibration sensors and historical operating data. By enabling continuous condition assessment and early abnormal-condition indication, the system improves operational reliability, reduces unplanned downtime, and enhances maintenance efficiency [16, 26].

Fig. 5 presents the vibration detection algorithm implemented in the proposed monitoring device designed for pump units used in the transportation of treated wastewater. The algorithm is intended to continuously evaluate the vibration condition of the pump and to identify abnormal operating states in real time.

The operating sequence of the algorithm is as follows. The digital output of the SW-420 was read by the Arduino Uno as a binary state. The digital output was converted in software into a binary event variable, where $v(k) = 1$ denoted a detected threshold-trigger event and $v(k) = 0$ denoted the absence of such an event.

At each controller cycle, the binary output state of the SW-420 module is sampled. If the sensor output indicates a threshold-trigger event, the corresponding counter variable is incremented. Repeated evaluation of threshold exceedances over the monitoring period enables the algorithm to distinguish between transient fluctuations and persistent abnormal vibration conditions. This approach improves the reliability of fault detection while minimizing the occurrence of false alarms caused by short-term disturbances [27, 28]

By incorporating continuous vibration-event assessment and cumulative decision logic, the proposed algorithm provides an effective mechanism for abnormal-condition indication and protective control of pump units operating in wastewater-treatment systems. Its implementation contributes to enhanced operational reliability, timely maintenance intervention, and the prevention of unexpected equipment failures.

Here $v(k)$ is the binary vibration-event variable at discrete time step k; i_1 and i_2 are the total numbers of sampling cycles for pumps N_1 and N_2, respectively; x_1 and x_2 are the numbers of threshold-trigger events detected for pumps N_1 and N_2; and t_s is the sampling interval between two consecutive controller cycles.

Step 1. The initial configuration parameters are established in the microcontroller.

Step 2. The operation of the GSM modules is verified, and a connection to the communication network is established.

Step 3. The pump designated for operation is selected and activated.

Step 4. The binary threshold-event output of the SW-420 sensor installed on the operating pump is monitored.

Step 5. If a vibration-trigger event is detected, the event counter is incremented and the monitoring procedure continues during the predefined observation interval.

Step 6. The persistence of consecutive vibration-trigger events is evaluated using the predefined decision thresholds.

Step 7. If the persistence criterion has not yet been satisfied, monitoring continues without changing the operating state of the pump.

Step 8. If the persistence counter exceeds the predefined protection threshold, the vibration condition is classified as abnormal and the corresponding pump is switched off.

Step 9. The subsequent pump is activated, and its vibration-event state is monitored during the predefined observation interval.

Step 10. If the second pump also produces a persistent sequence of threshold-trigger events exceeding the predefined protection criterion, both pumps are shut down and a fault notification is transmitted to the duty operator.

Step 11. If no persistent abnormal vibration condition is confirmed, the controller continues normal monitoring and verifies incoming operator commands.

Step 12. A logical “0” command deactivates the pumps.

Step 13. A logical “1” operator command requests the current operating status of the pump system.

If the cumulative event counter x exceeds 800 within the predefined observation interval, the vibration-event sequence is classified as persistent and abnormal. In this case, the operating pump is switched off and the standby pump is activated [9].

