



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

**DZIEDZINA NAUKI INŻYNIERYJNO-TECHNICZNE**

DYSCYPLINA AUTOMATYKA, ELEKTRONIKA, ELEKTROTECHNIKA I  
TECHNOLOGIE KOSMICZNE

**ROZPRAWA DOKTORSKA**

*Badanie możliwości zastosowania niskomocowych urządzeń  
monitorowania wykorzystujących technologię MEMS do wykrywania  
uszkodzeń wybranych maszyn wirnikowych*

Autor: mgr inż. Wojciech Staszewski

Promotor rozprawy: dr hab. inż. Adam Jabłoński, prof. AGH

Praca wykonana: Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie,  
Wydział Inżynierii Mechanicznej i Robotyki

Kraków, 2023



**AGH**

AGH UNIVERSITY OF KRAKOW

**FIELDS OF SCIENCE ENGINEERING AND TECHNOLOGY**

DISCIPLINES OF SCIENCE AUTOMATION, ELECTRONICS, ELECTRICAL  
ENGINEERING AND SPACE TECHNOLOGIES

**DOCTORAL THESIS**

*Investigation of the possibility of using low-power monitoring  
devices using MEMS technology to detect damages of selected  
rotating machines*

Author: mgr inż. Wojciech Staszewski

Supervisor: dr hab. inż. Adam Jabłoński, prof. AGH

Completed in: AGH University of Krakow,

Faculty of Mechanical Engineering and Robotics

Krakow, 2023

Pracę dedykuję najważniejszej osobie w moim  
życiu, mojej żonie **Sylwii**.

Pragnę serdecznie podziękować mojemu promotorowi – Profesorowi Adamowi Jabłońskiemu za nieocenioną pomoc, wsparcie oraz cierpliwość przy realizacji niniejszej pracy. Dziękuję za cierpliwość i wyrozumiałość oraz za wszystkie wspólne projekty i prace, a także za wiele godzin spędzonych na dyskusjach naukowych i światopoglądowych.

Serdecznie podziękowania kieruję również do Profesora Tomasza Barszcza, za wieloletnie wsparcie nie tylko w trakcie kariery naukowej, ale również zawodowej, a w szczególności za nieocenione wsparcie i możliwości rozwoju.

Dziękuję również całej mojej rodzinie, za wszelkie wsparcie które od Nich otrzymałem przez wszystkie dotychczasowe lata edukacji.

Składam również serdeczne podziękowania wszystkim moim współpracownikom, którzy mnie motywowali, bądź w inny sposób przyczynili się do powstania tej pracy, a w szczególności koleżankom i kolegom z EC SYSTEMS oraz z AMC TECH.

Pracę częściowo zrealizowano w ramach projektu badawczego POIR.04.01.04-00-0080/19 pt. "Nowa era FTF dla systemów oceny stanu technicznego maszyn wirnikowych - od urządzeń on-site do serwerów Industry 4.0" NCBiR.

# Spis treści

<b>Lista skrótów</b>	<b>9</b>
<b>Streszczenie</b>	<b>14</b>
<b>Abstract</b>	<b>17</b>
<b>1 Wstęp</b>	<b>20</b>
1.1 <i>Motywacja</i>	20
1.2 <i>Cel pracy</i>	25
1.3 <i>Teza pracy doktorskiej oraz problemy badawcze</i>	26
1.4 <i>Zakres pracy</i>	28
<b>2 Wybrane aspekty systemów monitorowania stanu</b>	<b>34</b>
2.1 <i>Czym jest przetwarzanie sygnałów oraz dlaczego badamy stan techniczny maszyn?</i>	34
2.2 <i>Rola systemów monitorowania w utrzymaniu ruchu</i>	38
2.3 <i>Zadania systemów monitorowania stanu</i>	39
2.4 <i>Klasyfikacja systemów monitorowania stanu</i>	46
2.4.1 <i>Pojęcia związane z monitorowaniem stanu</i>	46
2.4.2 <i>Klasyfikacja oparta na zadaniach</i>	47
2.4.3 <i>Inne rodzaje klasyfikacji</i>	49
2.5 <i>Architektura stacjonarnych systemów monitorowania stanu</i>	50
2.6 <i>Zasady wdrożenia oraz podstawowa konfiguracja system monitorowania stanu</i>	52
2.6.1 <i>Podstawowe etapy wdrożenia</i>	52
2.6.2 <i>Wybór właściwego systemu monitorowania</i>	54
2.6.3 <i>Wstępna konfiguracja, instalacja oraz uruchomienie systemu monitorowania</i>	55
2.6.4 <i>Strojenie systemu monitorowania</i>	56
2.6.5 <i>Ostateczne strojenie</i>	57
2.7 <i>Obsługa i wsparcie w działaniu systemu monitorowania</i>	59
<b>3 Charakterystyka komponentów w drganiach generowanych przez maszyny wirnikowe</b>	<b>60</b>
3.1 <i>Pojęcie częstotliwości charakterystycznej</i>	60
3.2 <i>Podstawowe elementy mechaniczne</i>	60
3.2.1 <i>Wały</i>	60
3.2.2 <i>Łożyska toczne (REB)</i>	62

3.3	<i>Komponenty harmoniczne</i>	63
3.3.1	Dlaczego w widmie generowane są składowe harmoniczne?	63
3.3.2	Dlaczego składowe harmoniczne są generowane w widmie obwiedni?	64
3.4	<i>Dodatkowe modulacje sygnałów</i>	64
<b>4</b>	<b>Wymagania funkcjonalne dla nowoczesnego systemu monitorowania</b>	<b>66</b>
4.1	<i>Metodologia badawcza</i>	66
4.2	<i>Tablice morfologiczne poszczególnych części składowych systemu</i>	68
4.2.1	Układy zasilania	72
4.2.2	Przetworniki analogowo-cyfrowe	75
4.2.3	Moduły przechowywania danych	77
4.2.4	Możliwości komunikacyjne – wewnętrzne	78
4.2.5	Transmisja danych	80
4.2.6	Moc obliczeniowa	82
4.3	<i>Protokoły</i>	84
4.3.1	Przetwornik drgań	86
4.3.2	Jednostka akwizycji drgań	93
4.3.3	Uniwersalny system nadawania – połączenia z hostem, użytkownikiem końcowym	95
4.3.4	Bezprzewodowe protokoły komunikacyjne	97
4.3.5	Przewodowe protokoły komunikacyjne	99
4.3.6	Analiza zbiorcza	103
4.4	<i>Przegląd i porównanie najpopularniejszych systemów wbudowanych</i>	107
4.4.1	ESP32	108
4.4.2	STM32	110
4.4.3	Raspberry Pi	112
4.4.4	NVIDIA Jetson	114
4.5	<i>Wybór platformy wbudowanej</i>	116
4.6	<i>Wybór akcelerometru</i>	119
<b>5</b>	<b>Analiza właściwości mechanicznych toru przenoszenia sygnału drgań</b>	<b>123</b>
5.1	<i>Akcelerometry wykonane w technologii MEMS</i>	123
5.2	<i>Procedura analizy modalnej</i>	126
5.3	<i>Badanie wpływu lokalizacji układu scalonego</i>	129
5.3.1	Pomiar za pomocą wibrometru laserowego PCV400	135
5.3.2	Pomiar za pomocą TIRAvib TV51144	139
5.3.3	Analiza wyników	141

5.4	<i>Badanie wpływu geometrii czujnika</i>	142
<b>6</b>	<b>Projekt wbudowanego systemu monitorowania stanu</b>	<b>147</b>
6.1	<i>Analiza stanu techniki</i>	147
6.2	<i>Główne cechy prezentowanego systemu</i>	151
6.3	<i>Czym tak naprawdę jest przedstawiony system, czyli klasyfikacja układu</i>	153
6.4	<i>Przykład zastosowania nowoczesnego systemu diagnostyki</i>	155
6.5	<i>Korzystne skutki uzyskiwane z rozwiązania</i>	156
<b>7</b>	<b>Pomiary i walidacja prototypów</b>	<b>161</b>
7.1	<i>Opis stanowiska badawczego</i>	161
7.2	<i>Uszkodzenie 1 – niewyrównoważenie wału wyjściowego</i>	162
7.3	<i>Uszkodzenie 2 – łożysko toczne</i>	164
7.4	<i>Aparatura pomiarowa</i>	165
7.5	<i>Plan pomiarów</i>	168
7.6	<i>Analiza wyników</i>	169
7.6.1	Poziom szumów	169
7.6.2	Niewyrównoważenie	169
7.6.3	Uszkodzenie bieżni zewnętrznej REB	173
<b>8</b>	<b>Przetwarzanie sygnałów dla cyfrowych czujników drgań wykorzystujących technologię MEMS</b>	<b>177</b>
8.1	<i>Analiza wymagań przemysłowych</i>	177
8.2	<i>Metodologia obliczania wartości wskaźników skalarnych</i>	180
8.3	<i>Wybrane zagadnienia algorytmów przetwarzania sygnałów</i>	181
<b>9</b>	<b>Podsumowanie</b>	<b>189</b>
9.1	<i>Uwagi końcowe</i>	189
9.2	<i>Wkład kandydata</i>	191
9.3	<i>Dalsze prace</i>	193
<b>10</b>	<b>Bibliografia</b>	<b>195</b>
	<b>Spis ilustracji</b>	<b>203</b>
	<b>Spis tabel</b>	<b>207</b>

<b>Dodatek 1 – Metodologia rozwoju oprogramowania (CMSIS + CRL)</b>	<b>209</b>
<b>Dodatek 2 – Analiza wyników liczbowych</b>	<b>228</b>
<b>Dodatek 3 – Porównanie czasu wykonania FFT</b>	<b>229</b>
<b>Dodatek 4 – Porównanie czasów obliczenia wskaźników</b>	<b>230</b>
<b>Dodatek 5 – Instrukcja przekształcenia <i>dsp.BiquadFilter</i> na <i>arm_biquad_cascade_df2T_f32</i></b>	<b>232</b>
<b>Dodatek 6 – Adaptacja metod automatycznej walidacji sygnałów drgań</b>	<b>237</b>

## **Lista skrótów**

ADC - (z ang. Analog to Digital Converter) przetwornik analogowo-cyfrowy,

AFH - (ang. Adaptive Frequency-Hopping) metoda rozpraszania widma w systemach szerokopasmowych,

AP – (z ang. Access Point) urządzenie zapewniające hostom dostęp do sieci komputerowej,

ARM - (z ang. Advanced RISC Machine) architektura mikroprocesorów,

BLE – (z ang. Bluetooth Low Energy) technologia bezprzewodowej sieci osobistej,

BPFI – (z ang. Ball Passing Frequency of the Inner Race) jest to szybkość, z jaką kulka/rolka pokonuje defekt bieżni wewnętrznej,

BPFO – (z ang. Ball Passing Frequency of the Outer Race) jest to szybkość, z jaką kulka/rolka pokonuje defekt bieżni zewnętrznej,

BSF – (z ang. Ball Spin Frequency) liczba obrotów wykonywanych przez kulkę lub wałek łożyska przy każdym pełnym obrocie wału,

CAN - (z ang. Controller Area Network) to standard magistrali samochodowej zaprojektowany w celu umożliwienia mikrokontrolerom i urządzeniom komunikowania się ze sobą,

CMS - (z ang. Condition Monitoring System) to proces monitorowania parametrów stanu maszyn,

CMSIS.DSP - (z ang. Cortex Microcontroller Software Interface Standard Digital Signal Processing) niezależna od producenta warstwa abstrakcji sprzętowej dla serii procesorów Cortex-M,

CPU - (z ang. Central Processing Unit) najważniejszy procesor w danym komputerze,

CRL – (z ang. Code Replacement Library) nakładka środowiska MATLAB

DAQ - (z ang. Data Acquisition Unit) zbiór oprogramowania i sprzętu, który pozwala mierzyć lub kontrolować właściwości fizyczne,

DMA - (z ang. Direct Memory Access) umożliwia niektórym podsystemom sprzętowym dostęp do głównej pamięci systemowej niezależnie od jednostki centralnej,

DSP - (z ang. Digital Signal Processing) wykorzystanie przetwarzania cyfrowego do wykonywania szerokiej gamy operacji przetwarzania sygnałów,

DSSS - (z ang. Direct Sequence Spread Spectrum) technika modulacji o rozproszonym widmie, stosowana głównie w celu zmniejszenia ogólnych zakłóceń sygnału,

FFT – (z ang. Fast Fourier Transform) algorytm obliczający dyskretną transformatę Fouriera,

FRF - (z ang. Frequency Response Function) miara amplitudy i fazy sygnału wyjściowego w funkcji częstotliwości wejściowej,

FTF - (z ang. Fundamental Train Frequency) odpowiada liczbie obrotów koszyka łożyska przy każdym pełnym obrocie wału.,

GMA - (z ang. General Morphological Analysis) metoda badania możliwych rozwiązań wielowymiarowego, niekwantyfikowanego złożonego problemu,

GMF – (z ang. Gear Mesh Frequency) szybkość, z jaką zęby koła zębatego łączą się ze sobą w przekładni,

GPIO - (z ang. General Purpose Input/Output) pin na płytce układu scalonego lub obwodu elektronicznego, który może być używany jako wejście lub wyjście, lub jedno i drugie, i może być kontrolowany przez oprogramowanie,

GPS - (z ang. Global Positioning System) satelitarny system nawigacji radiowej należący do rządu Stanów Zjednoczonych i obsługiwany przez Siły Kosmiczne Stanów Zjednoczonych,

HDD - (z ang. Hard Disk Drive) elektromechaniczne urządzenie do przechowywania danych, które przechowuje i pobiera dane cyfrowe za pomocą pamięci magnetycznej z jednym lub większą liczbą sztywnych, szybko obracających się talerzy pokrytych materiałem magnetycznym,

I2C - (z ang. Inter-Integrated Circuit) synchroniczna, wielomasterowa/wielopodrzędna (kontroler/cel), przełączana pakietowo, jednostronna, szeregową magistralą komunikacyjną,

IEPE - (z ang. Integrated Electronics Piezo-Electric) standard techniczny

IIoT - (z ang. Industrial Internet of Things) odnosi się do połączonych czujników, instrumentów i innych urządzeń połączonych w sieć z komputerami w zastosowaniach przemysłowych, w tym w produkcji i zarządzaniu energią.,

IoT - (z ang. Internet of Things) opisuje obiekty fizyczne (lub grupy takich obiektów) z czujnikami, możliwościami przetwarzania, oprogramowaniem i innymi technologiami, które łączą się i wymieniają dane z innymi urządzeniami i systemami za pośrednictwem Internetu lub innych sieci komunikacyjnych,

ISM - (z ang. Industrial, Scientific, and Medical) części widma radiowego zarezerwowane na szczeblu międzynarodowym do celów przemysłowych, naukowych i medycznych (ISM),

ISO - (z ang. International Organization for Standardization) międzynarodowa organizacja opracowująca normy, składająca się z przedstawicieli krajowych organizacji normalizacyjnych krajów członkowskich,

KIS – (Krajowa Inteligentna Specjalizacja) obejmuje branże, których potencjał rozwojowy został najwyżej oceniony przez ekspertów,

LCD - (z ang. Liquid-crystal Display) płaski wyświetlacz lub inne elektronicznie modulowane urządzenie optyczne, które wykorzystuje modulujące światło właściwości ciekłych kryształów w połączeniu z polaryzatorami,

MEMS - (z ang. Micro-Electro-Mechanical-System) technologia mikroskopijnych urządzeń zawierających zarówno części elektroniczne, jak i mechaniczne,

MES – (Metody Elementów Skończonych) popularna metoda numerycznego rozwiązywania równań różniczkowych pojawiających się w inżynierii i modelowaniu matematycznym,

OFDM - (z ang. Orthogonal Frequency-Division Multiplexing) rodzaj transmisji cyfrowej wykorzystywanej w modulacji cyfrowej do kodowania danych cyfrowych (binarnych) na wielu częstotliwościach nośnych,

PP - (z ang. Peak-to-Peak) zmiana między szczytem (najwyższa wartość amplitudy) a dołkiem (najniższa wartość amplitudy, która może być ujemna),.

RAM - (z ang. Random-Access Memory) pamięci komputerowej, która może być odczytywana i zmieniana w dowolnej kolejności, zwykle używana do przechowywania danych roboczych i kodu maszynowego,

RCA – (z ang. Root Cause Analysis) metoda rozwiązywania problemów wykorzystywana do identyfikowania przyczyn źródłowych usterek lub problemów,

REB - (z ang. Rolling Element Bearing) łożysko, które przenosi obciążenie poprzez umieszczenie elementów tocznych (takich jak kulki lub wałeczki) pomiędzy dwoma koncentrycznymi, rowkowanymi pierścieniami zwanymi bieżniami,

RMS - (z ang. Root Mean Square) wartość zdefiniowana jako pierwiastek kwadratowy ze średniej kwadratowej (średnia arytmetyczna kwadratów) zbioru,

RTC - (z ang. Real-Time Clock) urządzenie elektroniczne (najczęściej w postaci układu scalonego), które mierzy upływ czasu,

SaaS - (z ang. Software as a Service) model licencjonowania i dostarczania oprogramowania, w którym oprogramowanie jest licencjonowane na zasadzie subskrypcji i jest centralnie hostowane,

SNR – (z ang. Signal to Noise Ratio) miara stosowana w nauce i inżynierii, która porównuje poziom sygnału użytecznego z poziomem szumu tła,

SoC - (z ang. System-On-Chip) układ scalony, który integruje większość lub wszystkie komponenty komputera lub innego systemu elektronicznego,

SPI - (z ang. Serial Peripheral Interface) specyfikacja synchronicznego szeregowego interfejsu komunikacyjnego używanego do komunikacji na krótkich dystansach, głównie w systemach wbudowanych,

SSD - (z ang. Solid-state Drive) półprzewodnikowe urządzenie pamięci masowej, które wykorzystuje zespoły układów scalonych do trwałego przechowywania danych, zazwyczaj przy użyciu pamięci flash,

TF – (z ang. Transfer Function) funkcja matematyczna, która modeluje wyjście systemu dla każdego możliwego wejścia,

TSA - (z ang. Time Series Analysis) określony sposób analizowania sekwencji punktów danych zebranych w określonym przedziale czasu,

UART - (z ang. Universal Asynchronous Receiver-Transmitter) urządzenie do asynchronicznej komunikacji szeregowej, w którym format danych i prędkości transmisji są konfigurowalne,

USB - (z ang. Universal Serial Bus) standard branżowy określający fizyczne interfejsy i protokoły służące do podłączania, przesyłania danych i zasilania hostów, takich jak

komputery osobiste, urządzenia peryferyjne, np. klawiatury i urządzenia mobilne, oraz koncentratory pośrednie.,

VRMS - (z ang. Velocity Root Mean Square) wartość pierwiastka kwadratowego z sumy kwadratów wartości prędkości,

WEP - (z ang. Wired Equivalent Privacy) algorytm bezpieczeństwa dla sieci bezprzewodowych 802.11,

WPA - (z ang. Wi-Fi Protected Access) program certyfikacji bezpieczeństwa opracowany przez Wi-Fi Alliance w celu zabezpieczenia bezprzewodowych sieci komputerowych,

ZP - (z ang. Zero-to-Peak) połowa wartości międzyszczytowej Peak-to-Peak.