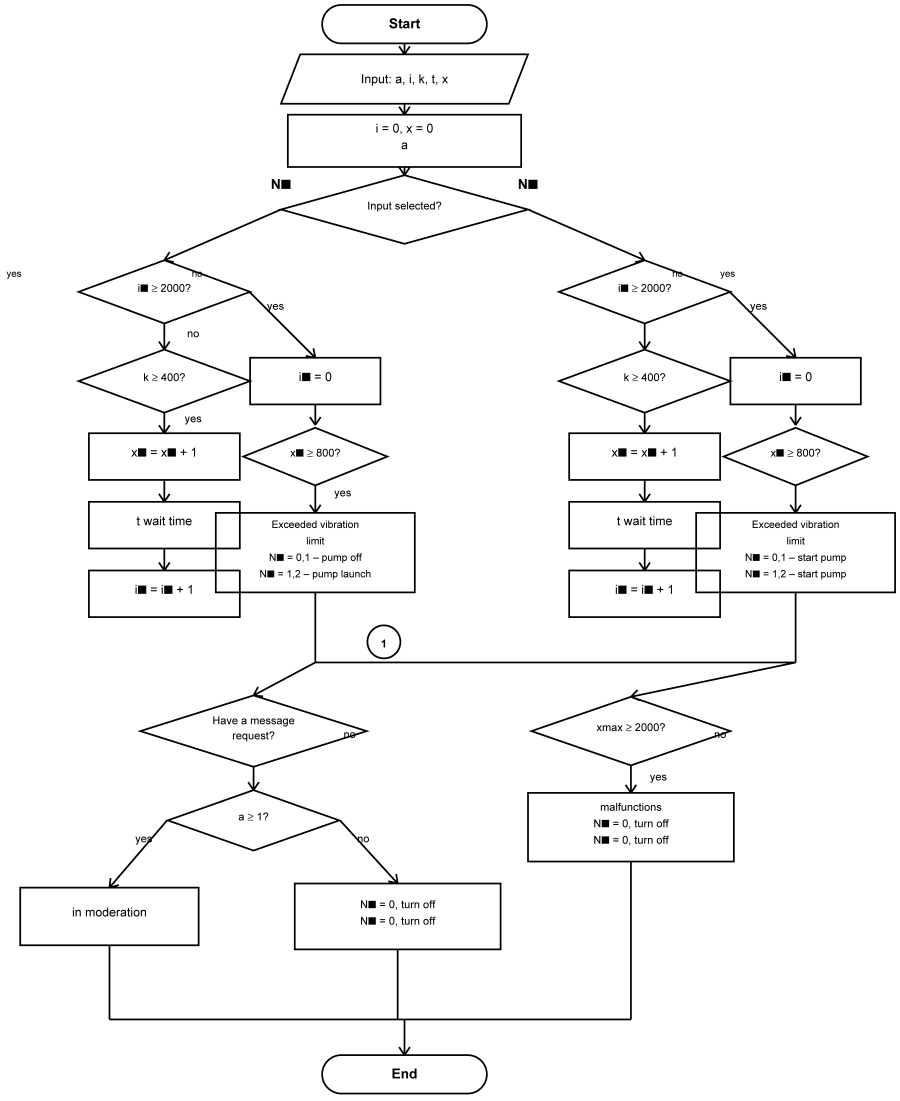


Fig. 5. Flowchart of the proposed threshold-event monitoring algorithm implemented on the Arduino platform.

Fig. 5 illustrates the operational sequence of the proposed threshold-event monitoring algorithm implemented on the Arduino Uno platform. The flowchart summarizes the logic executed by the controller, including event acquisition, counter updating, threshold comparison, pump switching, and operator notification. The diagram presents the implementation sequence of the embedded software rather than a mathematical representation of the algorithm.

The parameter i_{max} = 2000 defines the total observation length, k_{min} = 400 defines the minimum number of trigger events required to pass the preliminary filtering stage, and x_{crit} =800 defines the persistent-event criterion used for protective switching.

TABLE I. VARIABLES AND PARAMETERS USED IN THE ALGORITHM

No.	Symbol	Unit	Description and Physical Meaning
1	$\triangleleft t$	S	Duration of a single controller cycle (scan time)
2	n	-	Discrete time-step index
3	t_n	S	Real time, $(t_n = n/\triangleleft t)$
4	v_n	-	Binary vibration signal: 1 – vibration present, 0 – vibration absent
5	P_n	-	Pump operating state: 1 – operating, 0 – switched off
6	k_n	controller cycles	Counter representing the continuous duration of vibration
7	x_n	controller cycles	Vibration stability counter
8	K_1	controller cycles	Threshold level used for short-term filtering
9	K_2	controller cycles	Long-term limit corresponding to an emergency condition
10	X_2	controller cycles	Permissible threshold for sustained vibration conditions

B. Mathematical Model of the Threshold-Event Algorithm

Discrete time: $t_n = n \triangleleft t, n = 0,1,2, \ldots$ **Input signal:** $v_n \in \{0,1\}$, where $v_n = 1$ indicates the presence of vibration, and $v_n = 0$ indicates the absence of vibration.

Output signal: $P_n \in \{0,1\}$, where $P_n = 1$ denotes that the pump is operating, and $P_n = 0$ indicates that the pump has been switched off.

Internal counters: k_n —confirmation/waiting-time counter; x_n—vibration stability counter.

Threshold values: $K_1 = 400, K_2 = 2000, X_2 = 2000$ If an additional counter, i is also introduced (for example, $i < 2000$), it is generally used to limit the overall operating cycle or to initiate a system restart. However, the fundamental mathematical model of the vibration-based monitoring algorithm can be fully described using only the variables k and x. The variables and parameters used in the proposed algorithm are summarized in Table I.

If vibration persists continuously, the value of k_n increases incrementally at each discrete time step. Conversely, when vibration is no longer detected, the counter k is reset to zero.

$$k_{n+1} = \begin{cases} min(k_n + 1.K_2), & v_n = 1 \\ 0, & v_n = 0 \end{cases} \quad (1)$$

where: $k_n \geq K_1$, if the vibration signal passes the short-term filtering stage; $k_n \geq K_2$, if the vibration does not persist for a sufficiently long duration to satisfy the filtering criterion. The rules for vibration signal formation and the corresponding counter dynamics are summarized in Table II.

TABLE II. RULES FOR VIBRATION SIGNAL FORMATION

Assessment Criterion	No.	State	Update Rule	Physical Interpretation
Vibration Signal Formation	1	$A_n \geq A_{max}$	$v_n = 1$	Vibration is classified as an abnormal or emergency condition.
	2	$A_n \geq A_{max}$	$v_n = 0$	Vibration remains within the permissible operating range.
Dynamics of the Vibration Duration Counter (k_n)	1	$v_n = 1$	$k_{n+1} = k_n + 1$	Vibration continues uninterrupted.
	2	$v_n = 0$	$k_{n+1} = 0$	Vibration ceases, and the counting process is restarted.

The vibration stability counter x begins to increase only after the vibration duration counter K_1 has exceeded the predefined threshold K_1. This condition ensures that only sustained vibration events are considered in the stability assessment, while short-term fluctuations and transient disturbances are filtered out.

Accordingly, the counter dynamics can be expressed as:

$$x_{n+1} = \begin{cases} min(x_n + 1, X_2), & (v_n = 0 \wedge (k_n \geq K_1) \\ 0, & v_n = 0 \end{cases} \quad (2)$$

TABLE III. VIBRATION STABILITY ASSESSMENT COUNTER x_n

No.	State	Update Rule	Physical Interpretation
1	$v_n = 1 \; and \; k_n \geq K_1$	$x_{n+1} = x_n + 1$	The vibration condition is considered stable and persistent.
2	$v_n = 0$	$x_n + 1 = 0$	Stability is not achieved; the vibration condition is regarded as unstable.

This approach enables the monitoring system to distinguish persistent vibration conditions from temporary oscillations, thereby improving the reliability of the diagnostic process and reducing the likelihood of false alarm generation. The vibration stability assessment rules are summarized in Table III.

The decision to stop the pump unit can be determined according to two independent criteria:

$$P_{n+1} = \begin{cases} 0, & (k_n \geq K_2) \vee (x_n \geq X_2) \\ P_n, & \text{otherwise} \end{cases} \quad (3)$$

Initial conditions:

$$K_0 = 1, k_0 = 0, x_0 = 0$$

If the duration of a single controller cycle is equal to $\lhd t$ seconds, then: The decision logic for automatic shutdown of the pump unit based on vibration monitoring is summarized in Table IV. The initial conditions of the monitoring algorithm are presented in Table V, while the relationship between the discrete controller parameters and their equivalent real-time values is summarized in Table VI.

$T_{K_1} = K_1 \lhd t$, For example, if: $T_{K_2} = K_2 \lhd t$, and $T_{X_{2_1}} = X_2 \lhd t$, then: $\lhd t = ms = 0.001\ s$, $T_{K_1} = 0.4\ s, T_{K_1} = 2.0\ s$.