## Streszczenie

W rozprawie doktorskiej przedstawiono koncepcję modelu nowatorskiego czujnika drgań wykorzystującego technologię MEMS (z ang. Micro-Electro-Mechanical-System) do oceny stanu technicznego maszyn wirnikowych w rozproszonych systemach monitorowania i diagnostyki. Rozprawa ma charakter interdyscyplinarny i przedstawia szereg badań symulacyjnych i badań rzeczywistych prototypów nowego czujnika. Badania wykonane w pracy doktorskiej miały wykazać, czy możliwe jest takie zaprojektowanie cyfrowego czujnika wykorzystującego technologię MEMS, aby można było nim zastąpić znacznie mniej elastyczne i problematyczne w produkcji, analogowe czujniki drgań wymagające skomplikowanych układów akwizycji danych.

Praca posiada kontekstową część teoretyczną dotyczącą wybranych aspektów systemów monitorowania i diagnostyki maszyn wirnikowych oraz opis komponentów drganiowych generowanych przez maszyny wirnikowe, dalej część teoretyczną określającą funkcjonalne wymagania projektowe dla nowego czujnika oraz część praktyczną obejmującą szczegółowe badania kolejnych modeli czujnika na poziomie analitycznym oraz ostateczną weryfikację spełnienia założeń projektowych (szczególnie wymagań przemysłowych) na sygnałach drgań zarejestrowanych na obiekcie rzeczywistym.

Praca doktorska była częściowo realizowana w ramach projektu badawczego nr POIR.04.01.04-00-0080/19 pt. "Nowa era FTF dla systemów oceny stanu technicznego maszyn wirnikowych - od urządzeń on-site do serwerów Industry 4.0" współrealizowanego przez Akademię Górniczo-Hutniczą im. Stanisława Staszica w Krakowie oraz przy współpracy z amerykańskimi firmami Analog Devices Inc. oraz The Machine Instrumentation Group, Ltd. Prace rozwojowe zakończone opracowaniem prototypowej wersji czujników drgań wykonywane były głównie przez firmę AMC Tech Sp. z o.o., wobec czego w pracy doktorskiej opis tej części jest odpowiednio ograniczony.

Ostateczna forma pracy doktorskiej ma charakter modułowy, gdyż część prac miała charakter badań B+R, na podstawie których uzyskano nową wiedzę. Dotyczy to głównie samej konstrukcji mechanicznej czujnika, projektu informatyczno-elektronicznego architektury komunikacji oraz modelu rozwoju oprogramowania. W ramach opracowywania części sprzętowej nowego czujnika, prace badawcze realizowane były z wykorzystaniem układów scalonych nowej generacji, których opis literaturowy był znikomy. Wiele obszarów, zwłaszcza w kontekście wpływu zastosowanych materiałów na złożoną

charakterystykę amplitudowo-częstotliwościową odpowiedzi częstotliwościowej układu było jednocześnie badane poprzez symulacje, klasyczne metody analityczne oraz specjalistyczne pomiary empiryczne. Część przyjętych rozwiązań w obszarze komunikacji została podyktowana wymogami przemysłowymi pod kątem realizacji zadań systemów oceny stanu technicznego maszyn wirnikowych (wytyczne Polskiego Komitetu Organizacyjnego oraz ISO). Z kolei model rozwoju oprogramowania wymaga wielokrotnej aktualizacji z uwagi na bieżący rozwój narzędzi wspomagających oprogramowanie, przy jednoczesnym zapewnieniu możliwości utrzymywania kodu źródłowego. W konsekwencji, w ramach realizacji pracy doktorskiej, kandydat zagłębił się dość szczegółowo w zagadnienia związane z rozwojem oprogramowania na systemach wbudowanych, znajomością różnych języków oprogramowania oraz programowania komunikacji typu multi-platform w czasie rzeczywistym.

Główne problemy badawcze podjęte w pracy doktorskiej dotyczyły projektowania topologii architektury czujnika, hybrydowej analizy charakterystyk amplitudowo-częstotliwościowych odpowiedzi częstotliwościowych czujnika (z ang. FRF - Frequency Response Function) oraz rozwoju dedykowanych metod cyfrowego przetwarzania sygnałów drgań w celu wyznaczania wartości skalarnych wskaźników oceny stanu technicznego maszyn wirnikowych oraz opracowania metodologii zautomatyzowanego rozwoju oprogramowania. Najważniejszym zagadnieniem dotyczącym głównego problemu badawczego było opracowanie sposobu na uzyskanie wymaganych własności mechanicznych czujnika (jako całego systemu) podczas kolejnych modyfikacji w zakresie architektury układów elektronicznych i interfejsów komunikacji, przy zachowaniu wymaganej funkcjonalności diagnostycznej (tj. funkcji obliczeniowych). Od samego początku w pracy doktorskiej przyjęto metodologię projektowania mechatronicznego wykorzystującego tablice morfologiczne. Kolejne wersje poszczególnych rozwiązań testowano różnymi metodami. W wyniku prowadzonych badań zaproponowano model koncepcji czujnika, który spełnił stawiane wymagania zachowując względną łatwość wykonania (cecha bardzo istotna z punktu widzenia kosztów produkcji). Poza pracami o charakterze badawczym, charakter pracy doktorskiej wymagał dość rozległego studium istniejących rozwiązań w zakresie platform wbudowanych, systemów monitorowania i diagnostyki maszyn, analizy modalnej oraz chronionych rozwiązań dotyczących się czujników drgań (zwłaszcza z uwagi na prace przygotowawcze w ramach udzielonego patentu dla kandydata).

W wyniku przeprowadzonych badań naukowych potwierdzono główną tezę pracy doktorskiej, demonstrując porównywalną skuteczność nowego czujnika wykorzystującego technologię MEMS do skuteczności typowych przemysłowych piezoelektrycznych czujników drgań w ramach możliwości detekcji badanych uszkodzeń maszyn wirnikowych, zarówno w niskich częstotliwościach (zgodnie z ISO 20816), jak i w wysokich pasmach częstotliwości strukturalnych obejmujących lokalne uszkodzenia łożysk tocznych. Jednocześnie, jak wykazano w pracy, alternatywne zastosowanie cyfrowych czujników drgań niesie za sobą szereg nowych możliwości w zakresie łatwiejszej skalowalności i większej elastyczności architektury rozproszonych systemów monitorowania i diagnostyki maszyn, zarówno pod kątem zastosowań w prostych układach PLC, jak i w nowoczesnych realizacjach w ramach Industry 4.0. Najważniejsze naukowe wyniki pracy doktorskiej obejmują patent PL 238650 B1 pt. „Sposób i urządzenie pomiarowo-transmisyjne do akwizycji danych pomiarowych układów mechatronicznych”, publikacje w czasopiśmie Diagnostyka oraz wystąpienia konferencyjne na Ogólnopolskim Sympozjum Diagnostyka Maszyn. Ponadto, niektóre fragmenty pracy doktorskiej zrealizowano w ramach obecnie (2023 rok) realizowanego projektu badawczego nr POIR.01.01.01-00-0052/22 pt. "Cyfryzacja Utrzymania Ruchu w elastycznym modelu subskrypcyjnym", którego istotą jest zwiększenie poziomu cyfryzacji przemysłu maszynowego poprzez opracowanie systemu umożliwiającego w pełni autonomiczne wykrywanie uszkodzeń maszyn, na podstawie analizy danych przesyłanych z zakładu przemysłowego do oprogramowania serwerowego, w nowoczesnej architekturze IIoT (Industrial Internet of Things), dla której nowoczesne cyfrowe czujniki drgań wykorzystujące technologię MEMS są bardzo ważnym elementem.

**Słowa kluczowe:** nowoczesne cyfrowe czujniki drgań, architektura systemów wbudowanych, przetwarzanie sygnałów, biblioteka CMSIS.DSP, projektowanie mechatroniczne, zautomatyzowane generowanie kodu źródłowego, Cortex M4, analiza symulacyjna, analiza modalna, diagnostyka maszyn wirnikowych, wskaźniki diagnostyczne, IIoT.

## **Abstract**

The dissertation presents a conceptual model of a novel vibration sensor using MEMS (Micro-Electro-Mechanical-System) technology for condition assessment of rotating machinery in distributed monitoring and diagnostic systems. The dissertation is interdisciplinary in nature and presents a series of simulation studies and real-world tests of prototypes of the new sensor. The research carried out in the dissertation was intended to demonstrate whether it is possible to design a digital sensor using MEMS technology in such a way that it can be used to replace much less flexible and problematic in production, analog vibration sensors requiring complex data acquisition systems.

The paper has a contextual theoretical part on selected aspects of monitoring and diagnostic systems for rotating machinery and a description of the vibration components generated by rotating machinery, further a theoretical part defining the functional design requirements for the new sensor, and a practical part including detailed testing of successive sensor models at the analytical level and final verification of the fulfillment of design assumptions (especially industrial requirements) on vibration signals recorded on the real object.

The dissertation was partially carried out within the framework of the research project No. POIR.04.01.04-00-0080/19 entitled "New era of FTF for rotating machinery condition assessment systems - from on-site devices to Industry 4.0 servers" co-implemented by the AGH University of Science and Technology in Krakow and in cooperation with the American companies Analog Devices Inc. and The Machine Instrumentation Group, Ltd. The development work culminating in the prototype version of the vibration sensors was carried out mainly by AMC Tech Sp. z o.o., so the description of this part in the dissertation is accordingly limited.

The final form of the dissertation is modular, as part of the work was R&D, based on which new knowledge was obtained. This mainly concerns the mechanical design of the sensor itself, the computer-electronic design of the communication architecture and the software development model. As part of the development of the hardware part of the new sensor, research work was carried out using new-generation integrated circuits, the literature description of which was scarce. Many areas, especially in the context of the influence of the materials used on the complex amplitude-frequency characteristics of the frequency response of the system were simultaneously studied through simulations, classical analytical

methods and specialized empirical measurements. Some of the solutions adopted in the area of communication were dictated by industrial requirements in terms of the implementation of the tasks of systems for assessing the technical condition of rotating machinery (guidelines of the Polish Organizing Committee and ISO). In turn, the software development model requires multiple updates due to the ongoing development of software support tools, while ensuring the ability to maintain the source code. Consequently, in the course of completing the dissertation, the candidate has gone into quite a bit of detail on software development on embedded systems, knowledge of various software languages and programming of real-time multi-platform communication.

The main research problems undertaken in the dissertation were the design of the topology of the sensor architecture, the hybrid analysis of the amplitude-frequency response characteristics of the sensor (FRF - Frequency Response Function), and the development of dedicated methods for digital processing of vibration signals to determine the values of scalar indices for assessing the technical condition of rotating machinery and the development of a methodology for automated software development. The most important issue concerning the main research problem was the development of a way to obtain the required mechanical properties of the sensor (as a whole system) during subsequent modifications in terms of electronic circuit architecture and communication interfaces, while maintaining the required diagnostic functionality (i.e., computational functions). From the very beginning, the dissertation adopted a mechatronic design methodology using morphological tables. Successive versions of each solution were tested using various methods. As a result of the research, a model of the sensor concept was proposed, which met the requirements while maintaining relative ease of manufacture (a feature very important from the point of view of production costs). In addition to the work of a research nature, the nature of the dissertation required a rather extensive study of existing solutions in the field of embedded platforms, machine monitoring and diagnostics systems, modal analysis and protected solutions pertaining to vibration sensors (especially in view of the preparatory work under the granted patent for the candidate).

As a result of the conducted scientific research, the main thesis of the dissertation was confirmed, demonstrating the comparable effectiveness of the new sensor using MEMS technology to that of typical industrial piezoelectric vibration sensors within the framework of the ability to detect investigated faults in rotating machinery, both at low frequencies (according to ISO 20816) and in high structural frequency bands covering local rolling

bearing faults. At the same time, as demonstrated in the paper, the alternative use of digital vibration sensors brings with it several new opportunities for easier scalability and greater flexibility in the architecture of distributed machine monitoring and diagnostic systems, both in terms of applications in simple PLC systems and in modern implementations within Industry 4.0. The most important scientific results of the dissertation include patent PL 238650 B1 entitled: "Method and measurement-transmission device for the acquisition of measurement data of mechatronic systems", publications in the journal *Diagnostics*, and conference presentations at the National Symposium on Machine Diagnostics. In addition, selected parts of the realized doctoral thesis were completed as part of the currently (2023) executed research project No. POIR.01.01.01-00-0052/22 entitled. "Digitization of Maintenance in a Flexible Subscription Model", the essence of which is to increase the level of digitization of the machinery industry through the development of a system enabling fully autonomous detection of machine faults, based on the analysis of data sent from the industrial plant to the server software, in the modern IIoT (Industrial Internet of Things) architecture, for which modern digital vibration sensors using MEMS technology are a very important element.

**Key words:** modern digital vibration sensors, embedded systems architecture, signal processing, CMSIS.DSP library, mechatronic design, automated code generation, Cortex M4, simulations, modal analysis, rotary machinery diagnostics, machine health indicators, IIoT.

# 1 Wstęp

## 1.1 Motywacja

Pod pojęciem diagnostyki technicznej rozumiemy ogół czynności związanych z oceną stanu technicznego maszyny przy wykorzystaniu metod niewymagających demontażu. Z tego względu głównym procesem wchodzącym w jej skład jest pomiar sygnałów diagnostycznych maszyny, a następnie porównanie tych sygnałów z pomiarami nominalnymi, wykonanymi na maszynie, której stan nie zdążył się zdegradować.

Maszyny obsługują niemal każdy aspekt rozwiniętych społeczeństw. Ludzkość polega na maszynach przy produkcji większości towarów, żywności, paliw, części, papieru, tworzyw sztucznych, etc. Łącznie maszyny napędzają 85% PKB gospodarki krajów rozwiniętych [1.1]. Kiedy maszyny zawodzą lub osiągają gorsze wyniki, linie produkcyjne przestają funkcjonować prawidłowo, co może przerwać łańcuchy dostaw. Szacuje się, że nawet do 10% globalnej produktywności jest tracone z powodu nieoczekiwanych awarii maszyn [1.2], nie wliczając w to dodatkowego wpływu niedyspozycyjności maszyn na szersze procesy oraz czasu pracy człowieka do obsługi awarii. Wraz ze wzrostem naszej zależności od maszyn rośnie również potrzeba sposobów uczynienia ich bardziej niezawodnymi, wydajnymi i skutecznymi. W odpowiedzi na te zapotrzebowania od wielu dekad na całym świecie rozwijają się różne technologie zmniejszające ryzyko awarii maszyn oraz koszty eksploatacji maszyn w trakcie okresu ich eksploatacji. A jednak maszyny wciąż zawodzą. Dzieje się tak nie wskutek braku możliwości zapobiegania awariom, lecz wskutek niedostatecznego wdrożenia systemów oceny stanu technicznego maszyn i urządzeń.

Przyczyn awarii maszyn jest kilka. Po pierwsze, poza dużymi turbozespołami (które zwykle stanowią niewielki ułamek całej populacji maszyn, tj. poniżej 2%), większość maszyn nie jest monitorowana w sposób ciągły. Innymi słowy, w większości zakładów średnio nawet 98% maszyn nie jest monitorowanych w sposób stały, lecz podlegają strategiom pomiarów interwałowych z użyciem przenośnych mierników (rzadziej analizatorów) drgań. W badaniach statystycznych prowadzonych już od lat 70-tych XX w. wykazano, że zaledwie 11% awarii można przewidzieć na podstawie pomiarów okresowych [1.3]. Innymi słowy, prawie 90% nie może być uniknionych, jeśli nie będą wdrożone systemy monitorowania ciągłego. Co ciekawe, niska skuteczność monitorowania okresowego została już zbadana przez angielskiego naukowca Conrada Hala Waddingtona (1905–1975), który wykazał, że planowane - lecz niepotrzebne - przestoje mogą wpłynąć bardzo negatywnie na całkowitą

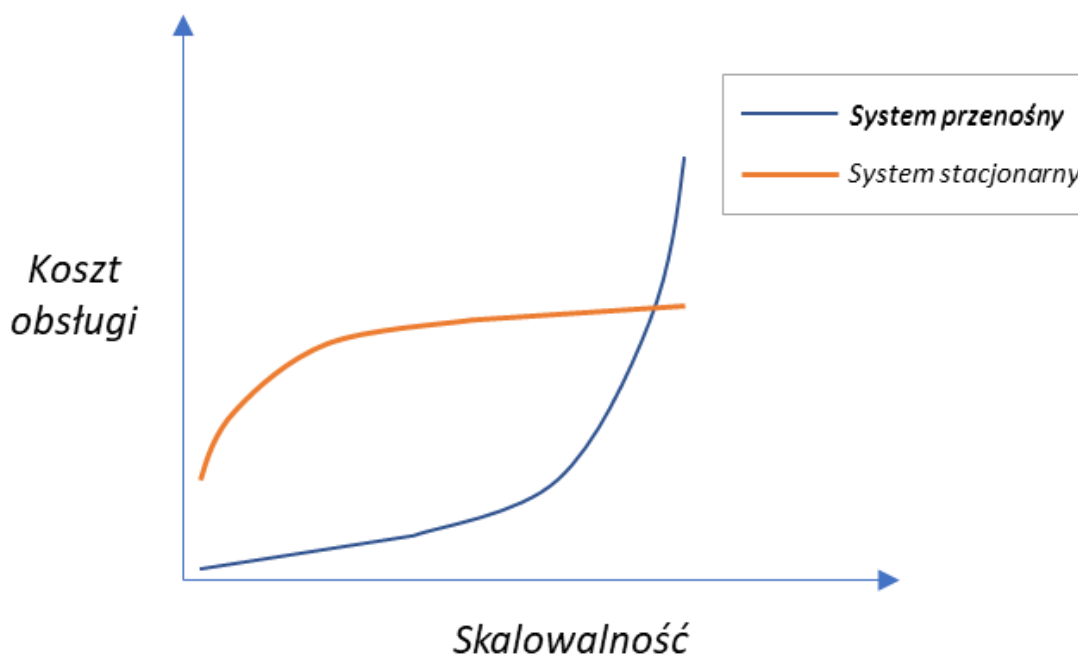
dostępność maszyny i w finalnym rozliczeniu znacząco zwiększyć koszty utrzymania ruchy [1.4].

Po drugie, wnioskowanie z danych zbieranych okresowo z definicji wymaga analizy manualnej, co skutkuje nieuniknionym czynnikiem ryzyka. Przenośne gromadzenie danych pojawiło się po raz pierwszy na początku lat 80. XX, jako alternatywa dla bardzo kosztownych (ponad 3000 USD za punkt pomiarowy) systemów monitorowania ciągłego. Chociaż nowe instrumenty redukowały koszty do jednego urządzenia, to wymagały wykwalifikowanego pracownika do obsługi - chodzenia do maszyny, zbierania danych, ustawiania alarmów, reagowania na alarmy, interpretacji danych oraz wnioskowania. Ten stan rzeczy miał miejsce do niedawna. Jednakże, w międzyczasie koszty pracy wzrosły czterokrotnie (średnio z 7 USD/godzinę do 28 USD na godzinę [1.5]). W konsekwencji, w celu utrzymania kosztów monitorowania maszyn, typową praktyką stało się wydłużanie okresów przeprowadzania pomiarów, co negatywnie wpływa na skuteczność detekcji uszkodzeń. Ręczne zbieranie danych napotyka również na problem skalowania systemów, tj. niewielkie możliwości zwiększania liczby urządzeń i częstotliwości pomiarów przy tym samym zapleczu.