The proposed decision-making mechanism enables the monitoring system to identify sustained abnormal vibration conditions and initiate a protective shutdown of the pump unit when the predefined operational limits are exceeded. Such an approach improves equipment safety and contributes to the prevention of vibration-related failures.

TABLE IV. DECISION LOGIC FOR AUTOMATIC SHUTDOWN OF THE PUMP UNIT BASED ON VIBRATION MONITORING

No.	Condition	Output Action	Description
1	$k_n \geq K_1$	$P_{n+1} = 1$	The vibration has persisted for an excessively long period.
2	$x_n \geq X_2$	$P_{n+1} = 0$	A sustained and stable vibration condition is identified as a fault criterion.
3	Otherwise	$P_{n+1} = P_n$	The pump continues normal operation.

TABLE V. INITIAL CONDITIONS OF THE MONITORING ALGORITHM

No.	Variable	Value	Description
1	P_0	1	The pump is activated and operates in the normal working state.
2	k_0	0	Initial value of the vibration duration counter.
3	x_0	0	Initial value of the vibration stability counter.

TABLE VI. RELATIONSHIP BETWEEN DISCRETE PARAMETERS AND REAL-TIME VALUES

No.	Parameter	In Discrete Controller cycles	Equivalent Real-Time Expression
1	Short-term filtering threshold	K_1	$T_1 = K_1 \lhd t$
2	Long-duration threshold	K_2	$T_2 = K_2 \lhd t$

IV. RESULTS AND DISCUSSION

A. Experimental Evaluation of the Threshold-Event Monitoring Algorithm

The proposed threshold-event monitoring algorithm was evaluated under different operating conditions of the centrifugal pump unit. During each observation interval, the Arduino Uno continuously sampled the digital output of the SW-420 sensor. The acquired binary signal was processed to determine the occurrence, duration, recurrence, and persistence of threshold-triggered vibration events.

The experimental evaluation was focused on verifying the ability of the proposed decision logic to distinguish isolated short-duration disturbances from sustained abnormal vibration-event sequences. Since the SW-420 module provides only a digital switching output, the experiments were not intended to determine calibrated vibration amplitude, velocity, acceleration, or frequency-domain characteristics.

During normal pump operation, isolated switching events occasionally occurred because of short-term mechanical impacts, start-up disturbances, and temporary hydraulic fluctuations. These events caused temporary increments of the internal counter; however, the counter returned to its initial state when the event sequence was interrupted. Consequently, short-duration disturbances did not affect the operating state of the pump.

When threshold-trigger events occurred repeatedly over a predefined observation interval, the cumulative counters increased progressively. The algorithm first applied a preliminary filtering condition to reject isolated events. If the event sequence continued and the persistence threshold was exceeded, the operating condition was classified as abnormal.

The total observation interval was defined as i_{max} = 2000 controller cycles. The preliminary filtering threshold was defined as k_{min} = 400, while the persistent-event protection threshold was defined as x_{crit} = 800. These threshold values were used to distinguish temporary disturbances from sustained abnormal vibration-event conditions.

When the cumulative event counter remained below the protection threshold, the pump continued normal operation. When the counter exceeded x_{crit}, the operating pump was switched off and the standby pump was activated. If the standby pump also generated a persistent threshold-event sequence, both pumps were stopped and a fault notification was transmitted to the duty operator.

The experimental observations demonstrated that the proposed algorithm performs three main functions: short-duration event filtering, persistent-event confirmation, and protective pump control. This decision structure reduces false switching caused by isolated disturbances and supports timely protective actions during sustained abnormal operating conditions.

The proposed method should therefore be interpreted as a low-cost abnormal-condition indication and protective-control approach. It does not provide calibrated vibration severity, fault-frequency identification, or detailed mechanical fault classification. Such functions

would require calibrated accelerometers and additional signal-processing equipment [8, 29].

As shown in Table VII, the proposed algorithm does not respond to each isolated sensor trigger as a fault condition. Instead, the control output depends on the temporal persistence of the detected event sequence. This approach improves the robustness of the monitoring system and reduces the probability of false protective actions.