W ramach nowoczesnych inicjatyw transformaty cyfrowej, systemy muszą być jednocześnie budżetowe, efektywne i skalowalne – czyli jednocześnie posiadać zalety dotychczasowych systemów ciągłych oraz dotychczasowych systemów przenośnych. Tematyka pracy doktorskiej dotyczy potencjalnego rozwiązania ważnego elementu nowego systemu, tj. cyfrowego czujnika drgań, który umożliwia skalowalność systemów oraz korzystnie wpływa na zmniejszenie kosztów instalacji. Istotą badań kandydata było takie zaprojektowanie nowego czujnika, aby ograniczyć zmniejszenie jego efektywności (w ramach wykrywania uszkodzeń) względem piezoelektrycznych czujników analogowych.

Podczas współpracy z amerykańską firmą Analog Devices Inc., USA oraz The Machine Instrumentation Group, Ltd., okazało się, że wiele rozwiązań opracowanych w ramach transformacji cyfrowej nie poradziło sobie z problemem skalowalności na etapie pilotażowym, co stanowiło motywację do pracy doktoranta. Jednocześnie, w ramach opracowania nowego rozwiązania, na podstawie konsultacji, przyjęto, że ręczne zbieranie danych w jednym miejscu po prostu nie spełnia wymogów skalowalności, gdyż ogranicza się do kilkunastu punktów w ramach jednej pętli. Ponadto, akwizycja danych urządzeniami przenośnymi siłą rzeczy separuje etap wykrywania od procesu reagowania. Programy do gromadzenia danych w urządzeniach przenośnych są zorientowane na wykrywanie stanów

uszkodzenia, a nie na podjęcie odpowiednich działań, gdyż wymaga to wiedzy kontekstowej o służbach utrzymania ruchu. W konsekwencji, skalowanie systemów monitorowania opartych na rejestracji urządzeniami przenośnymi wymaga osobnego modułu – i osobnej obsługi – do wnioskowania nt. podejmowania działań korygujących. Różnicę w zakresie wpływu zwiększania skalowalności na koszt obsługi całego systemu przedstawiono poglądowo na Rys. 1.

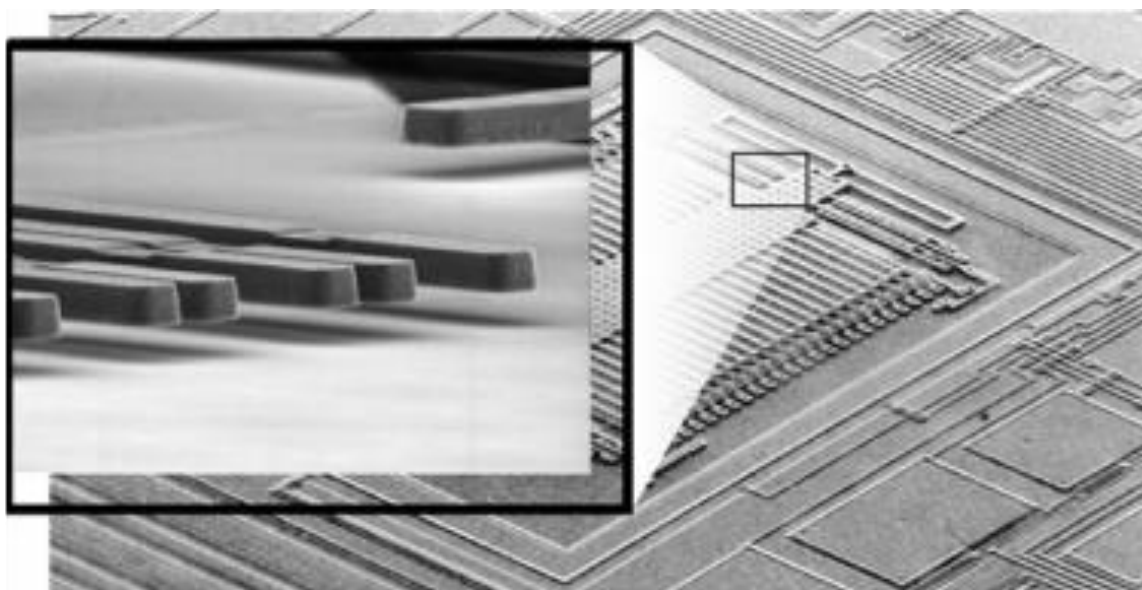


*Rysunek 1.1: Porównanie wpływu skalowalności na koszt obsługi systemu dla systemów stacjonarnych oraz przenośnych*

Jak wskazuje Rysunek 1.1, dla niewielkiej ilości punktów pomiarowych, diagnostyka prewencyjna z zastosowaniem przenośnych analizatorów do pomiarów okresowych jest rozwiązaniem korzystniejszym ekonomicznie od systemu stacjonarnego. Jednakże, wraz ze wzrostem punktów pomiarowych, koszt obsługi per punkt (a więc ostatecznie koszt obsługi całego systemu) dla systemu stacjonarnego maleje.

Z powyższych rozważań wziął pomysł na usprawnienie procesu oraz monitorowanie i diagnostykę systemów mechatronicznych przy pomocy systemów wbudowanych małej mocy, które dzięki swoim parametrom mogą być powszechniej dostępne oraz umożliwić diagnostykę maszyn, które do tej pory nie były diagnozowane.

Niniejsza praca doktorska dotyczy opracowania nowego, cyfrowego czujnika drgań. Ze względu na wykorzystanie nowoczesnej technologii MEMS, udostępnionej w ramach współpracy przez amerykańską firmę Analog Devices, Inc., nowy czujnik jest w stanie rejestrować drgania w szerokim paśmie częstotliwościowym koniecznym do diagnostyki uszkodzeń elementów maszyn generujących komponenty cyklostacjonarne drugiego rzędu, szczególnie łożysk tocznych. Dotychczasowe rozwiązania w tym zakresie posiadały pasmo wielokrotnie węższe, co ograniczało ich zastosowanie w przemyśle. Rys. 1.2 ilustruje architekturę nowoczesnej technologii MEMS.



*Rysunek 1.2: Obraz z mikroskopu elektronowego (SEM) inercyjnego akcelerometru MEMS [https://www.analog.com/en/technical-articles/mems-accelerometers-for-condition-monitoring.html]*

Cześć badań naukowych realizowanych w pracy była zdefiniowana jako wynik współpracy z amerykańską firmą Analog Devices Inc. w latach 2019-2022. Wynikiem współpracy są prototypy nowoczesnych czujników drgań opracowanych w technologii MEMS. Jednocześnie, inne prace badawcze realizowane równolegle dotyczyły zastosowania tej technologii w ramach budowy większych systemów, tj. pod kątem funkcjonalnym. Wynikiem tych prac jest m.in. wynalazek PL 423514 współautorstwa kandydata obejmujący "Sposób i urządzenie pomiarowo-transmisyjne do akwizycji danych pomiarowych układów mechatronicznych". Możliwości implementacji technologii i architektury opracowanych przez kandydata zostały pozytywnie ocenione w ramach konkursu Szybka Ścieżka NCBiR 1.1.1/2022 (Program Operacyjny Inteligentny Rozwój, Działanie 1.1 Projekty B+R

przedsiębiorstw, Poddziałanie 1.1.1 Badania przemysłowe i prace rozwojowe realizowane przez przedsiębiorstwa) poprzez zakwalifikowanie do finansowania wniosku pt. "Cyfryzacja Utrzymania Ruchu w elastycznym modelu subskrypcyjnym" nr POIR.01.01.01-00-0052/22, w ramach którego wykonano niektóre elementy niniejszej pracy. Przedmiot projektu jest bardzo istotny gospodarczo, gdyż zakłada zwiększenie poziomu cyfryzacji przemysłu maszynowego poprzez opracowanie systemu umożliwiającego w pełni autonomiczne wykrywanie uszkodzeń maszyn na podstawie analizy danych przesyłanych z zakładu przemysłowego do oprogramowania serwerowego, w nowoczesnej architekturze IIoT. Istotą proponowanego systemu jest ułatwienie dostępu do zaawansowanych metod oceny stanu technicznego maszyn wirnikowych dla licznej grupy przedsiębiorców, dla których inwestycja w przemysłowe systemy monitorowania i diagnostyki maszyn jest nieuzasadniona ekonomicznie albo zbyt kłopotliwa. Rolą czujników MEMS w tym systemie jest zastąpienie technologii analogowej i tym samym umożliwienie przesyłania danych w modelu elastycznym, w którym klasyczne jednostki DAQ wraz z dedykowanymi interfejsami użytkownika mogą być zastąpione algorytmami Machine Learning działającymi w odpowiednio przygotowanym środowisku obliczeniowym, co wymaga jedynie prostych modułów do przesyłania danych. W rezultacie, dzięki zastosowaniu opracowanej technologii, przedsiębiorcy będą mogli wdrożyć system oceny stanu w zakładach produkcyjnych, pokrywając w wielu przypadkach jedynie koszty czujników MEMS, a usługę diagnostyki uzyskać z zależności od potrzeb, wykorzystując nowe technologie, takiej jak np. SaaS (Software as a Service).

W ten sposób, technologia opracowana przez kandydata może wpisywać się w bieżący rozwój gospodarczy, gdyż umożliwia rozwój skalowalnych rozwiązań z zakresu cyfryzacji przemysłu, szczególnie z zakresu monitorowania maszyn. Jednocześnie, opracowana technologia umożliwia dalszy rozwój inteligentnych czujników drgań. Bardziej długoterminowym wyzwaniem w ramach implementacji opracowanej technologii jest opracowanie systemu chmurowego, gdzie czujniki drgań w technologii MEMS będą źródłem danych drganiowych umożliwiających tworzenie historii pracy maszyn oraz będą analizowały ich stan techniczny, co łączy podejście IIoT oraz Industry 4.0 i dostarcza funkcjonalności potrzebne do dalszej optymalizacji procesów produkcyjnych.

W ramach dalszych prac badawczych, kandydat zamierza opracowywać algorytmy automatycznej synchronizacji danych w systemach rozproszonych oraz metody automatycznego wnioskowania diagnostycznego z dużych zbiorów danych.

## 1.2 Cel pracy

Celem pracy doktorskiej jest zbadanie możliwości zastosowania niskomocowych urządzeń monitorowania wykorzystujących nowoczesną technologię MEMS do wykrywania uszkodzeń maszyn wirnikowych. Jako „niskomocowe urządzenia” zastosowano prototypy czujników drgań opartych o technologię MEMS, które były zaprojektowane w ramach współpracy z amerykańską firmą Analog Devices Inc. oraz w ramach wykonywania projektu badawczego nr POIR.04.01.04-00-0080/19 pt. ”Nowa era FTF dla systemów oceny stanu technicznego maszyn wirnikowych - od urządzeń on-site do serwerów Industry 4.0” współrealizowanego przez Akademię Górniczo-Hutniczą im. Stanisława Staszica w Krakowie. Badanie możliwości zastosowania dotyczyło analizy różnych aspektów mechatronicznych zaprojektowanych układów, z uwagi na charakterystykę implementacyjną w systemach monitorowania i diagnostyki maszyn CMS (z ang. *Condition Monitoring Systems*). Badane właściwości dotyczyły parametrów mechanicznych związanych z charakterystyką amplitudowo-częstotliwościową odpowiedzi częstotliwościowej układów oraz architektury rozwiązań elektroniczno-informatycznych do zastosowań w systemach wbudowanych. Cel pracy został osiągnięty poprzez zastosowanie podejścia interdyscyplinarnego, głównie z uwagi na to, że charakterystyka opracowanego rozwiązania wymagała weryfikacji parametrów mechanicznych (tj. odpowiedzi z urządzenia), które zależały od przyjętych rozwiązań elektroniczno-informatycznych na poziomie platformy wbudowanej czujnika. Z jednej strony, osiągnięcie celu wymagało spełnienia warunku niskiej mocy poboru prądu przez wbudowany układ obliczeniowy, a z drugiej użyteczność opracowanego prototypu wymagała zachowania właściwej charakterystyki amplitudowo-częstotliwościowej odpowiedzi częstotliwościowej układu w porównaniu z analogowymi układami piezoelektrycznymi. Opracowany układ spełnił oba wymagania, dzięki czemu umożliwia opracowanie rozproszonych systemów monitorowania i diagnostyki maszyn nowej klasy pod kątem elastyczności rozwiązań, ze szczególnym uwzględnieniem ułatwionej skalowalności. Dzięki zastosowaniu nowoczesnych czujników drgań wykorzystujących technologię MEMS tworzenie nowych systemów CMS oraz rozbudowa istniejących systemów staje się prostsza i tańsza, co stanowiło nadrzędną istotę pracy badawczej doktoranta. Kluczowe elementy rozwiązań opracowanych w ramach realizacji prac badawczych zostały zawarte w opisie wynalazku kandydata nr PL 423514 „Sposób i urządzenie pomiarowo-transmisyjne do akwizycji danych pomiarowych układów mechatronicznych”. Są to między innymi: miniaturyzacja - zmniejszenie parametrów rozmiaru znamionowego urządzenia dzięki zastosowaniu nowoczesnych układów

elektronicznych; modularyzacja - wykorzystanie niskonapięciowego procesora (układu o zmniejszonym taktowaniu pozwalający na uzyskanie mniejszego zużycia energii elektrycznej przy jednoczesnym ograniczeniu wydzielanego podczas pracy ciepła) oraz implementacja różnych sposobów transmisji danych; technologia bezprzewodowa – wykorzystanie technologii bezprzewodowej umożliwiające stosowanie urządzenia w miejscach trudno dostępnych.

### 1.3 Teza pracy doktorskiej oraz problemy badawcze

W dziedzinie diagnostyki maszyn istnieje potrzeba opracowania efektywnych rozwiązań umożliwiających uzyskanie wymaganych własności mechanicznych czujnika oraz jego architektury elektronicznej i interfejsów komunikacji. Celem tego rozdziału jest przedstawienie tezy, że możliwe jest opracowanie sposobu na osiągnięcie tych celów podczas kolejnych modyfikacji, przy jednoczesnym zachowaniu wymaganej funkcjonalności diagnostycznej.

Zanim przejdziemy do omówienia tezy, warto zdefiniować pojęcie systemu w kontekście automatyki. System można określić jako zbiór elementów wzajemnie powiązanych i oddziałujących na siebie, które razem realizują określone funkcje. W przypadku diagnostyki drgań mechanicznych, system obejmuje nie tylko sam czujnik, ale także elementy takie jak układy elektroniczne i interfejsy komunikacji, które współpracują w celu zbierania, przetwarzania i analizowania danych diagnostycznych.

Przygotowana przez autora praca doktorska skupia się na badaniu tezy:

*„Możliwe jest opracowanie sposobu na uzyskanie wymaganych własności mechanicznych czujnika (jako całego systemu) podczas kolejnych modyfikacji w zakresie architektury układów elektronicznych i interfejsów komunikacji, przy zachowaniu wymaganej funkcjonalności diagnostycznej (tj. funkcji obliczeniowych)”.*

Nowością zaproponowanego podejścia jest taki wybór komponentów systemu i takie ich rozmieszczenie, które jednocześnie zapewniają wysokie wartości parametrów akwizycji (częstotliwość próbkowania 32 kHz), niski poziom szumu (porównywalny do akcelerometrów piezoelektrycznych) oraz obliczanie wskaźników diagnostycznych kolejno z sygnału przyspieszenia, sygnału prędkości drgań oraz sygnału obwiedni (dzięki opracowaniu i implementacji nowych algorytmów cyfrowego przetwarzania sygnałów). Według wiedzy kandydata, obecnie nie ma na rynku oferty inteligentnych czujników drgań jednocześnie spełniających wszystkie wskazane kryteria jednocześnie.

**Wymagania przemysłowe** odgrywają istotną rolę w projektowaniu systemu akwizycji drgań mechanicznych. Jednym z kluczowych wymagań jest relacja czasu akwizycji do widma, która zależy od liczby obrotów maszyny. Dla maszyn o różnych prędkościach obrotowych, istotne jest dostosowanie czasu akwizycji, aby uwzględnić charakterystykę drgań w zależności od prędkości. Innym ważnym wymaganiem jest zakres widmowy  $f_s$  w stosunku do zakresu drgań strukturalnych. Oznacza to, że system musi być w stanie rejestrować drgania w odpowiednim zakresie częstotliwości, aby umożliwić skuteczną diagnostykę i analizę stanu maszyny.

Niniejsza praca doktorska dotyczy badań wybranych aspektów rozwoju nowego czujnika drgań wykorzystującego technologię MEMS, ze szczególnym uwzględnieniem możliwości jego zastosowania do wykrywania uszkodzeń maszyn wirnikowych. Główne **problemy badawcze** podjęte w pracy doktorskiej wynikają ze względnie słabych parametrów cyfrowych czujników drgań dostępnych na rynku (pasmo do 10 kHz; podstawowe analizy PP, RMS i VRMS obliczane z krótkich sygnałów drgań) i dotyczyły kolejno:

- zaprojektowania architektury czujnika,
- wykonania hybrydowej analizy charakterystyk amplitudowo-częstotliwościowych odpowiedzi częstotliwościowych czujnika (z ang. FRF - Frequency Response Function),
- rozwoju dedykowanych metod cyfrowego przetwarzania sygnałów wraz z opracowaniem metodologii ich rozwoju z wykorzystaniem nowoczesnych narzędzi IT.

Rozwiązanie pierwszego problemu badawczego, tj. zaprojektowania odpowiedniej topologii architektury nowego czujnika wykorzystującego technologię MEMS, obejmuje autorskie koncepcje kandydata w ujęciu mechatronicznym z wykorzystaniem tablic morfologicznych (zob. m.in. patent PL 238650 B1 "Sposób i urządzenie pomiarowo-transmisyjne do akwizycji danych pomiarowych układów mechatronicznych"). W ramach projektowania mechatronicznego kandydat wykonał dość obszerne studium nowoczesnych technologii sprzętowych, komunikacji i metod programowania.

Drugi problem badawczy dotyczy wyznaczenia charakterystyk amplitudowo-częstotliwościowych FRF i obejmuje jednoczesne wykonanie modeli MES, badań laboratoryjnych i testów na obiekcie rzeczywistym dla dość licznych, kolejnych wersji czujników drgań wykorzystujących technologię MEMS. Dzięki przeprowadzonym kolejnym pracom badawczym uzyskiwano wiedzę, jaki jest wpływ modyfikacji poszczególnych

elementów projektu czujnika na jego wybrane parametry modalne. W pracy zamieszczono wyniki wybranych, najistotniejszych symulacji, badań i testów. Część związana z pracami rozwojowymi nad nowym czujnikiem została mocno ograniczona w pracy z uwagi na różnego rodzaju obostrzenia wynikające z zapisów dotyczących praw własności intelektualnej w ramach realizacji projektów B+R oraz z zapisów w umowach cywilno-prawnych w ramach współpracy przemysłowej kandydata.