TABLE VII. EXPERIMENTAL DECISION STATES OF THE THRESHOLD-EVENT MONITORING ALGORITHM

No.	Operating Condition	Sensor-Event Behavior	Counter Response	Controller Decision
1	Normal operation	No persistent threshold-trigger sequence	Counters remain below the filtering threshold	Pump continues operation
2	Isolated disturbance	One or several short-duration trigger events	Counter temporarily increases and then resets	Disturbance is filtered
3	Repeated disturbance	Trigger events recur but remain below the protection threshold	Persistence is evaluated	Monitoring continues
4	Sustained abnormal condition	Persistent threshold-trigger sequence	$x \geq x_{crit}$	Operating pump is switched off
5	Abnormal condition in both pumps	Persistent events are detected in both pump channels	Protection criterion is confirmed for both pumps	Both pumps are stopped and a notification is sent

B. Static Characteristics of the Algorithm

The static characteristics of the proposed algorithm are defined by a set of decision rules expressed as functions of the input signal $v(k)$, the output state $P(k)$, and the specified threshold parameters, without explicitly considering the time factor. Under these conditions, the algorithm is represented by its current state, which reflects the relationship between the input conditions and the corresponding control actions at a particular instant.

In the static analysis framework, the decision-making process is determined solely by the current values of the monitored variables and the predefined threshold levels. Consequently, the algorithm can be described as a discrete-state logical model in which each combination of input parameters corresponds to a specific operational state and a corresponding output response. This representation provides a convenient basis for evaluating the algorithm's decision logic and operational behaviour under different vibration conditions. The static decision characteristics of the vibration monitoring algorithm are summarized in Table VIII.

TABLE VIII. STATIC DECISION CHARACTERISTICS OF THE VIBRATION MONITORING ALGORITHM

No.	System Input State	Additional Condition	Output State	Description
1	$v = 0$	-	$P = 1$	No vibration is detected; the pump operates normally.
2	$v = 1$	$k < K_1$	$P = 1$	Short-term vibration is ignored by the control system.
3	$v = 1$	$K_1 \leq k \leq K_2$	$P = 1$	Vibration is present but does not represent an emergency condition.
4	$v = 1$	$k \geq K_2$	$P = 0$	Persistent vibration condition is detected.
5	$v = 1$	$x \geq X_2$	$P = 0$	Sustained abnormal vibration is classified as an emergency condition.

C. Dynamic Characteristics of the Algorithm

The dynamic characteristics describe the response of the algorithm to variations in the vibration signal over time. In this process, the counters k_n and x_n play a fundamental role in evaluating the duration and stability of the detected vibration events. Unlike the static representation, the dynamic behavior of the algorithm takes into account the temporal evolution of the input signal and the corresponding changes in the internal states of the monitoring system.

The time-dependent response is determined by the accumulation and updating of the counter values during successive measurement cycles. As vibration persists, the counters progressively increase, enabling the algorithm to distinguish transient disturbances from sustained abnormal operating conditions. Consequently, the decision-making process is not based solely on the instantaneous value of the vibration signal but also on its persistence and stability over a specified observation period.

This dynamic approach improves the reliability of vibration monitoring by reducing the influence of short-term fluctuations and measurement noise, while ensuring timely detection of prolonged vibration-event conditions that may indicate abnormal operation requiring protective intervention. As a result, the algorithm provides a robust framework for real-time condition monitoring and protective control of pump units operating under variable dynamic loads. The dynamic characteristics of the proposed algorithm are summarized in Table IX.

TABLE IX. DYNAMIC CHARACTERISTICS OF THE ALGORITHM

No.	Parameter	Expression	Physical Interpretation
1	Response time	$T_1 = K_1 \lhd t$	Short-term filtering of vibration signals
2	Fault-induced delay	$T_2 = K_2 \lhd t$	Time interval before pump shutdown
3	Stability confirmation time	$T_x = X_2 \lhd t$	Confirmation period for a persistent fault condition
4	Integration effect	k_n	Aggregation of vibration information over time
5	Memory effect	x_n	Consideration of previous system states in the decision-making process

The algorithm execution schedule and the corresponding processing time intervals are presented in Table X.