Trzeci problem badawczy dotyczy rozwoju dedykowanych metod cyfrowego przetwarzania sygnałów drgań w celu wyznaczania wartości skalarnych wskaźników oceny stanu technicznego maszyn wirnikowych. W ramach rozwoju metod przetwarzania sygnałów w pracy doktorskiej opisano opracowanie trzech konkretnych algorytmów cząstkowych, tj. algorytmu szybkiego obliczania tendencji rozwojowych przebiegów czasowych (dalej z ang. zwanej "detrend"), algorytmu zwiększania wiarygodności wskaźników diagnostycznych (tj. redukcji fałszywych alarmów) oraz nowego algorytmu obliczania obwiedni sygnału dla względnie małego dopuszczalnego rozmiaru bufora FFT CMSIS.DSP (funkcja `arm_rfft_fast_f32`, 4096 próbek) względem sztywnej ilości próbek sygnału wejściowego (INPUT 32000). Dodatkowo, w celach usprawnienia skróśnego modelu rozwoju oprogramowania (z ang. Host-Target), w pracy doktorskiej przedstawiono opracowanie metodologii zautomatyzowanego generowania kodu źródłowego dla wybranych platform STM32 wyposażonych w procesor Cortex-M. W tym celu kandydat opracował zautomatyzowane rozwiązanie generowania kodu źródłowego z użyciem narzędzia MATLAB Embedded Coder oraz nakładki CRL (z ang. Code Relacement Library) umożliwiającej wykorzystanie interfejsu programistycznego CMSIS.DSP (z ang. Cortex Microcontroller Software Interface Standard Digital Signal Processing). Szczegółowy opis metodologii wraz z wynikami czasowymi ilustrującymi korzyści z opracowanej metodologii oraz innymi istotnymi komentarzami przedstawiono w Dodatkach 1-6. Jako punkt odniesienia dla opracowanej metodologii, w Dodatkach zamieszczono odpowiednie zrzuty ekranów, wskazujące, że do obecnej wersji środowiska MATLAB (R2023a), na stronie producenta nie jest dostępne wsparcie w zakresie przedstawionym w pracy doktorskiej, chociaż takie opisy są przewidziane w przyszłości.

## 1.4 Zakres pracy

Niniejsza rozprawa doktorska posiada następującą strukturę. Pierwszy rozdział wprowadza czytelnika w tematykę diagnostyki technicznej przedstawiając kluczowe aspekty tego obszaru oraz punktuje najczęstsze przyczyny awarii maszyn, ich skutki oraz nakreśla,

dlaczego diagnostyka jest taka ważna. W rozdziale tym sformułowano problem badawczy i tezę, jak również cel samej rozprawy doktorskiej.

W drugim rozdziale autor przybliży zagadnienie systemu monitorowania stanu, przedstawiając jego istotę i znaczenie w dziedzinie technologii. Autor wyjaśnia, że przetwarzanie sygnałów odgrywa kluczową rolę w monitorowaniu stanu, umożliwiając analizę i interpretację danych diagnostycznych. Warto badać stan techniczny maszyn, ponieważ pozwala to na wczesne wykrywanie awarii, minimalizowanie czasu przestoju oraz zwiększanie efektywności i niezawodności procesów produkcyjnych. Następnie autor skupia się na opisie roli, jaką pełnią systemy monitorowania stanu. Przedstawia, że głównym zadaniem tych systemów jest ciągłe monitorowanie parametrów i sygnałów związanych z działaniem maszyn, w celu identyfikacji potencjalnych problemów i awarii. Systemy te służą również do gromadzenia i analizy danych diagnostycznych, umożliwiając podejmowanie odpowiednich działań naprawczych i optymalizacyjnych. Autor dokonuje również klasyfikacji systemów monitorowania, uwzględniając różne ich zadania. Wskazuje, że można je podzielić na systemy diagnostyczne, predykcyjne, kontrolne i nadzorcze. Każdy z tych rodzajów ma swoje unikalne cechy i funkcje, które przyczyniają się do skutecznego monitorowania stanu maszyn. W rozdziale autor opisuje również generyczną architekturę systemu monitorowania stanu. Przedstawia, że taka architektura składa się z czujników zbierających dane, modułów przetwarzających sygnały, systemu zarządzania danymi oraz interfejsu użytkownika. Opisuje również proces wdrożenia systemu, obejmujący jego konfigurację, instalację czujników, inicjalizację systemu oraz szkolenie personelu odpowiedzialnego za jego obsługę. Na koniec autor omawia obsługę i wsparcie systemu monitorowania stanu. Wskazuje, że konserwacja, kalibracja i regularne aktualizacje systemu są niezbędne dla zapewnienia jego niezawodności i dokładności. Autor zwraca uwagę na potrzebę szkolenia personelu i zapewnienia dostępu do wsparcia technicznego w przypadku problemów z funkcjonowaniem systemu.

Trzeci rozdział przybliży tematykę związaną z charakterystyką komponentów obecnych w drganiach generowanych przez maszyny wirnikowe. Autor rozpoczyna od omówienia pojęcia częstotliwości charakterystycznej, która jest kluczowym parametrem w analizie drgań. Wyjaśnia, że częstotliwość charakterystyczna jest wartością, która odpowiada naturalnej częstotliwości drgań systemu. Jest to istotne pojęcie, ponieważ pozwala określić dominujące częstotliwości w drganiach maszyn wirnikowych i identyfikować potencjalne problemy. Kolejnym pojęciem, które autor omawia, jest rząd charakterystyczny. Rząd charakterystyczny

odnosi się do liczby modów drgań, które dominują w analizowanym systemie. Jest to wskaźnik, który informuje o ilości składowych drgań, które mają istotny wpływ na zachowanie maszyny wirnikowej. Autor wyjaśnia, że analiza rzędu charakterystycznego pozwala na lepsze zrozumienie dynamiki systemu i identyfikację głównych czynników wpływających na generowane drgania. Następnie autor skupia się na podstawowych elementach mechanicznych wykorzystywanych w maszynach wirnikowych. Opisuje, że te elementy są kluczowe dla generowania i przenoszenia drgań w systemie. Przykładowymi podstawowymi elementami mechanicznymi są wały, wirniki, tarcze, łożyska oraz inne komponenty, które są odpowiedzialne za przekazywanie sił i generowanie drgań. Autor podkreśla, że zrozumienie charakterystyki komponentów mechanicznych jest istotne w analizie drgań maszyn wirnikowych. Pozwala to na identyfikację potencjalnych źródeł drgań, ocenę ich wpływu na cały system oraz opracowanie odpowiednich strategii eliminacji lub redukcji drgań.

W rozdziale czwartym autor pochylił się nad zebraniem wymagań funkcjonalnych dla nowoczesnego systemu monitorowania stanu technicznego maszyn. Autor rozpoczyna ten rozdział przez podzielenie całego systemu na kategorie komponentów składowych. Celem tego podziału jest umożliwienie dokładnego rozpoznania badanego problemu oraz przeprowadzenie analizy i syntezy. W dalszej części rozdziału, autor skupia się na opracowaniu tablic morfologicznych dla poszczególnych kategorii komponentów, które posłużyły do szczegółowego zrozumienia wymagań funkcjonalnych i technicznych dla każdego komponentu. Poprzez ich przygotowanie, możliwe jest uwzględnienie różnorodnych aspektów, takich jak niezawodność, precyzja pomiarów, interfejsy komunikacyjne i wiele innych. Następnie, autor przeprowadza analizę syntetyczną wszystkich komponentów. Proces ten obejmuje ocenę i porównanie różnych wariantów rozwiązań dla poszczególnych elementów systemu. Na podstawie tej analizy, autor podejmuje decyzję dotyczącą wyboru platformy wbudowanej, która będzie stanowiła podstawę do wykonania kolejnych prac. Wybór platformy wbudowanej jest kluczowy dla zapewnienia efektywnego i niezawodnego działania systemu monitorowania stanu maszyn. Autor dokładnie analizuje dostępne platformy, uwzględniając takie czynniki jak wydajność, możliwości rozbudowy, integrację z innymi systemami oraz dostępność odpowiednich narzędzi i bibliotek programistycznych. Ostatecznym efektem rozdziału czwartego jest wybór odpowiedniej platformy wbudowanej, która będzie spełniać wszystkie określone wymagania funkcjonalne. Wybór ten stanowi podstawę do kontynuacji prac nad implementacją systemu monitorowania stanu maszyn, zapewniając optymalne podstawy techniczne dla dalszych działań.

Piąty rozdział koncentruje się na analizie właściwości toru przenoszenia drgań mechanicznych. Autor rozpoczyna opis od przedstawienia podstaw analizy modalnej, która jest metodą wykorzystywaną w analizie drgań przy użyciu elementów skończonych. Wyjaśnia również charakterystykę czujników wykonanych w technologii MEMS, które są szeroko stosowane w monitorowaniu drgań mechanicznych. Następnie autor opisuje przeprowadzone badania, których celem jest wykorzystanie metody elementów skończonych (MES) do projektowania układu pomiarowego. W badaniach skupiono się przede wszystkim na dwóch aspektach. Pierwszym z nich jest wpływ lokalizacji czujnika, czyli akcelerometru, na płytce przetwornika na mierzone sygnały drgań. Analizowano, jak różne umiejscowienie czujnika może wpływać na otrzymywane wyniki pomiarów. Drugim aspektem badanych był wpływ samej geometrii przetwornika na sygnał drgań. Autor badał specjalne elementy konstrukcyjne, które miały na celu usztywnienie przetwornika i poprawę jego charakterystyk. Przeprowadzono szereg eksperymentów, aby określić, jak różne konfiguracje geometryczne przetwornika wpływają na sygnał pomiarowy. Dzięki przeprowadzonym badaniom możliwe było wybranie optymalnego modelu przetwornika do dalszych prac. Autor dokładnie analizował wyniki i porównywał różne konfiguracje, aby znaleźć najbardziej efektywny i precyzyjny model. Wybór tego optymalnego modelu stanowi kluczowy krok w projektowaniu układu pomiarowego i zapewnia wysoką jakość pomiarów oraz wiarygodność otrzymywanych danych. Rozdział piąty stanowi istotną część pracy, ponieważ skupia się na analizie właściwości toru przenoszenia drgań mechanicznych, co ma kluczowe znaczenie w monitorowaniu stanu maszyn. Przeprowadzone badania i wybór optymalnego modelu przetwornika stanowią solidne podstawy do dalszych badań i implementacji systemu monitorowania, zapewniając precyzyjne pomiary i dokładną analizę drgań mechanicznych

W rozdziale szóstym autor opisuje model wbudowanego systemu monitorowania stanu technicznego. Skupia się on na klasyfikacji rozwiązania i analizie stanu techniki, a następnie formułuje komponenty systemu i przybliża jego budowę. Na końcu rozdziału opisuje potencjalne zastosowania nowopowstałego rozwiązania, która zostało opatentowane: patent nr PL 238650 B1 „Sposób i urządzenie pomiarowo-transmisyjne do akwizycji danych pomiarowych układów mechatronicznych”.

Rozdział siódmy skupia się w całości na pomiarach i walidacji powstałego prototypu systemu monitorowania stanu. Autor rozpoczyna ten rozdział od opisu wykorzystanego stanowiska badawczego - Vibstand 2. Jest to zaawansowane urządzenie, które umożliwia wprowadzanie i badanie różnych rodzajów uszkodzeń maszyn wirnikowych. Autor wybiera

dwa konkretnie uszkodzenia do analizy: niewyrównoważenie wału wyjściowego oraz uszkodzenie łożyska tocznego. Dzięki temu można przeprowadzić precyzyjne pomiary i ocenić działanie systemu monitorowania w warunkach zbliżonych do rzeczywistych. Następnie autor przedstawia aparaturę pomiarową, która została użyta do monitorowania stanu maszyny. Opisuje dokładnie rodzaje czujników i urządzeń pomiarowych wykorzystanych w badaniach. Definiuje również plan pomiarów, który obejmuje określenie parametrów, częstotliwości oraz metody zbierania danych. Plan pomiarów jest kluczowym elementem, który zapewnia systematyczne i zorganizowane zbieranie informacji na potrzeby analizy. Kolejnym krokiem w rozdziale jest analiza otrzymanych danych. Autor dokładnie analizuje wyniki pomiarów i porównuje je z oczekiwanymi rezultatami. W przypadku niewyrównoważenia wału wyjściowego i uszkodzenia łożyska tocznego, autor ocenia wpływ tych uszkodzeń na parametry drgań maszyny i poszukuje charakterystycznych sygnałów w danych. Przeprowadza także analizę statystyczną, aby ocenić powtarzalność i stabilność otrzymanych wyników. Wreszcie, autor prezentuje wyniki pomiarów i porównuje je z założeniami i oczekiwaniami. Ocena dokładności, precyzji i wiarygodności systemu monitorowania jest kluczowym aspektem tego rozdziału. Autor przedstawia wnioski z pomiarów oraz omawia potencjalne ulepszenia i dalsze kierunki rozwoju systemu. Rozdział siódmy stanowi istotną część pracy, ponieważ koncentruje się na potwierdzeniu skuteczności i dokładności prototypowego systemu monitorowania stanu. Przeprowadzone pomiary, analiza danych i porównanie wyników z oczekiwaniami dostarczają niezbitych dowodów na efektywność systemu. Wnioski i zalecenia z tego rozdziału mogą posłużyć do udoskonalenia systemu monitorowania i wykorzystania go w praktyce

W rozdziale ósmym skoncentrowano się na zagadnieniu przetwarzania sygnałów dla cyfrowych czujników drgań. dokonano analizy wymagań przemysłowych dotyczących diagnostyki drgań, co pozwoliło na określenie kluczowych czynników wpływających na skuteczność systemu. Kolejny podrozdział poświęcony został metodologii obliczania wartości wskaźników skalarnych, które są istotne dla analizy drgań mechanicznych. Przedstawiono różne techniki obliczania takich wskaźników, uwzględniając ich zastosowanie w diagnostyce maszyn. W ostatnim podrozdziale skupiono się na wybranych zagadnieniach związanych z algorytmami przetwarzania sygnałów.

W dziewiątym rozdziale autor podsumowuje samodzielnie przeprowadzane badania naukowe oraz przedstawia wyciągnięte z nich wnioski. Na koniec kandydat określa

potencjalne kierunki do dalszych badań naukowych mających na celu dalsze potencjalnego ulepszenia zaprezentowanego systemu.

W Dodatkach 1-6 przedstawiono autorską metodologię szybkiego rozwoju oprogramowania zaawansowanych metod przetwarzania sygnałów w skrótnym modelu rozwoju oprogramowania (z ang. Host-Target) na platformę wbudowaną wyposażoną w procesor ARM Cortex-M z wykorzystaniem narzędzia MATLAB Embedded Coder oraz modułu Code Replacement Library umożliwiającego wykorzystanie bibliotek CMSIS.DSP w procesie zautomatyzowanego generowania kodu źródłowego na podstawie wzorcowej funkcji w środowisku MATLAB oraz skryptu testowego wraz z ilustracją sposobu szybkiego dostosowywania kodu do własnych potrzeb. Jako przykład demonstrujący użyto istotnego skalarnego wskaźnika diagnostycznego, jakim jest wartość skuteczna prędkości drgań, zgodnie z ISO 20816. Podany kod źródłowy oraz opis kolejnych kroków pozwalają na kompletne odtworzenie – i sprawdzenie uzyskanych wyników.

## 2 Wybrane aspekty systemów monitorowania stanu

Wstęp teoretyczny pracy ma na celu szczegółowe przedstawienie kontekstu przedmiotu pracy, tj. projektu czujnika drgań wykorzystującego technologię MEMS, jako elementu systemu monitorowania i diagnostyki maszyn wirnikowych. W przeciwieństwie do klasycznych analogowych czujników drgań, działających na zasadzie fizycznych zjawisk ściskania (albo ścinania) elementów piezoelektrycznych, czujniki cyfrowe wykonane w technologii MEMS same są systemami wbudowanymi, zdolnymi do wykonywania względnie skomplikowanych algorytmów przetwarzania sygnałów. W ujęciu funkcjonalnym, cyfrowe czujniki drgań są rozszerzeniem czujników analogowych (które przetwarzają ruch oscylacyjny obiektu na sygnał napięciowy), gdyż dodatkowo umożliwiają przesłanie samych wartości zdefiniowanych wskaźników diagnostycznych. W rozdziałach 2-4 pracy doktorskiej przedstawiono wybrane zagadnienia teoretyczne, które wyjaśniają zasadę działania systemów monitorowania i diagnostyki maszyn oraz przedstawiają ich klasyfikację i architekturę, opisują znaczenie poszczególnych wskaźników diagnostycznych w procesie oceny stanu technicznego maszyn, jak również opisują przegląd istniejących rozwiązań IT dotyczący indywidualnych części składowych czujnika. Wstęp teoretycznych wykonano na podstawie analizy literatury przedmiotu [2.1-2.11]. Pozycję [2.1] wykorzystano w opisie zagadnień związanych z teorią systemów monitorowania i komponentami sygnałów drgań. Źródła [2.2, 2.8-2.11] użyto w odniesieniu do metod diagnostyki technicznej opartych na sygnałach drgań. Pozycje [2.3] i [2.4] posłużyły do analizy sygnałów obwiedni, dalej [2.5] stanowi odniesienie do teorii analizy modalnej, z kolei pozycję [2.6] użyto jako źródło sposobu zapisu metod cyfrowego przetwarzania sygnałów w środowisku MATLAB.

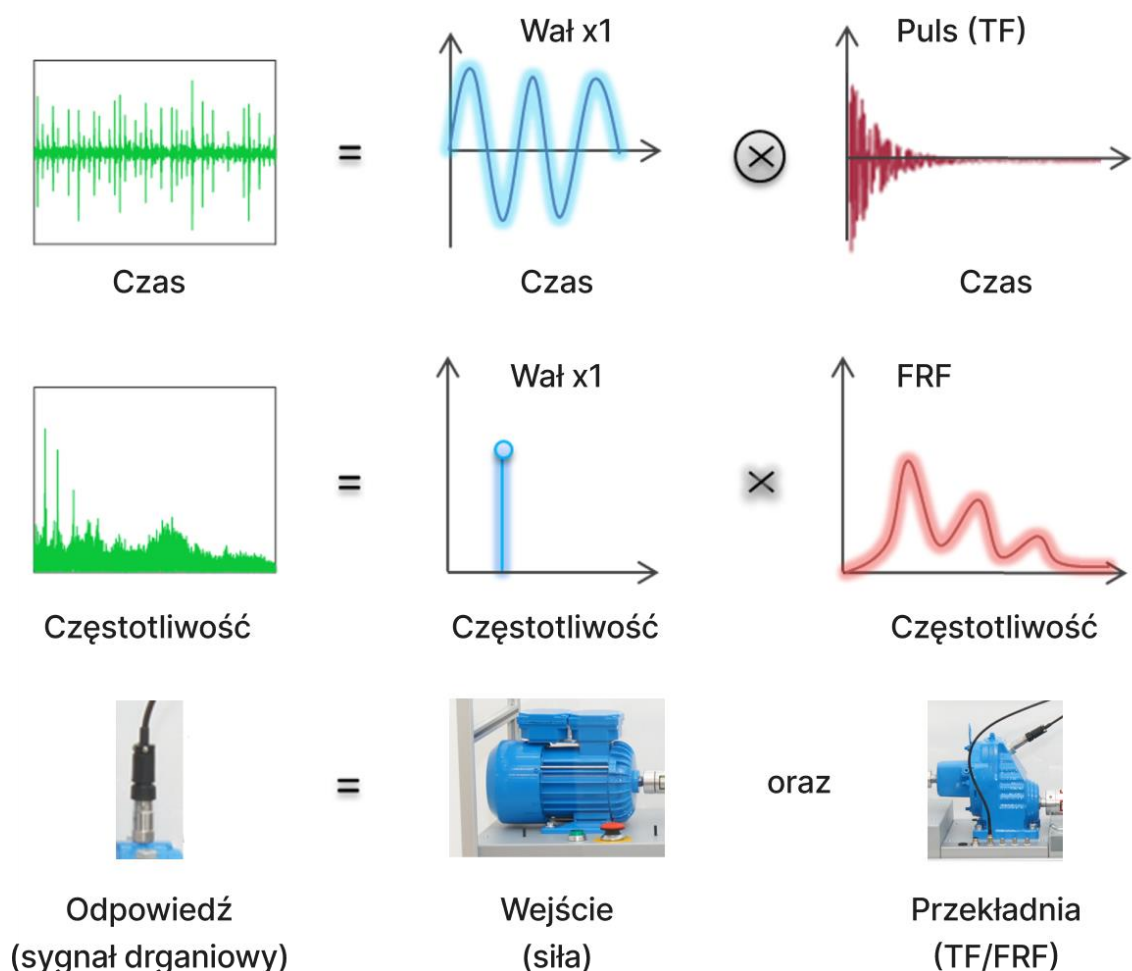
### 2.1 Czym jest przetwarzanie sygnałów oraz dlaczego badamy stan techniczny maszyn?