Based on the processing intervals presented in Table X, the estimated duration of one complete execution cycle of the algorithm was: Tcycle = 5.3 ms Accordingly, the total execution time for 2000 consecutive cycles can be calculated as: T_{total} = 5.3 × 2000 = 10,600 ms ≈ 10.6 s.

TABLE X. ALGORITHM EXECUTION SCHEDULE IN MILLISECOND TIME INTERVALS

No.	Processing Step	Description	Time Interval (ms)	Duration (ms)
1	Start	Algorithm execution is initiated	0.0–0.1	0.1
2	Initialization	$i = 0, k = 0, x = 0$	0.1–0.4	0.3
3	Input acquisition	Presence of vibration is evaluated	0.4–0.7	0.3
4	Filtering stage (K_i)	Vibration signal is filtered	0.7–1.5	0.8
5	State verification	The value of $i \geq 2000$ is evaluated	1.5–1.7	0.2
6	Internal loop	If $k < 400$ is, $x = x + 1$	1.7–2.5	0.8
7	Clock cycle	One control cycle is completed	2.5–3.5	1.0
8	Adjustment stage	The value of $x \geq 3000$ is evaluated	3.5–4.1	0.6
9	Message verification	Signal reception is confirmed	4.1–4.4	0.3
10	Decision-making	$x \geq 4$ Decision on	4.4–4.6	0.2
11	Execution result	Values of N_1, N_2 are updated	4.6–5.2	0.6
12	Completion	Algorithm execution cycle is terminated	5.2–5.3	0.1

The estimated timing results indicate that one processing cycle can be completed in approximately 5.3 ms. This execution time is sufficiently short to support real-time event processing in pump-monitoring applications.

Furthermore, all conditional transitions, state evaluations, filtering procedures, and computational operations are executed within predefined time intervals, ensuring deterministic behaviour and stable algorithm performance. The relatively short processing time confirms that the proposed algorithm is capable of responding promptly to changes in vibration conditions and can therefore be effectively implemented in real-time monitoring and event-based detection for pump units. The achieved temporal performance also demonstrates the suitability of the algorithm for continuous condition monitoring and protective control.

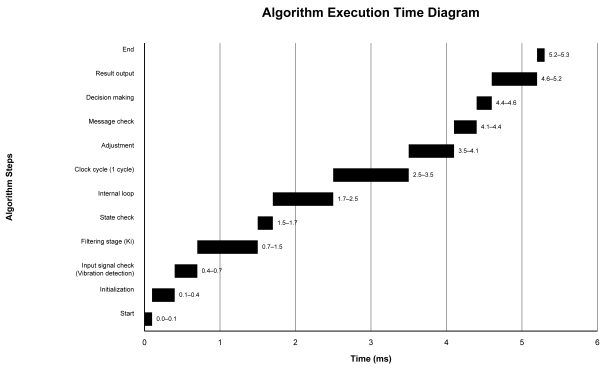


Fig. 6. Estimated execution schedule of one algorithm cycle.

Fig. 6 shows diagram illustrates the sequential execution of the algorithmic stages together with the corresponding time intervals required for each processing step. The horizontal axis (X-axis**)** represents time expressed in milliseconds (ms), whereas the vertical axis (Y-axis) denotes the individual stages of the algorithm.
Each horizontal bar indicates the start and end times associated with a particular processing operation.

As shown in the diagram, all algorithmic stages are executed sequentially, and the total execution time of a complete processing cycle is 5.3 ms. The largest portion of the execution time is associated with the clock-cycle stage and the signal-processing stage, indicating that these operations constitute the principal computational workload of the vibration-monitoring procedure.

The obtained timing characteristics confirm that the proposed algorithm is capable of performing vibration-signal processing within the constraints of real-time operation. The relatively short execution time ensures prompt system response and supports the implementation of continuous event monitoring and protective-control functions in pump-unit systems.

The practical applicability of monitoring and diagnostic methods for technological units in industrial wastewater treatment systems has previously been demonstrated [29].