Pomiaru maszyn wirujących dokonuje się w głównej mierze z dwóch ogólnych powodów, po pierwsze w celu oceny jej stanu technicznego (proces ten nazywany wówczas „diagnostyką maszyny”), a po drugie w celu zbadania jej parametrów modalnych [2.12]. W pierwszym przypadku, sygnał jest rejestrowany w trakcie normalnej pracy maszyny, a następnie jest przechowywany, przetwarzany i analizowany w celu uzyskania wiedzy o aktualnym stanie technicznym, najlepiej w odniesieniu do niektórych danych referencyjnych. W tym drugim przypadku obiekt jest mierzony w celu poznania jego nieodłącznych właściwości, a mianowicie częstotliwości drgań własnych, parametrów tłumienia i kształtów modalnych. Chociaż na przestrzeni lat opracowano wiele technik analizy modalnej,

najbardziej powszechna technika wymaga kontrolowanego, mierzonego wzbudzenia siły obiektu podłączonego równoległe z układem pomiarowym odpowiedzi systemu wielopunktowego. Oba te podejścia do pomiarów są powszechnie reprezentowane w dziedzinie częstotliwości za pomocą równania:

$$\text{Response} = \text{Force} \times \text{Frequency Response Function (FRF)} \quad (1)$$

W praktycznym podejściu, „odpowiedzią” pomiaru maszyn obrotowych są rzeczywiste dane zebrane przez system akwizycji danych z czujnika, takiego jak akcelerometr, zamontowanego na obudowie elementu maszyny.

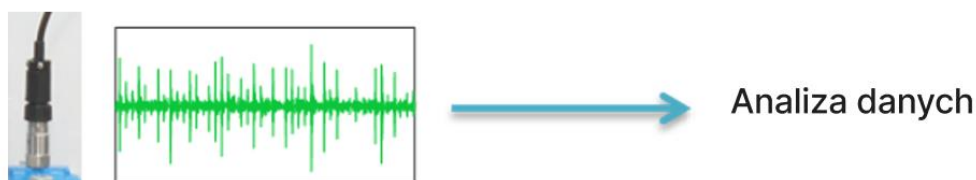


Rysunek 2.1: Zasada pomiaru przyspieszenia drgań

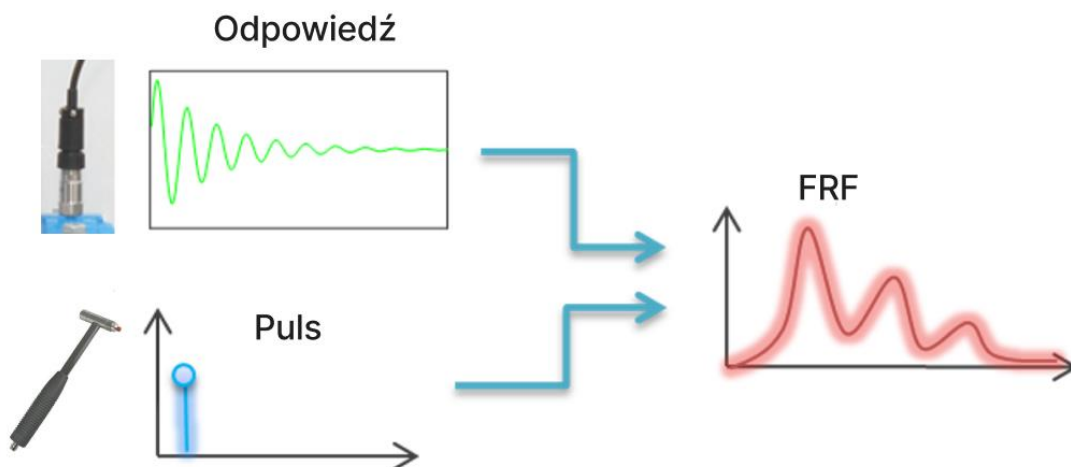
W warunkach laboratoryjnych (przy praktycznie pomijalnym poziomie szumów i rozsądnie dobranym zakresie dynamicznym „kanału”) lewa strona równania (1) jest bezpośrednio znana, jednakże jego prawa strona nabiera innego znaczenia dla dwóch opisanych powyżej podejść pomiarowych.

W momencie, kiedy przeprowadzana jest diagnostyka maszyny, zwykle mierzona jest tylko „odpowieź” systemu, co prezentuje Rysunek 2.1. W tym przypadku trudno jest określić udział „siły” w porównaniu z udziałem „funkcji odpowiedzi częstotliwościowej” (FRF – Frequency Transfer Function) w wypadkowych amplitudach widmowych mierzonego sygnału. W przypadku prostej analizy modalnej „siła” jest mierzona podczas wzbudzenia oddzielnie np. podczas postoju maszyny, co pozwala na stosunkowo łatwe obliczenie FRF obiektu. Rysunek 2.1 prezentuje tę koncepcję. W PATH 1 sygnał mierzony przez akcelerometr jest mierzony bezpośrednio. W PATH 2 sygnał siły generowany przez modalny młotek jest mierzony równoległe. W tym przypadku oba sygnały są wykorzystywane do obliczenia FRF.

### Ścieżka 1: Monitorowanie w oparciu o drgania



### Ścieżka 2: Monitorowanie w oparciu o analizę modalną



Rysunek 2.2: Ścieżki analizy pomiaru drgań

Warto zauważyć, że uproszczony schemat, który obrazuje Rysunek 2.2 odnosi się do amplitudy sygnału. Jednak FRF (podobnie jak każde przetwarzanie oparte na FFT – Fast

Fourier Transform) umożliwia analizę zarówno amplitudy w funkcji częstotliwości, jak i fazy w funkcji częstotliwości. W analizie modalnej faza jest używana na przykład do weryfikacji częstotliwości rezonansowych i wykreślania kształtów modalnych. W diagnostyce maszyn obrotowych faza jest zwykle wykorzystywana do ponownego próbkowania sygnału, co wymaga dodatkowego pomiaru danych fazowych.

Można by pomyśleć, że aby oddzielić „siłę” od FRF wystarczy zmierzyć i odjąć FRF od danych eksploatacyjnych, ale nie jest to możliwe, ponieważ FRF zatrzymanej maszyny różni się od FRF maszyny pracującej dzięki dodatkowej sile napędowej i przenoszeniu obciążeń.

W konsekwencji typowa analiza drgań w diagnostyce maszyn zakłada uproszczony scenariusz, w którym FRF obiektu jest stała, a zatem analiza ukierunkowana jest na wykrywanie, monitorowanie i analizę zmian sił wzbudzenia pochodzących od uszkodzonych silników, wałów, przekładni, ostrza, paski i łożyska. Większość technik opartych na drganiach zakłada zatem, że każda usterka w elemencie mechanicznym generuje dodatkowe siły, które są równoważone przez nowo generowane drgania, ogólnie zwane „sygnałami zainteresowania” (SOI - Signal Of Interest), oraz że głównym celem takiego pomiaru jest albo wykryć jego obecność (jeśli nie występuje wcześniej), zilustrować względny wzrost (jeśli dostępne są dane referencyjne) lub określić tempo ich wzrostu (jeśli pożądane są prognozy). W niektórych przypadkach maszyn obrotowych analiza ta jest dodatkowo problematyczna, ponieważ niektóre siły mają charakter okresowy, jak wzbudzenie zazębienia przekładni z silnika elektrycznego, podczas gdy niektóre lokalne parametry modalne są zmienne w czasie, takie jak funkcja sztywności zazębienia przekładni.

Warto zwrócić uwagę na dwie reprezentacje sygnałów zilustrowane na rysunku 1.1. Gdy sygnał jest analizowany w dziedzinie czasu, odpowiedzią jest tak zwany „splot” siły i funkcji przenoszenia (TF – Transfer Function); podczas gdy w domenie częstotliwości jest to proste pomnożenie siły i FRF (TF reprezentowane w domenie częstotliwości). Odnosząc się do rysunku, splot jest funkcją filtrowania przetwarzania sygnału z dwiema funkcjami, jako argumentami wejściowymi, siłą i TF. Dokładniej, TF jest odpowiedzią impulsową systemu, która działa jak filtr. Praktycznie splot można wyjaśnić, jako zbiór kilku operacji matematycznych na sygnale, a mianowicie odwracania sygnału, przesuwania sygnału, mnożenia i całkowania [2.13].

Szeroki zakres możliwych ścieżek przetwarzania danych w ramach analizy drgań skutkuje różnymi czujnikami, sprzętem, technologią pomiarową, wstępnym przetwarzaniem sygnału, metodami przetwarzania sygnału, technologiami wizualizacji sygnału i architekturą systemów monitorowania stanu. Niektóre firmy opracowały oparte na drganiach systemy utrzymania maszyn z wykorzystaniem przenośnych (mobilnych) analizatorów drgań do planowych pomiarów drgań. Inne polegają na bardziej zaawansowanych stacjonarnych systemach monitorowania stanu (CMS – Condition Monitoring System), które w sposób ciągły przetwarzają dane dotyczące wibracji. Niektórzy diagnostycy wykorzystują wyłącznie sygnały wibracyjne, inni dodatkowo wykorzystują temperaturę, emisję akustyczną itp. Warto wspomnieć, że różne ścieżki analizy drgań stawiają różne cele, które mogą obejmować praktycznie dowolną kombinację wykrywania uszkodzeń, identyfikacji uszkodzeń, oceny rozwoju uszkodzeń, prognozowania uszkodzeń, receptę błędów i analizę przyczyn źródłowych błędów. Wreszcie, niektóre firmy oparte na wibracjach oferują tak zwane usługi rozwiązywania problemów, które są działaniami na miejscu opartymi na sesjach, których wyniki zawsze sprowadzają się do następującej pary pytań: Co jest nie tak z moją maszyną? oraz Jak mogę to naprawić?

Mimo ogromnego rozwoju aparatury do pomiaru drgań i technik przetwarzania sygnałów wibracyjnych, powyższe stwierdzenia wydają się niezmiennie, co sugeruje, że profesjonalny pomiar maszyn nie powinien opierać się na dowolnym sprzęcie, który akurat był dostępny w chwili obecnej, ale ma charakter interdyscyplinarny. Nauka, która powinna wykorzystywać – a przynajmniej brać pod uwagę – różne techniki pomiaru i przetwarzania danych, jak na przykład koncepcyjnie zaproponowane przez Mendroka w [2.14].

## **2.2 Rola systemów monitorowania w utrzymaniu ruchu**

Potraktujmy zakład przemysłowy, jako swego rodzaju środowisko zamknięte. W takiej sytuacji, jego podstawowym zadaniem jest zapewnienie nieprzerwanej produkcji, natomiast produkcja podlega pewnym (regularnym) czynnościom konserwacyjnym. Na konserwację produkcji składa się wiele czynników i działań człowieka, które możemy podzielić na trzy główne kategorie: działania zapobiegawcze, reaktywne i proaktywne. Głównym celem każdego systemu monitorowania stanu (nie ważne o jakich zakładach produkcyjnych mówimy) jest wspieranie części proaktywnej konserwacji, dostarczając tym samym istotnych informacji niezbędnych do planowania procesu konserwacji. W takim scenariuszu główną funkcją systemu CMS jest informowanie, (jeśli to możliwe, z należywym wyprzedzeniem) personelu utrzymania ruchu i diagnostyki o zbliżającej się potencjalnej awarii elementu

maszyny. Oczekiwania wobec takiego systemu są proste: informacje dostarczone dla personelu powinny być jasne, przejrzyste i jak najbardziej jednoznaczne. Tak jak każda maszyna rotacyjna w zakładzie jest tylko kosztownym, niezbędnym narzędziem, tak każdy CMS jest kosztownym narzędziem pomocniczym w utrzymaniu produkcji.

Aby omówić pełną charakterystykę komercyjnego systemu CMS należy przeanalizować i zrozumieć wiele aspektów związanych z monitorowaniem stanu. Na początku należy odpowiedzieć na serię pytań: Jakie zadania system taki powinien realizować? Z jakich komponentów się taki system składa? W jaki sposób można sklasyfikować takie systemy? Jakie funkcjonalności powinien spełniać? W jaki sposób powinien działać, aby zrealizować żadaną funkcjonalność? W kolejnych podrozdziałach postaram się odpowiedzieć na te tematy, aby przybliżyć tematykę systemów CMS.

### 2.3 Zadania systemów monitorowania stanu

Typowy system monitorowania stanu ma kilka jasno sprecyzowanych zadań, których spełnienia oczekują jego użytkownicy. Są to między innymi: Wykrywanie i identyfikacja usterki maszyny, ocena powagi usterki, prognoza pogorszenia usterki w czasie, analiza przyczyn wystąpienia usterki, recepta w jaki sposób należy postępować.

**Wykrywanie usterek** odpowiada na prosto postawione pytanie o usterkę, to znaczy czy usterka występuje, czy maszyna pracuje prawidłowo, bez usterki. W większości przypadków, detekcja ta odbywa się na podstawie estymatorów szerokopasmowych, najczęściej Peak-to-Peak lub estymator średniej kwadratowej prędkości VRMS (Velocity Root Mean Square). W ten sposób detekcję można przeprowadzić stosunkowo niewielkim nakładem pracy, ponieważ takie systemy wymagają jedynie niewielkiej konfiguracji wstępnej.

Zakładając, że typowa implementacja systemu „*Fault Detection*” wykorzystuje estymator VRMS obliczany w zakresie od 2 Hz lub 10 Hz (wartość ta jest zależna od normatywnej prędkości pracy maszyny) do 1000 Hz. W literaturze można znaleźć wartości referencyjne tego estymatora dla wielu typów maszyn. Zgodnie z tymi wartościami referencyjnymi VRMS maszyna jest klasyfikowana jako „nowa”, z „nieograniczoną długoterminową” eksploatacją, z „krótkoterminową” dopuszczalną eksploatacją lub taka, która nie powinna już działać, ponieważ drgania pojawiające się podczas jej pracy mogą skutkować uszkodzeniem maszyny. Aby znaleźć odpowiednie referencyjne poziomy VRMS, najpierw należy odpowiedzieć na kilka pytań, a mianowicie:

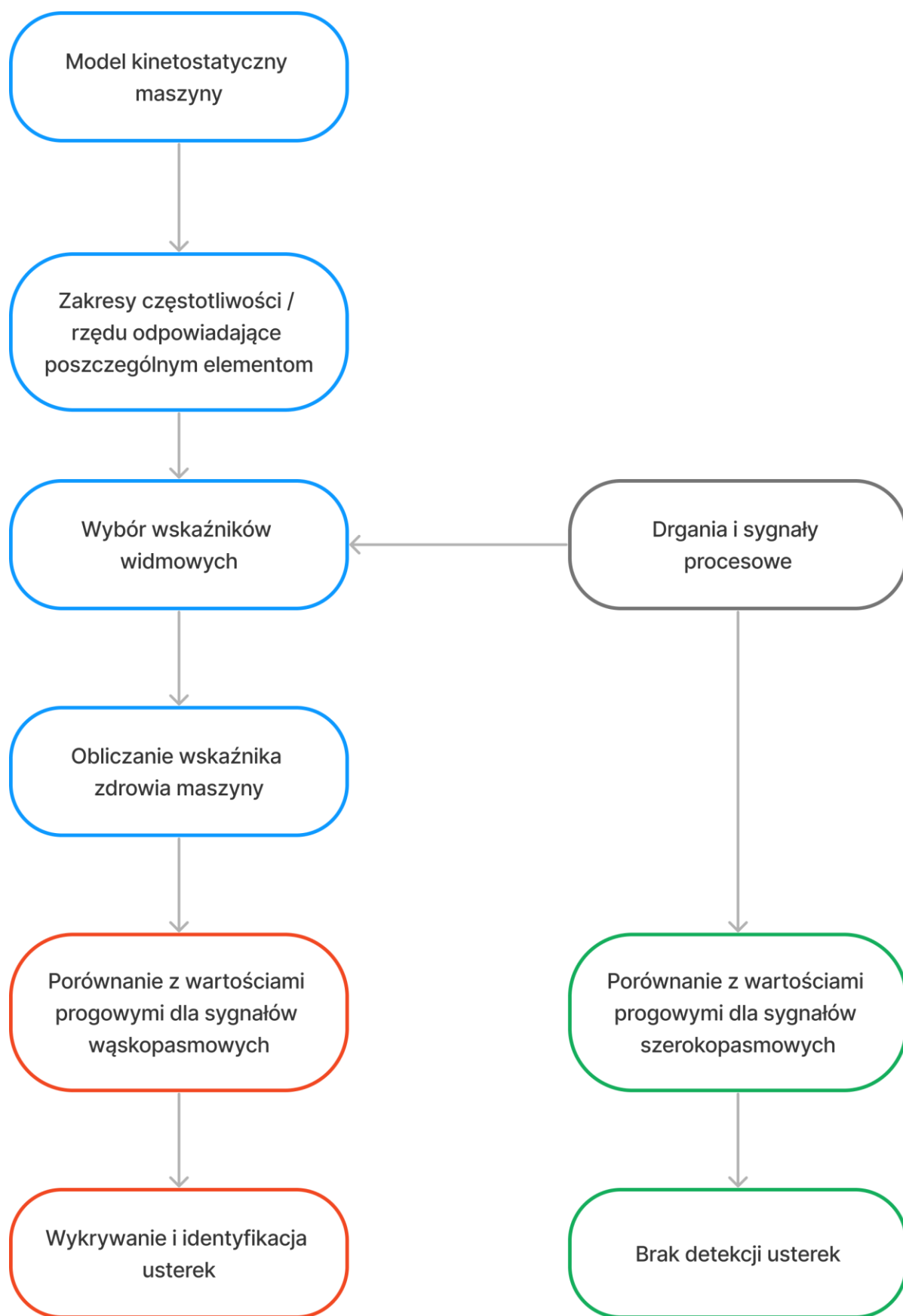
- Jaki jest typ maszyny, którą chcemy zdiagnozować?

- Jaka jest moc znamionowa tej maszyny?
- Jaka jest nominalna prędkość obrotowa maszyny?
- Czy silnik napędowy jest zintegrowany z maszyną czy zewnętrzny?
- Jaki rodzaj wału został w maszynie zastosowany- sztywny czy elastyczny?