V. CONCLUSIONS

This study developed a threshold-event-based monitoring algorithm for real-time protective control of a centrifugal pump unit driven by a 75 kW electric motor. The SW-420 module was used as a binary vibration-event detector rather than as a calibrated vibration-measurement device. The proposed algorithm evaluated the duration, recurrence, and persistence of threshold-trigger events through cumulative counters and time-based decision rules.

The static analysis demonstrated the relationship between the binary sensor state, internal threshold conditions, and the corresponding pump-control output. The dynamic analysis showed that short-duration trigger events could be filtered, whereas sustained event sequences activated warning, pump-switching, or shutdown actions.

The estimated execution time of one processing cycle was approximately 5.3 ms. This result indicates that the algorithm is computationally suitable for implementation on an Arduino Uno platform. The proposed approach provides a low-cost solution for abnormal-condition indication and protective pump control. However, it does

not replace calibrated vibration measurement or frequency-domain fault diagnosis. Future work will focus on experimental calibration, adaptive threshold selection, and integration with calibrated accelerometers and additional operating sensors.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

NA contributed to the conceptualization of the study and supervised the overall research process. AK developed the research methodology, organized the experimental investigation, and coordinated manuscript preparation. NN contributed to data analysis, interpretation of the results, and manuscript revision. AM contributed to software implementation, algorithm development, and technical analysis. DS contributed to validation, literature analysis, and critical revision of the manuscript. MA contributed to data curation, visualization, and final editing of the manuscript. All authors have read and approved the final version of the manuscript.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to Tashkent State Technical University and the International Islamic Academy of Uzbekistan for their academic and organizational support during the preparation of this study.

REFERENCES

[1] F. Bekjanov, I. Khudayev, and I. Turdibekov, "Detection of Errors in the Vibration Diagnostics of Pump Units," *International Journal of Advanced Research in Science, Engineering and Technology*, vol. 6, no. 9, pp. 10662–10665, 2019.

[2] O. Y. Glovatsky, R. R. Ergashev, and F. A. Bekchanov, "Results of diagnostics of pump units," *Irrigation and Land Reclamation*, vol. 1, no. 11, pp. 36–39, 2018.

[3] C. E. Sunal, V. Velisavljevic, V. Dyo, B. Newton, and J. Newton, "Centrifugal pump fault detection with convolutional neural network transfer learning," *Sensors*, vol. 24, no. 8, 2442, 2024. doi: 10.3390/s24082442

[4] S. S. Rao, *Mechanical Vibrations*, 5th ed. Mexico: Pearson Education, 2012.

[5] V. Wowk, *Machinery Vibration: Measurement and Analysis*, New York, NY, USA: McGraw-Hill, 1991.

[6] R. B. Randall and J. Antoni, "Rolling element bearing diagnostics—A tutorial," *Mechanical Systems and Signal Processing*, vol. 25, no. 2, pp. 485–520, 2011.

[7] J. Antoni, "Fast computation of the kurtogram for detection of transient faults," *Mechanical Systems and Signal Processing*, vol. 21, pp. 108–124, 2007.

[8] Z. K. Peng and F. L. Chu, "Application of the wavelet transform in machine condition monitoring and fault diagnostics," *Mechanical Systems and Signal Processing*, vol. 18, pp. 199–221, 2004.

[9] Y. Lei, J. Lin, Z. He, and M. J. Zuo, "A review on empirical mode decomposition in fault diagnosis of rotating machinery," *Mechanical Systems and Signal Processing*, vol. 35, pp. 108–126, 2013.

[10] J. F. Gülich, *Centrifugal Pumps*, 3rd ed. Berlin, Germany: Springer, 2014.

[11] F. Bekchanov, R. Ergashev, T. Mavlanov, and O. Glovatskiy, "Mathematical Model of Vibrating Air Pump Unit," *E3S Web of Conferences*, vol. 97, 05045, 2019

[12] A. K. Mustafin, D. N. Kabylkaiyr, T. A. Iskakova, and V. V. Grebenkin, "Development of Methods for Protecting Pump Units from Vibration Loads," *Bulletin of Pavlodar State University, Energy Series*, no. 1, pp. 296–300, 2019. (in Russian)

[13] R. Ergashev, F. Bekchanov, A. Atajanov, I. Khudaev, and F. Yusupov, "Vibrodiagnostic Method of Water Pump Control," *E3S Web of Conferences*, vol. 264, 04026, 2021.