Odpowiedź na te pytania oraz (w niektórych przypadkach) dodatkowe informacje o lokalizacji wału, pozwalają na dobór odpowiednich wartości progowych, będącymi wartościami odniesienia VRMS oraz dolnego zakresu częstotliwości VRMS (wspomniane wcześniej 2 Hz lub 10 Hz).

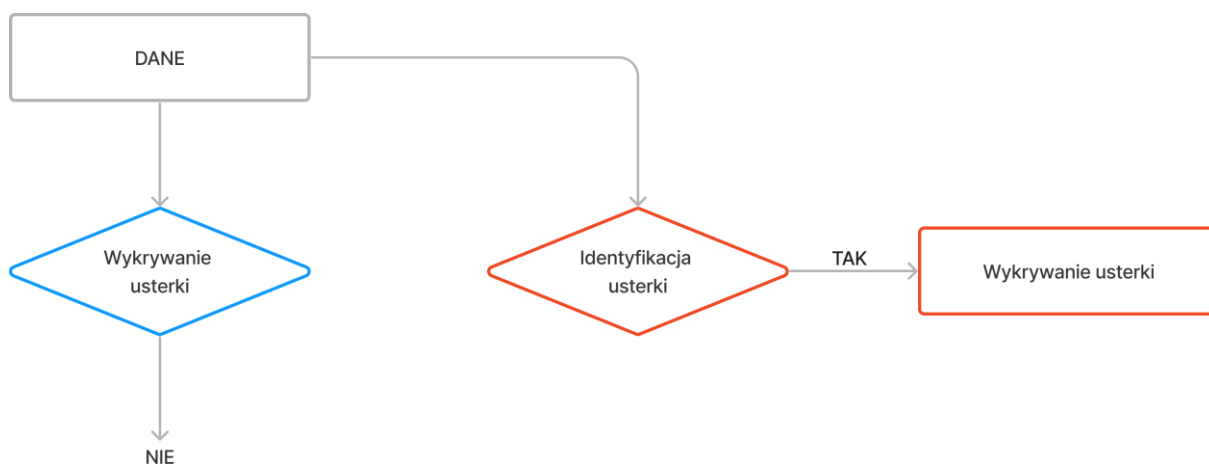
**Identyfikacja usterek** ma z kolei dwa znaczenia. Pierwsze z nich odnosi się do elementu mechanicznego, czyli odpowiada na pytanie, który element maszyny uległ awarii, czy jest to łożysko, wał czy przekładnia. Drugie znaczenie odnosi się do tzw. „trybu awaryjnego” danego elementu maszyny. Na przykład wał może być niewyrównoważony bądź zamontowany niewspółosiowo. Element przekładni może doświadczyć odprysków, łuszczenia się lub pęknięcia zęba. Łożysko z elementem tocznym REB (Rolling Element Bearing) może doświadczyć łuszczenia, wżerów lub złuszczenia na bieżni zewnętrznej i/lub wewnętrznej lub na elemencie tocznym. Co więcej, łożysko może również doświadczyć defektu koszyka itp., Dlatego pełna identyfikacja usterek często daje odpowiedzi na oba pytania, a mianowicie, który element części maszyny jest uszkodzony, a także jaki jest to rodzaj usterki.

Z tego powodu, identyfikacja usterek wymaga dodatkowego przetwarzania danych i sygnałów. Po pierwsze potrzebny jest model kinetostatyczny układu napędowego, który umożliwi generowanie analiz wąskopasmowych. Na podstawie modelu maszyny określa się zakresy częstotliwości lub zakresy rzędów w zależności od analizy odpowiednio częstotliwości lub rzędów. Następnie dla konkretnego sygnału drgań dobierane są odpowiednie indeksy widmowe, uwzględniające aktualną rozdzielczość widmową. Koncepcję tę ilustruje rysunek 2.3.



Rysunek 2.3: Po lewej: ścieżka identyfikacji uszkodzeń, po prawej: ścieżka detekcji uszkodzeń, dla pojedynczego pakietu danych

Rysunek 2.3 zawiera czerwone bloki, które oznaczają przekroczenie przez wskaźniki wąskopasmowe z dopuszczalnych poziomów referencyjnych, natomiast zielone bloki oznaczają wskaźniki szerokopasmowe, których wartości nie przekroczyły dopuszczalnych progów. Jak pokazano na rysunku 2.3, może wystąpić sytuacja, w której wszystkie estymatory szerokopasmowe zdefiniowane zadania detekcji znajdują się na dopuszczalnych poziomach, podczas gdy niektóre estymatory wąskopasmowe, przeznaczone do identyfikacji uszkodzeń, przekraczają poziomy progowe. W tym przypadku nie ma bezpośredniego wykrycia awarii, ale proces identyfikacji awarii prowadzi bezpośrednio do pośredniego wykrycia awarii, ponieważ żadne inne wskaźniki awarii nie są pozytywne. Powoduje to inny (w pewnym sensie paradoksalny) proces, jak pokazano na rysunku 2.4.



Rysunek 2.4: Pośrednie wykrywanie błędów poprzez identyfikację uszkodzeń

W tak przedstawionym scenariuszu identyfikacja błędu prowadzi pośrednio do wykrycia błędu. Dzieje się w przypadku, gdy stosunkowo duża zmiana stanu technicznego maszyny, będąca skutkiem rozwoju uszkodzenia mechanicznego nie przekłada się na stosunkowo dużą zmianę mocy sygnału.

**Ocena dotkliwości usterki** zapewnia inżynierom działu utrzymania dodatkowe informacje związane z dotkliwością lub powagą wykrytej lub zidentyfikowanej usterki. Ocena taką można przeprowadzić za pomocą specjalistycznego oprzyrządowania na miejscu lub na podstawie analizy danych, co przesunęło cel CMS z monitorowania na diagnostykę. Z punktu widzenia wibrodiagnostyki, każda usterka objawia się pewną zmianą cech charakterystycznych sygnału. Niektóre funkcje związane z dotkliwością obejmują:

- Zwiększenie wartości składowych addytywnych, synchronizowanych fazowo, deterministycznych, związanych z usterkami, takich jak podstawowa częstotliwość wału,
- Wzrost wartości składowych stochastycznych, często rozłożonych, związanych z rezonansem (czasami określanymi również jako „poziom szumu”),
- Zmiana charakterystyki zaniku impulsu w przebiegu czasowym dla REB [2.15],
- Modyfikacja układu składowych harmonicznymi (zarówno składowych nośnych, jak i SB),
- Modyfikacja stosunku składowych modulujących do fal nośnych [2.16],
- Inne.

Z biegiem lat opracowano niektóre kombinacje wskaźników stanu technicznego maszyny, takie jak współczynnik „szczytu”, aby wskazywać zmieniający się poziom dotkliwości usterki, niezależnie od poziomu sygnału. Niemniej jednak w większości przypadków wiarygodna ocena nasilenia jest częścią pełnego procesu diagnostycznego; dlatego wymaga narzędzia do zaawansowanej analizy sygnału i jest uważane za działanie nadzorowane.

**Narzędzia prognostyczne** to stosunkowo zaawansowana część, która powinna informować operatora, jak długo może bezpiecznie pracować maszyną. Choć w świecie akademickim zaprezentowano wiele prac naukowych związanych z prognozowaniem, okazuje się, że zazwyczaj wypracowane reguły prognostyczne są oparte na przypadku. Dzieje się tak, ponieważ główną ideą prognozowania jest obliczenie wartości odciętej dla domniemanej wartości rzędnej poprzez ekstrapolację krzywej dopasowanej do danych historycznych. Zatem praktyczna implementacja prognozy zakłada PRAWDA dla następujących:

- Zebrane dane reprezentują drgania maszyny,
- Zarejestrowane drgania w wystarczającym stopniu odzwierciedlają stan maszyny,
- Charakterystyki eksploatacyjne, w tym parametry eksploatacyjne maszyny są stałe,
- Poziom progowy (tj. poziom odciętej) odpowiada znaczącemu, niepożądanemu stanowi zdrowia maszyny.

Spełnienie wszystkich tych warunków jest bardzo trudne, szczególnie tego ostatniego. W dzisiejszych czasach narzędzia statystyczne, dopasowujące krzywą są bardzo popularne i stosunkowo łatwe do wdrożenia. Modelowanie statystyczne umożliwia ekstrapolację, a następnie prognozykę, do pewnej wstępnie określonej wartości z wyjątkową dokładnością.

Problem w tym, że tak dokładne obliczenia nie odzwierciedlają rzeczywistego stanu technicznego maszyny, ponieważ dane dotyczące drgań są z natury zmienne. W rezultacie, chociaż werbalnie atrakcyjne, moduły prognostyczne oparte na przesuwaniu krzywych nie znalazły jeszcze solidnego miejsca w monitorowaniu stanu maszyn w sposób systematyczny. Niektóre najnowsze zaawansowane badania w zakresie prognozowania maszyn, w tym różne techniki przewidywania pozostałej użyteczności (RUL) mogą można znaleźć w [2.17].

**Analiza przyczyn źródłowych** (RCA – Root Cause Analysis) reprezentuje kolejny poziomy zaawansowania CMS zorientowane na wyjaśnienie podstawowej, tj. głównej przyczyny rozwoju błędu. Chodzi o to, że jeśli podstawowy, niewykryty błąd „A” spowodował wykryty błąd „B”, wszelkie działania konserwacyjne rozwiązujące problem „B” są bezcelowe, ponieważ usterka „B” będzie generowana przez usterkę „A” tak długo, jak jako błąd „A”. Chociaż koncepcja RCA brzmi bardzo atrakcyjnie, rzeczywiste „przyczyny źródłowe” opisywane w literaturze są albo prostą identyfikacją błędów, albo odnoszą się do materiałoznawstwa [2.18].

**Analityka preskryptywna** ma na celu przejęcie analityki opisowej, która jest zorientowana na pytanie „Co się stało?” Wręcz przeciwnie, analiza nakazowa daje zalecenia dotyczące naprawczych lub zapobiegawczych działań konserwacyjnych. Chociaż nakazowa część CMS brzmi atrakcyjnie, do tej pory opracowano tylko kilka ogólnych, teoretycznych koncepcji [2.19]. W konsekwencji zakres wprowadzonego niedawno poziomu „Recepty” częściowo pokrywa się z zakresem wirtualnym RCA.

Tabela 2.1: Typowe zadania systemu CMS

Zadanie	Pytanie	Funkcja CMS
Wykrywanie uszkodzeń	Czy jest jakaś usterka? ISO 20816	Estymatory szerokopasmowe
Identyfikacja usterek (element mechaniczny)	Który element mechaniczny jest uszkodzony?	Model kinetostatyczny + estymatory wąskopasmowe
Identyfikacja usterek (tryb usterek)	Jaki jest rodzaj usterek (niezrównoważenie, BPFO itp.)	Zaawansowana analiza sygnału
Ocena usterek	Jak poważna jest usterka?	Analiza trendów, zaawansowana analiza sygnału
Prognozy	Jak długo maszyna może pracować bez katastrofalnej awarii?	Analiza trendów + dopuszczalne wartości progowe + narzędzia do dopasowania krzywej
RCA	Jaka jest główna przyczyna rozwoju usterek?	Zaawansowana analiza danych historycznych
Recepta	Jakich instrukcji powinien przestrzegać personel zajmujący się konserwacją?	Wysokopoziomowy, nienadzorowany moduł systemu interfejsu maszyna-człowiek (MHI)

Tabela 2.1. przedstawia główne pytanie odpowiadające każdemu zadaniu, a następnie główną funkcję CMS odpowiedzialną za jego rozwiązanie. Dodatkowo w tabeli wymieniono ogólną rodzinę funkcjonalności niezbędnych do realizacji poszczególnych zadań.

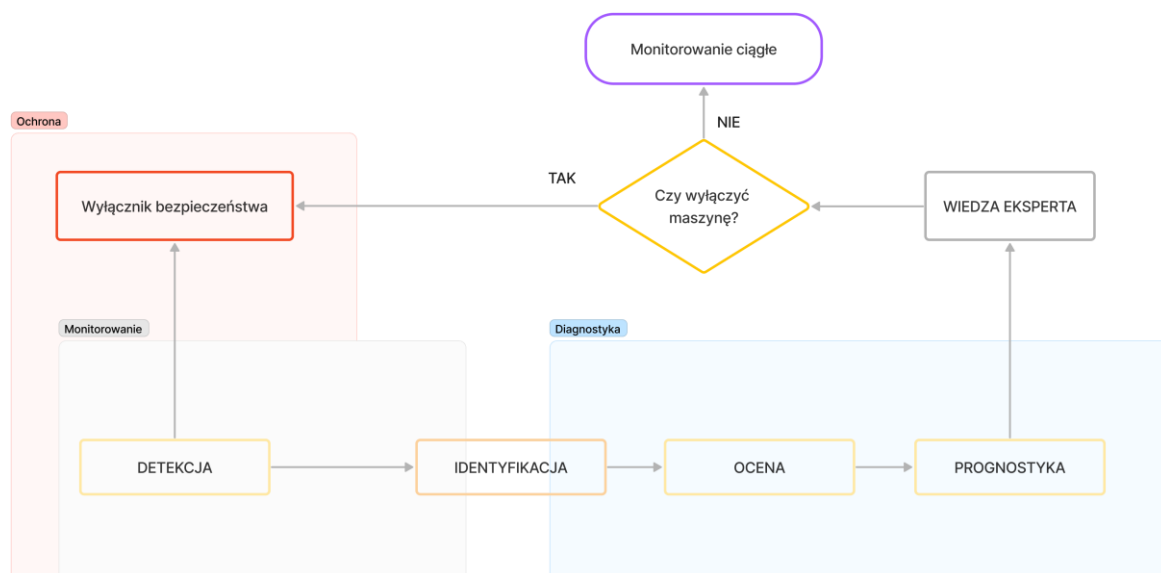
Podsumowując, wykrywanie błędów i uszkodzeń jest głównym zadaniem każdego CMS. W praktyce większość komercyjnych systemów zapewnia skuteczną częściową identyfikację usterek. Dotkliwość błędów jest praktycznie rozwiązywana przez wspólną definicję poziomu ostrzeżenia i alarmu. Reszta celów, jak na dziś, jest jeszcze w fazie rozwoju i najczęściej jest przedstawiana w literaturze naukowej lub broszurach marketingowych z uwzględnieniem indywidualnych studiów przypadku.

## 2.4 Klasyfikacja systemów monitorowania stanu

Systemy monitorowania stanu maszyn wirnikowych oparte na pomiarach drgań występują w bardzo różnych formatach. Ponieważ niektóre cechy nakładają się na różne typy systemów, wyraźnie widać, że klasyfikacja CMS nie akceptuje kształtu drzewiastego. Jednak główne kryteria klasyfikacji obejmują zadania CMS, funkcjonalność, skalę, infrastrukturę sprzętową, poziom dostosowania, format danych i typy źródeł danych. Poniżej przedstawiono wybrane szczegóły w ramach proponowanej klasyfikacji.

### 2.4.1 Pojęcia związane z monitorowaniem stanu

Chociaż nazwa „System Monitorowania Stanu” jest powszechnie używana, rzeczywiste znaczenie takich systemów nakłada na nią pewne ograniczenia. Samo pojęcie „stan” odnosi się do oceny aktualnego stanu technicznego obiektu, opcjonalnie może również zawierać prognozy o jego przyszłym stanie. Pojęcie „monitorowanie” wskazuje, że system prezentuje swoje dane wyjściowe w jakiejś formie, ale jest wyłączony ze sterowania maszyną. Prowadzi to do wyraźnego oddzielenia systemów zabezpieczeń, monitorowania i diagnostyki. Rysunek 2.5 ilustruje rozmieszczenie każdego z tych systemów w odniesieniu do ogólnych zadań CMS. Z rysunku jasno wynika, że system zabezpieczeń jest zorientowany na wykrywanie błędów i jest w stanie automatycznie wyłączyć maszynę. Z drugiej strony systemy monitorowania i diagnostyki mogą obsługiwać wszystkie zadania poza automatycznym wyłączaniem maszyny.



Rysunek 2.5: Typy i zadania systemów CMS

W praktyce wiele systemów CMS umożliwia generowanie wyjściowych sygnałów sterujących, czyniąc je po części rodzajem systemów zabezpieczających, podczas gdy inne oferują zaawansowane przetwarzanie sygnałów offline, czyniąc je bardziej systemami diagnostycznymi.

W konsekwencji powyższych, termin „System Monitorowania Stanu” nie jest obecnie wystarczający do prawidłowego zdefiniowania systemu, a aby właściwie określić jego zakres, należy wymienić kilka podstawowych cech poszczególnych urządzeń i oprogramowania.

#### 2.4.2 Klasyfikacja oparta na zadaniach

Systemy CMS oparte na wibracjach różnią się wyraźnie w zależności od ich przeznaczenia, które zazwyczaj obejmuje systemy zabezpieczające, systemy monitorowania, systemy diagnostyczne, rejestratory danych lub mieszankę niektórych lub wszystkich z nich. Każdy typ tych systemów ma kilka unikalnych cech, które zostały opisane poniżej.

Systemy **zabezpieczeń** to dobrze znane systemy wbudowane działające w czasie rzeczywistym, które stale odczytują krytyczne dane maszyny i są zawsze gotowe do wysłania sygnału o najwyższym priorytecie do systemu sterowania maszyną. Zazwyczaj takie systemy odczytują i analizują dane procesowe lub względne dane dotyczące wibracji, tj. czujniki zbliżeniowe. Systemy zabezpieczeń, choć nietypowe, mogą wykorzystywać bezwzględne sygnały wibracyjne z akcelerometrów, ale takie systemy zwykle muszą być starannie

dopasowane do konkretnej instalacji, ponieważ akcelerometry są stosunkowo podatne na wszelkie zakłócenia mechaniczne lub elektryczne i w pewnych warunkach charakteryzują się stosunkowo długim czasem ustalania sygnału. Inne cechy profesjonalnego systemu zabezpieczeń to wysokiej jakości elementy elektroniczne, wysoce niezawodne konstrukcje wbudowane w czasie rzeczywistym oraz wysokiej jakości kondycjonowanie i ekranowanie sygnału. W przeciwieństwie do innych systemów, systemy zabezpieczeń oferują ograniczone interfejsy i konfigurację. Zamiast funkcji analizy danych są one zorientowane na nienadzorowane gromadzenie i przetwarzanie danych. Gdy rozważany jest standardowy pomiar przyspieszenia, systemy zabezpieczeń są zatem ściśle powiązane z ogólnym zadaniem CMS wykrywania usterek, ponieważ zazwyczaj działają one na podstawie estymatorów szerokopasmowych opartych na normach.

Systemy **monitorowania** są zaprojektowane tak, aby dostarczać cennych informacji do planowania konserwacji. W tym celu moduły monitorujące mają dostęp do stosunkowo dużego zakresu danych. Dane te obejmują zarówno ustandaryzowane estymatory szerokopasmowe, jak i dedykowane estymatory wąskopasmowe w dowolnej domenie sygnału (w domenie czasu, częstotliwości lub porządku). Z tej perspektywy system monitorowania łączy zadania wykrywania i identyfikacji usterek. Monitoring może być zaprojektowany do pracy częściowo lub całkowicie w sposób nadzorowany. W pierwszym przypadku system monitorowania wyświetla w sposób ciągły albo serie trendów, albo odświeża wartości w widoku synoptycznym (zwanym również ekranem synoptycznym), podobnie jak w systemach SCADA. W tym drugim rozwiązaniu, typowym dla dużych systemów rozproszonych, program danych wysyła informację o tzw. zdarzeniu, które powinno wywołać szereg nadzorowanych akcji. Zgodnie z normą ISO 20816 system monitoringu powinien sprawdzać odpowiednie dane zgodnie z dopuszczalnymi poziomami niższymi i wyższymi, odpowiednio poziomami OSTRZEŻENIE i ALARM. Jako ogromne skróty w praktycznej konserwacji maszyn, poziomy te są powszechnie uważane za spełniające zadanie oceny pogorszenia stanu technicznego. Bardziej wiarygodna, w prostych przypadkach, wizualna analiza kształtu trendu może prowadzić do znacznych dedukcji prognoz maszynowych.

Systemy **diagnostyczne** to zazwyczaj w pełni nadzorowane moduły przetwarzania i wizualizacji danych. Jako standardowe działanie monitorujące, najpierw dane są starannie dobierane, filtrowane zgodnie z warunkami operacyjnymi, procesami wstępnymi i przetwarzane do odpowiedniego algorytmu dającego generalnie liczbę. Inne typowe dane diagnostyczne obejmują:

- Widmo w dziedzinach częstotliwości i rzędów,
- Uśrednione widmo w dziedzinach częstotliwości i rzędów,
- Widmo obwiedni w dziedzinach częstotliwości i rzędów,
- Uśrednione szeregi czasowe (TSA),
- Spektrogram,
- Wykresy łączone (np. wielokrotne widma lub szeregi czasowe z szeregami trendów)
- Wizualizacja na żądanie mocy ograniczonej pasmem na podstawie danych historycznych.

Wyżej wymienione dane diagnostyczne są wykorzystywane na wiele różnych sposobów. W celu identyfikacji trybu awaryjnego dane są porównywane z różnymi wzorcami charakterystycznych składowych za pomocą kursorów harmonicznym lub wstęgowym. W celu oceny dotkliwości awarii dane są porównywane z predefiniowanymi lub tymczasowo wybranymi sygnałami referencyjnymi. Do celów prognostycznych dane mogą być przetwarzane klasycznymi metodami statystycznymi lub nowoczesnymi metodami analizy danych. System diagnostyczny może mieć bardzo szeroki zakres dodatkowych funkcjonalności, takich jak ekstrakcja danych, eksport/import danych oraz przetwarzanie danych użytkownika i strumieniowanie surowych danych na żywo. Jednak w odpowiednio skonfigurowanym systemie monitorowania stanu ręczna analiza danych nie jest pożądanym działaniem i powinna mieć miejsce tylko wtedy, gdy konkretny problem nie może być w wystarczającym stopniu rozwiązany przez system monitorowania niższego poziomu.

### 2.4.3 Inne rodzaje klasyfikacji

Opisana powyżej klasyfikacja zadaniowa jest praktycznie niezależna od innych klasyfikacji, które obejmują między innymi rozmiar systemu, mobilność, źródło zasilania i funkcje formatu danych. Prosta klasyfikacja CMS pozwala podzielić systemy na: podstawowe, ręczne czujniki drgań; analizatory przenośne; systemy stacjonarne. Inna prosta klasyfikacja odnosi się do źródła zasilania i rozróżnia systemy zasilane z sieci od systemów bezprzewodowych. Warto również oddzielić systemy ogólnego przeznaczenia od systemów zorientowanych na konkretny typ maszyny lub układu napędowego. Następna klasyfikacja izoluje bazy danych lub systemy gromadzenia danych zorientowanych od lokalnej architektury przechowywania danych w plikach. Bliską charakterystyką do wymienionej powyżej jest ważna klasyfikacja systemów z wewnętrznymi zakodowanymi danymi, w których dostawca systemu w pełni kontroluje dostęp do danych oraz systemów, które dają nieograniczony dostęp do danych po ich zakupie. Co więcej, niektóre systemy są

przeznaczone tylko do przechowywania surowych danych podczas pomiaru, tj. tzw. „rejestratory danych”, inne są przeznaczone do prezentowania liczb (głównie widma częstotliwości) z aktualnie przechowywanego, stosunkowo krótkiego uśrednionego sygnału, podczas gdy niektóre są w stanie analizować zebrane dane w trybie offline.

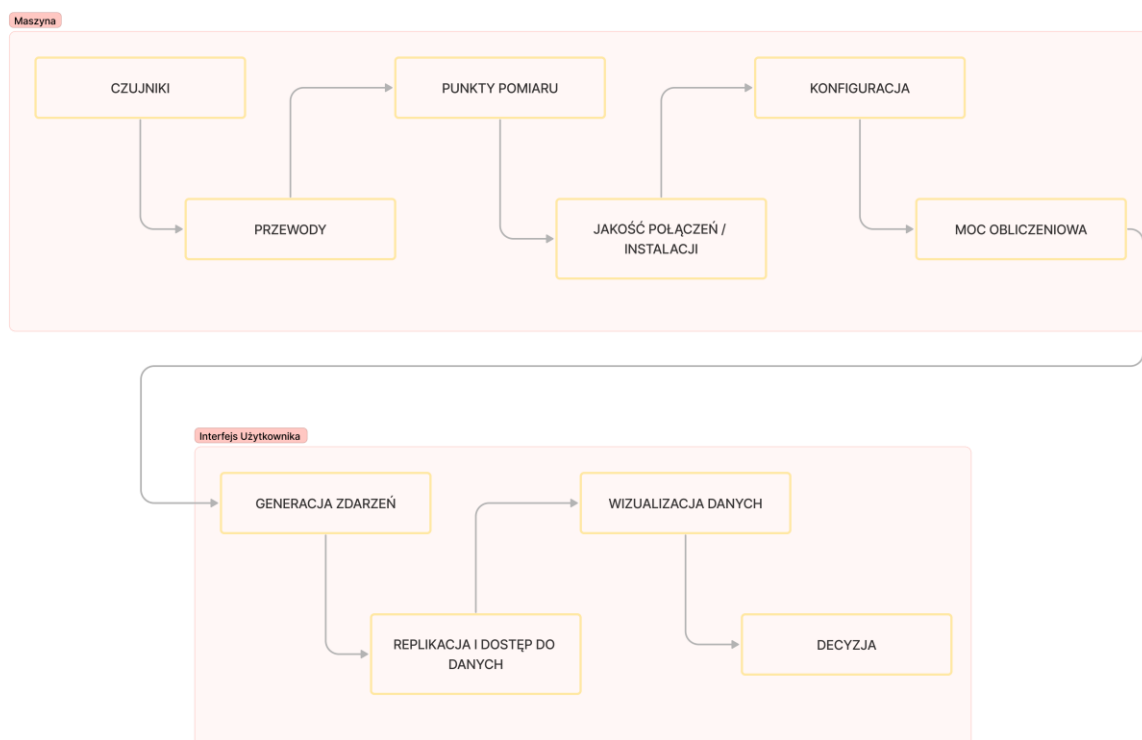
Co istotne, dowolną liczbę CMS można porównać przy użyciu listy ich indywidualnych, unikalnych pod względem częstotliwości funkcjonalności. Do takich funkcjonalności można zaliczyć m.in. mobilny dostęp do danych, transfer danych plug-and-play, inteligentny dobór danych dla maszyn pracujących w warunkach niestacjonarnych oraz praktycznie nieograniczony silnik wizualizacji danych.

## **2.5 Architektura stacjonarnych systemów monitorowania stanu**

Analizę architektury w pełni funkcjonalnego, zaawansowanego systemu CMS należy rozpocząć od powiązania jej elementów z następującymi grupami:

- Elementy sprzętowe: czujniki, okablowanie oraz DAQ i jednostka przetwarzająca,
- Dane wirtualne: pliki danych, pliki konfiguracyjne,
- Cechy systemu: logika przechowywania danych, algorytmy przetwarzania danych,
- Wydajność systemu: replikacja danych, generowanie zdarzeń, dostęp do danych, wizualizacja danych,
- Działania użytkownika: wybór punktów pomiarowych, instalacja systemu, wstępne przygotowanie konfiguracji, strojenie konfiguracji, analiza danych i podejmowanie decyzji.

Niestety, w systemach monitorowania większość elementów jest połączonych ze sobą szeregowo, jak pokazano na rysunku 2.6, więc generalnie awaria jednego bloku może łatwo spowodować awarię całego system.



Rysunek 2.6: Graficzna reprezentacja głównych elementów architektury CMS

Szczególną uwagę należy zwrócić na blok konfiguracyjny, gdyż konfiguracja jest niezbędnym elementem oprogramowania CMS, który zarządza działaniem całego systemu. Decyduje o tym, co zrobić, kiedy to zrobić i jakie dane mają być przetwarzane. Konfiguracja odnosi się zatem do wielu aspektów, w tym elementów sprzętowych, elementów oprogramowania, parametrów pomiarowych, parametrów przetwarzania sygnału i graficznego interfejsu użytkownika. Najważniejsze elementy konfiguracji obejmują:

- Częstotliwość próbkowania,
- Długość pomiaru sygnału drgań,
- Nazwy kanałów drgań,
- Zasięg kanału (wraz z czułością czujnika),
- Zakresy stanów pracy użytkowników,
- Parametry wskaźników stanu technicznego maszyny (RMS, szerokość pasma analiz wąskopasmowych itp.),
- Warunki przechowywania sygnałów surowych,
- Wartości progów ostrzegawczych i alarmowych,

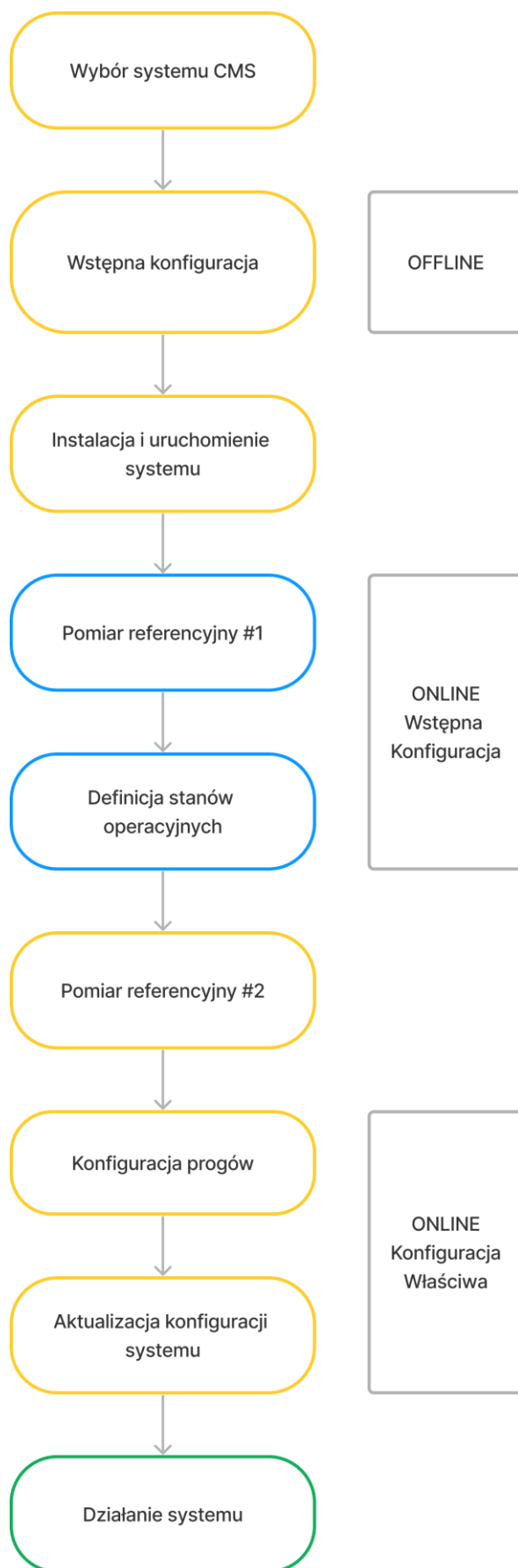
- Parametry histerezy progów ostrzegawczych i alarmowych.

Praktycznie proces konfiguracji systemu nigdy się nie kończy, ponieważ nawet jeśli struktura mechaniczna jest stała, użytkownicy CMS generalnie chcieliby ulepszać swój system zgodnie z rozwojem technologii codziennego użytku, takich jak mobilny dostęp do danych czy bezprzewodowa transmisja zdarzeń. Z drugiej strony, w zastosowaniach przemysłowych zawsze zaleca się minimalizowanie zarówno zakresu konfiguracji, jak i działań modyfikujących konfigurację, ponieważ obie z nich same w sobie nie wnoszą żadnego wkładu; są jednak stosunkowo kosztowną czynnością.

## **2.6 Zasady wdrożenia oraz podstawowa konfiguracja system monitorowania stanu**

### **2.6.1 Podstawowe etapy wdrożenia**

W praktyce, każde wdrożenie dowolnego systemu CMS do maszyn wirnikowych charakteryzuje się pewnym większym lub mniejszym poziomem dostosowania. Im bardziej „typowa” instalacja, tym łatwiejsza realizacja. Ogólną procedurę implementacji CMS przedstawiono graficznie na rysunku 2.7, który ilustruje kolejne etapy tego procesu.



Rysunek 2.7: Ogólny schemat wdrożenia stacjonarnego CMS

Pierwsze dwa kroki zilustrowane na rysunku 2.7, czyli wybór CMS i wstępna konfiguracja, są wykonywane na podstawie analizy wymagań systemowych. Następnie po zamontowaniu systemu następuje jego uruchomienie i weryfikacja. Podczas kolejnej konfiguracji post referencyjnej, na podstawie zarejestrowanych danych, wprowadzane są niektóre elementy (np. stany operacyjne użytkownika), a inne (np. długość przebiegu czasowego) są dostrajane. Po tej modyfikacji system działa przez pewien czas, a ostateczne parametry zależne od danych, takie jak próg, są zdefiniowane. W tym momencie system jest gotowy do pracy w pełnym wymiarze godzin.

Pełne zakończenie wdrożenia systemu nie oznacza, że jest on wolny od działań wspierających. Przez cały cykl życia systemu, administrator systemu jest stale odpowiedzialny za właściwe dostrojenie konfiguracji, weryfikację interakcji użytkowników oraz konserwację sprzętu. Obowiązki te obejmują monitorowanie wydajności systemu i ustawień systemu, a także długoterminową politykę sprzętową z uwzględnieniem bieżących aktualizacji technologicznych.

### **2.6.2 Wybór właściwego systemu monitorowania**

Sekcja 2.3 ilustruje klasyfikację CMS zgodnie z ich przeznaczeniem, zadaniami i cechami. Widać więc wyraźnie, że szereg możliwych rozwiązań, niezależnie od producenta, jest ogromny, a co za tym idzie wybór CMS jest trudny. Co więcej, komercyjne CMS są prawie zawsze projektami zamkniętymi (co rzekomo zmienia się przez IIoT), a pojedynczy klient rzadko kiedy ma jakikolwiek wpływ na rozwój systemu.

Wybór „właściwego” systemu CMS należy zacząć od dokładnej analizy własnych potrzeb, a nie od analizy specyfikacji technicznych konkretnego systemu czy nawet portfolio jakiegoś dostawcy. Ogólna analiza potrzeb obejmuje rodzaje mierzonych wielkości fizycznych, liczbę punktów pomiarowych, ograniczenia zasilania i okablowania, niezbędną integrację z innymi systemami, parametry akwizycji i przetwarzania danych, zakres konfiguracji systemu i pożądane narzędzia HMI. Często taka lista wymagań jest sformalizowana w tzw. dokumencie SIWZ.

Zalecane jest znalezienie najbardziej pożądanego rozwiązania dla indywidualnych potrzeb, czego efektem jest uproszczona tabela morfologiczna [2.20]. Taka tabela umożliwia bezpośrednie przekształcenie wymagań i ograniczeń funkcjonalnych na specyfikację parametrów i rozwiązań technicznych oprogramowania i sprzętu, co ilustruje tabela 2.2.

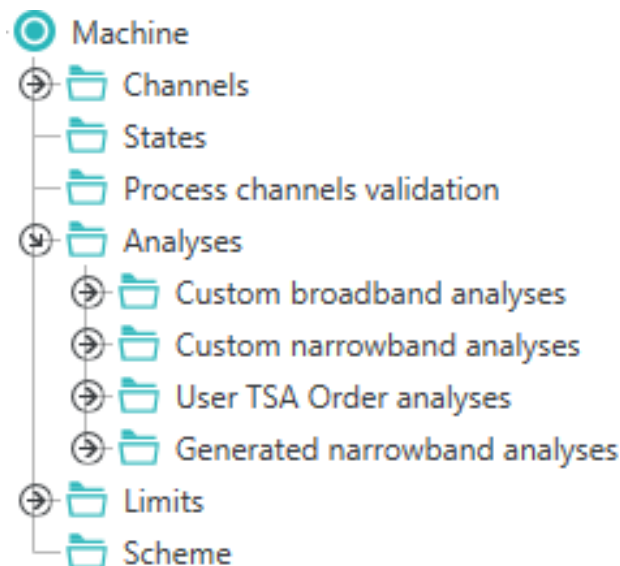
Tabela 2.2: Przykładowe elementy specyfikacji

Specyfikacja funkcjonalna	Odpowiednia funkcja systemu
Pomiar drgań o niskim poziomie szumu własnego	Kanały wibracyjne IEPE
Integracja z danymi procesowymi maszyny	Zaimplementowana obsługa protokołu
Analiza REB wysokiej częstotliwości	Zakres częstotliwości akcelerometrów minimum 10kHz
Śledzenie składowych widmowych generowane przez przekładnię	Definicja wąskopasmowego pasma spektralnego HIs
Analiza jednoczesnego przyspieszenia i prędkości	Filtry i integratory
Analiza szerokopasmowych zmian spektralnych	Wykresy kaskadowe lub wodospadowe
Indywidualna analiza danych historycznych trendów	Przechowywanie danych trendów i dodatkowy dostęp
Wyświetlanie na żywo wybranych HI na niestandardowym wykresie	Ekran synoptyczne

Jak pokazano w tabeli 2.2, specyfikacja funkcjonalna użytkownika zazwyczaj dotyczy różnych aspektów systemu, a nie tylko funkcji przetwarzania sygnałów i powinna być analizowana w znacznie szerszym zakresie. Gdy wybrany sprzęt jest gotowy, można go wstępnie skonfigurować i uruchomić.

### 2.6.3 Wstępna konfiguracja, instalacja oraz uruchomienie systemu monitorowania

W typowym scenariuszu system jest wstępnie konfigurowany zgodnie z własnymi potrzebami i ogólną specyfikacją. Typowa konfiguracja wstępna obejmuje dodatkowe parametry akwizycji danych i warunki przechowywania, a także wstępną definicję HI, w tym różne estymatory wąskopasmowe wynikające z analizy kinetostatycznej układu napędowego. Nadwyżka generalnie odzwierciedla się w stosunkowo częstym przechowywaniu stosunkowo długich surowych sygnałów czasowych w stosunkowo dużej liczbie punktów pomiarowych. Typową praktyką na tym etapie jest również dodatkowe przechowywanie danych operacyjnych z dodatkowych kanałów operacyjnych. Rysunek 2.8. przedstawia graficzną reprezentację niektórych węzłów części konfiguracji systemu z zaawansowanego, nowoczesnego CMS.