[14] C. Li, R. V. Sánchez, G. Zurita, M. Cerrada, D. Cabrera, and R. E. Vásquez, "Multimodal deep support vector classification with homologous features and its application to gearbox fault diagnosis," *Neurocomputing*, vol. 168, pp. 119–127, 2015.

[15] B. A. Tama, M. Vania, S. Lee, and S. Lim, "Recent advances in the application of deep learning for fault diagnosis of rotating machinery using vibration signals," *Artificial Intelligence Review*, vol. 56, no. 5, pp. 4667–4709, 2023. doi: 10.1007/s10462-022-10293-3

[16] C. E. Sunal, V. Dyo, and V. Velisavljevic, "Review of machine learning based fault detection for centrifugal pump induction motors," *IEEE Access*, vol. 10, pp. 71344–71355, 2022. doi: 10.1109/ACCESS.2022.3187718

[17] Y. Zhi, Q. Huang, F. Tao, H. Li, D. Hao, H. Qin, and Q. Fu, "Research on fault diagnosis of vertical centrifugal pump based on multi-channel information fusion," *Processes*, vol. 13, no. 4, 1152, 2025. doi: 10.3390/pr13041152

[18] M. Song, Y. Liu, X. Wang, and H. Zhang, "Centrifugal pump cavitation fault diagnosis based on feature-level multi-source information fusion," *Processes*, vol. 12, 196, 2024.

[19] N. Ullah, Z. Ahmad, M. F. Siddique, K. Im, D. K. Shon, T. H. Yoon, D. S. Yoo, and J. M. Kim, "An intelligent framework for fault diagnosis of centrifugal pump leveraging wavelet coherence analysis and deep learning," *Sensors*, vol. 23, 8850, 2023.

[20] A. K. Mustafin, D. N. Kabylkayr, T. A. Iskakova, V. V. Grebenkin, and B. K. Kairolla, "Development of methods for protecting pump units from vibration loads," *Science and Technology of Kazakhstan,* no. 2, pp. 6–9, 2019. (in Russian)

[21] W. Zaman, Z. Ahmad, and J. M. Kim, "Fault diagnosis in centrifugal pumps: A dual-scalogram approach with convolution autoencoder and artificial neural network," *Sensors*, vol. 24, no. 3, 851, 2024. doi: 10.3390/s24030851

[22] P. D. McFadden and J. D. Smith, "Model for the vibration produced by a single point defect in a rolling element bearing," *Journal of Sound and Vibration*, vol. 96, no. 1, pp. 69–82, 1984

[23] H. O. A. Ahmed, M. L. Wong, and A. K. Nandi, "Vibration image representations for fault diagnosis of rotating machines: A review," *Machines*, vol. 10, no. 12, 1113, 2022. doi: 10.3390/machines10121113

[24] S. Braun, D. Ewins, and S. S. Rao, Eds., *Encyclopedia of Vibration*, London, U.K.: Academic Press, 2001.

[25] Q. He, "Time–frequency manifold for nonlinear feature extraction in machinery fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 35, no. 1–2, pp. 200–218, 2013. doi: 10.1016/j.ymssp.2012.08.018

[26] V. Muralidharan, V. Sugumaran, and V. Indira, "Fault diagnosis of monoblock centrifugal pump using support vector machine," *Engineering Science and Technology, an International Journal*, vol. 17, pp. 153–156, 2014.

[27] N. B. Alimova and A. R. Khaitova, "Software for monitoring and remote control of pump equipment vibration," Certificate of State Registration of Computer Software, No. DGU 202407296, May 28, 2024.

[28] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing*, vol. 20, pp. 1483–1510, 2006.

[29] N. B. Alimova *et al.*, "Methods and means of control and diagnostics of technological units in the treatment of industrial wastewater based on optoelectronic and hollow light guides," in *IOP Conference Series: Earth and Environmental Science,* vol. 1043, 2022, 012007. doi:10.1088/1755-1315/1043/1/012007

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).