Rysunek 2.8: Częściowe drzewo konfiguracji CMS [Dzięki uprzejmości AMC Vibro®][2.1]

Wstępna konfiguracja zwykle pomija elementy konfiguracji opartej na danych, które obejmują definicję stanów użytkownika, dodatkowe ograniczenia logiczne przechowywania danych i progi. Po zainstalowaniu sprzętu system jest gotowy do uruchomienia. Podczas rozruchu wykonywana jest szczegółowa inspekcja instalacji, w tym walidacja odczytów elektronicznych, dowód wystarczającej wydajności i dostępności systemu, a następnie walidacja modułów transferu danych i przetwarzania danych. Zawsze zalecana jest również dodatkowa weryfikacja wydajności systemu. W zależności od indywidualnych cech monitorowanego obiektu, zwłaszcza biorąc pod uwagę zmienność warunków eksploatacyjnych, okres referencyjny może trwać od kilku dni do kilku tygodni. Jednak w przypadku maszyn przeznaczonych do użytku na zewnątrz, pracujących w zmieniających się warunkach temperaturowych sezonu, pełny okres odniesienia może zostać znacznie wydłużony.

#### 2.6.4 Strojenie systemu monitorowania

Na tym etapie wszystkie oprócz kilku elementów konfiguracyjnych są wprowadzane lub ostatecznie dostrajane na podstawie danych referencyjnych. Modyfikacja konfiguracji post-referencyjnej jest zwykle dokonywana na podstawie nadzorowanej analizy danych referencyjnych, wykonywanej przez doświadczonego inżyniera diagnostycznego i zazwyczaj obejmuje:

- Definiowanie stanów pracy użytkownika,
- Zdefiniowanie dodatkowych obliczeń HIs i ograniczeń przechowywania danych,

- Optymalizacja (głównie redukcja) kanałów wibracyjnych i operacyjnych,
- Modyfikacja parametrów przetwarzania danych ( $F_s$ , długości surowych sygnałów drgań),
- Modyfikacja definicji HI (dodaj/usuń),
- Modyfikacja parametrów istniejących HI (np. analizy spektralne).

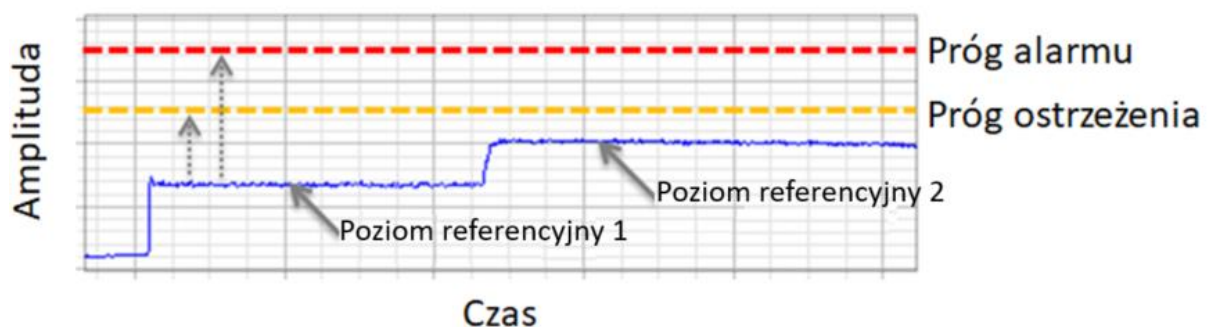
W celu maksymalizacji wydajności i minimalizacji kosztów systemu, modyfikacja jest ukierunkowana na redukcję kanałów fizycznych i HI, z wyjątkiem sytuacji, gdy nadzorowana analiza zaleca coś przeciwnego. W razie potrzeby system działa ponownie przez jakiś czas ze dostrojoną konfiguracją, zanim zostanie ostatecznie dostosowany.

### 2.6.5 Ostateczne strojenie

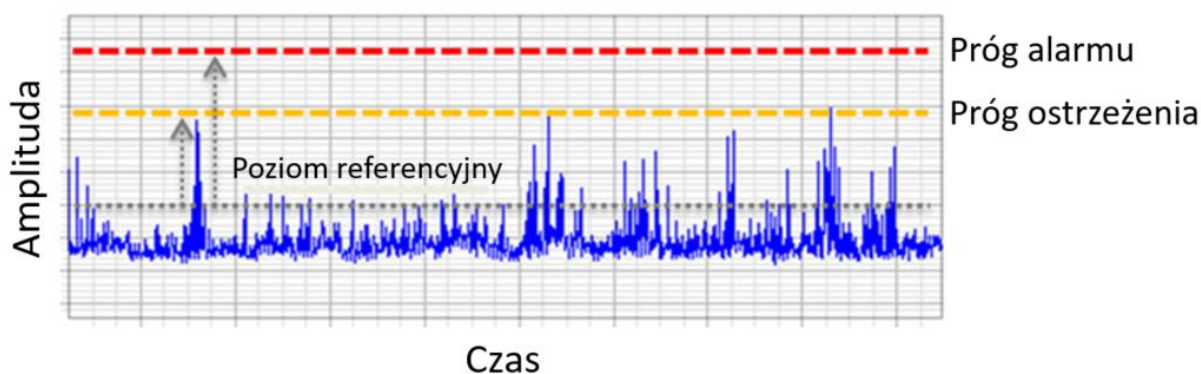
Duża część typowego, końcowego strojenia CMS dotyczy definicji progu (tj. poziomów ostrzegawczych i alarmowych) dla każdego aktywnego HI. Definiowanie poziomów alarmowych jest bardzo ważnym i bardzo trudnym zadaniem i nigdy nie powinno być wykonywane niedbale. Dla inżyniera diagnosty podstawowe pytanie brzmi:

#### *Jak ustawić poziomy alarmowe dla danego zestawu danych?*

Odpowiedź na to pytanie nie jest prosta, bo pytanie jest... za krótkie. Dane z turbiny wiatrowej mają zupełnie inną charakterystykę (np. parametry statystyczne) niż dane z turbiny gazowej o stałej prędkości; w związku z tym algorytm, poziom odniesienia i względny dopuszczalny wzrost drgań są różne. Normy oparte na wibracjach, patrz [2.21], strony 22-23 zalecają stosowanie wzrostu o 6-8 dB dla ostrzeżenia (patrz odpowiednio 2-2,5 wzrostu) i 10 dB dla alarmu (odpowiednik względnego 10-krotnego wzrostu). Jednak normy nie dają jasnej recepty na określenie tego poziomu odniesienia. Rysunek 2.9 ilustruje ten problem graficznie.



Rysunek 2.9: Przykładowe ustawienia progów – prosty przypadek [Dzięki uprzejmości AMC Vibro®]



Rysunek 2.10: Przykładowe ustawienia progów – trudny przypadek [Dzięki uprzejmości AMC Vibro®]

W przypadku stałych wartości przedstawionych na rysunku 2.9 wybór poziomu odniesienia jest jasny. Warto wspomnieć, że jeśli dane mają wiele stałych poziomów, normy zalecają uwzględnienie tylko najwyższego poziomu wibracji (w tym przypadku Poziom odniesienia 2). W przypadku danych charakteryzujących się stosunkowo dużą wariacją, zilustrowanych na rysunku 2.10, wybór poziomu odniesienia jest nie tylko trudny, ale zawsze subiektywny. W konsekwencji jego poprawność nie może być potwierdzona analitycznie. Jest to użyteczność, która zostanie potwierdzona użytecznością praktyczną. Oto kilka zaleceń dotyczących postępowania z takimi danymi:

- Oddzielne dane według zdefiniowanych przez użytkownika stanów pracy,
- Nie używaj narzędzi statystycznych opartych na rozkładzie Gaussa, jeśli Twoje dane nie przejdą testów rozkładu normalnego (zazwyczaj nie przejdą),
- Zdefiniować zasadę usuwania wartości odstających,
- Nie uwzględniają danych o poziomie szumu,
- Nie uwzględniają stosunkowo małego zbioru danych,
- Usunąć HI, które nie są przydatne,
- Utrzymuj stałe ustawienia zaawansowane (takie jak parametry histerezy) dla wszystkich HI, jeśli to możliwe,
- Unikaj bezsensownej dokładności (w większości przypadków wystarczy pierwszy przecinek),
- Okresowo sprawdzaj swoje progi wizualnie, jeśli to możliwe,
- Ponowne zdefiniowanie poziomów progowych po jakiegokolwiek modyfikacji ścieżki pomiarowej (np. inna lokalizacja czujnika, różne kable lub czujniki, różne

przetwarzanie danych) lub jakiegokolwiek czynności mechanicznej (np. wymiana części lub ponowny montaż).

Generalnie, jedynym powszechnie stosowanym wskaźnikiem drgań z akcelerometru, dla którego dopuszczalne poziomy podano w liczbach bezwzględnych, jest średni pierwiastek kwadratowy prędkości drgań, tzw. „Velocity RMS”. Dopuszczalne poziomy VRMS można bardzo łatwo znaleźć w książkach, normach lub źródłach internetowych; należy jednak wiedzieć, że klasyfikacja grupy maszyn to nie tylko moc znamionowa, ale także rodzaj fundamentu, typ konstrukcji i prędkość (dla maszyn pracujących poniżej 600 obr./min filtr dolnoprzepustowy jest ustawiony na 2 Hz).

Definicja progu wymaga dodatkowego okresu odniesienia w stanie, zwłaszcza gdy pierwsze dane odniesienia doprowadziły do zdefiniowania stanów operacyjnych użytkownika, które klasyfikują dane zgodnie z regułami użytkownika. Najczęściej zasady te opierają się na niestandardowym doborze dopuszczalnych zakresów wartości parametrów procesu lub sprzężenia zwrotnego danych o wibracjach. Ponieważ stany użytkownika mogą znacząco modyfikować poszczególne zestawy danych, zaleca się rejestrowanie nowych zestawów w celu zdefiniowania progu.

## **2.7 Obsługa i wsparcie w działaniu systemu monitorowania**

Jak wspomniano w rozdziale 2.4, lista standardowych działań użytkownika w ramach korzystania z CMS jest stosunkowo ograniczona. W najbardziej pożądanym scenariuszu użytkownik CMS w ogóle nie wykonuje żadnej interakcji z CMS, a jest tylko odbiorcą istotnych informacji z systemu. Praktyka przemysłowa pokazuje jednak, że systemy te wymagają uwagi. Jedną z takich wszechobecnych wad CMS są fałszywe alarmy generowane przez przekroczenia progów. W takim przypadku operator CMS musi albo na nowo zdefiniować progi, albo naprawić wadliwy element ścieżki pomiaru drgań.

Praktyka przemysłowa wyraźnie pokazuje, że po pewnym czasie działania CMS inżynier utrzymania ruchu uważa, że dane rozwiązanie CMS jest pomocne w konserwacji lub jest sfrustrowany dodatkowym obciążeniem niezbędnym do „utrzymania CMS”.

## 3 Charakterystyka komponentów w drganiach generowanych przez maszyny wirnikowe

### 3.1 Pojęcie częstotliwości charakterystycznej

W najbardziej pożądanym scenariuszu każdy element mechaniczny danego układu napędowego generuje unikalny zestaw elementów wibracyjnych, z których generowana jest sygnatura drgań. Rozważana analiza drgań zakłada, że każdy element mechaniczny układu napędowego maszyny wirującej generuje ruch powtarzalny, oscylacyjny – a więc okresowy, którego okres dla wygody zamieniany jest na częstotliwość.

Częstotliwości charakterystyczne umożliwiają kojarzenie składowych widmowych z elementami mechanicznymi. Jednak ze względu na wahania prędkości elementy te mają tendencję do rozmazywania się. Warto zwrócić uwagę, że im wyższa częstotliwość komponentu, tym bardziej podatny jest na rozmazanie widmowe. W tym celu analiza oparta na drganiach preferuje stosowanie tak zwanej analizy „rzędów”, w której komponenty są umieszczone na znormalizowanej skali prędkości odniesienia, zamiast analizy częstotliwości.

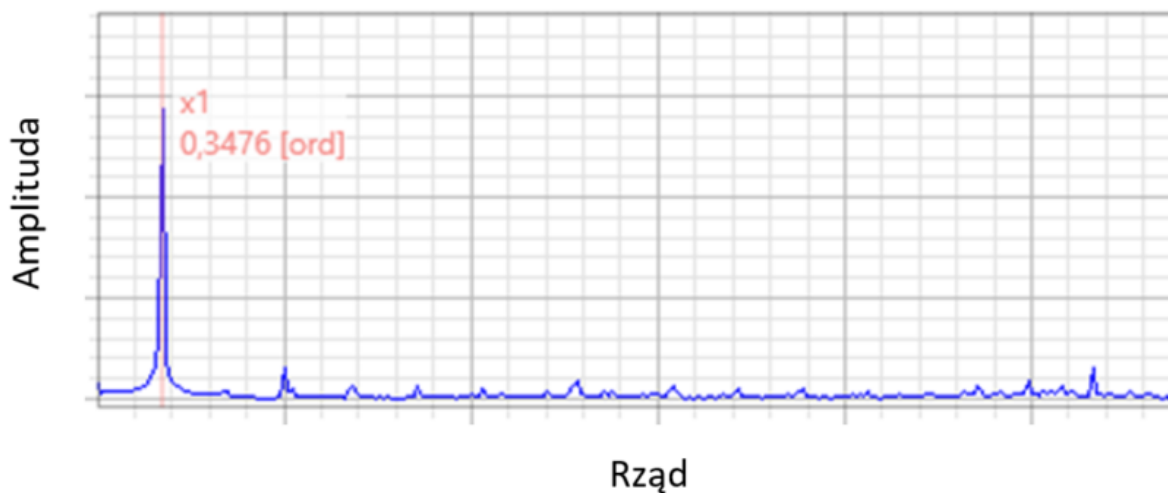
### 3.2 Podstawowe elementy mechaniczne

#### 3.2.1 Wały

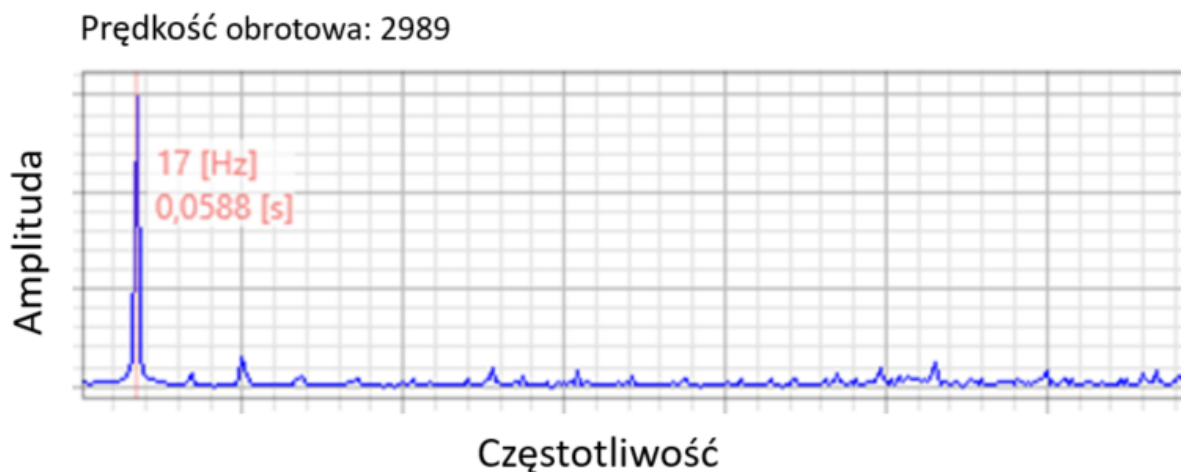
Wały generują wielokrotności prędkości wału. Zazwyczaj pierwsze trzy harmoniczne prędkości wału są śledzone w celu wykrycia niewyważenia lub niewspółosiowości. Biorąc pod uwagę układ napędowy z reduktorem o przełożeniu 23/67 i niewyważeniem na wale wolnym, charakterystyczne rzędy są następujące:

- Rząd wału: 1
- Przełożenie przekładni:  $23/67 = 0,343$
- Kolejność wałów wolnych: 0,343

Odpowiednie widma rzędów przedstawiono na rysunku 3.1.



Rysunek 3.1: Niewyważenie wału w widmie rzędów [Dzięki uprzejmości AMC Vibro®]



Rysunek 3.2: Niewyważenie wału w widmie częstotliwości [Dzięki uprzejmości AMC Vibro®]

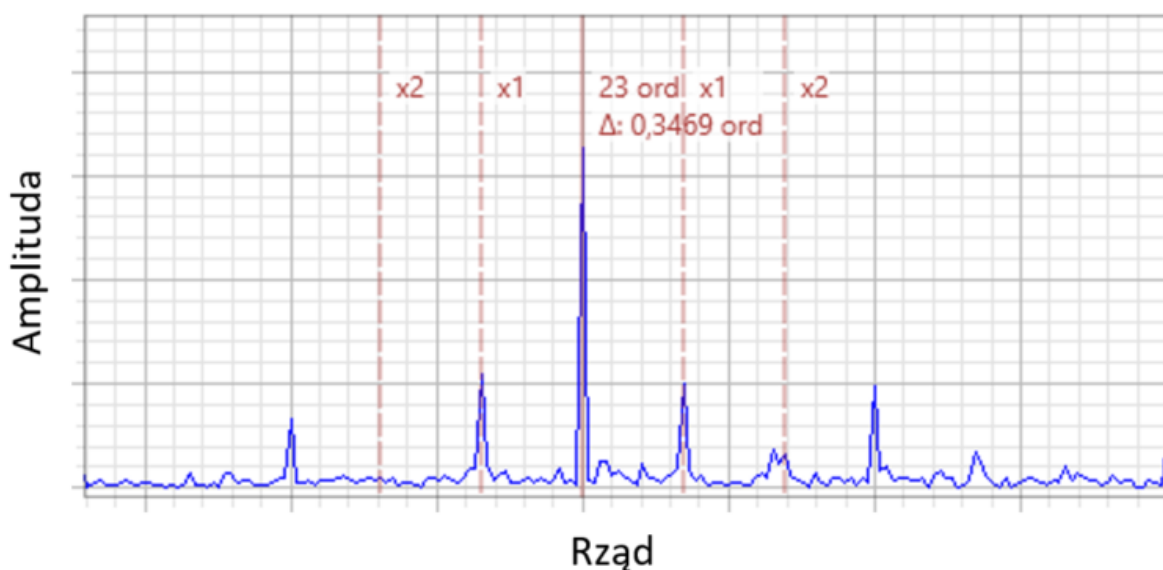
Do analizy osi częstotliwości, rysunek 3.2, system wykorzystuje uśrednioną prędkość maszyny. Prędkość ta jest obliczana jako średnia poszczególnych prędkości chwilowych z jednego obrotu na drugi. W tym przypadku średnia prędkość wynosi 2989 obr/min, co odpowiada:

$$\frac{2989}{60} [Hz] = 49,82 [Hz] \quad (2)$$

Ponieważ jest to prędkość wału odniesienia, prędkość wolnego wału wynosi:

$$49,82 [Hz] * 0,343 = 17,1 [Hz] \quad (3)$$

Ponieważ sygnał ma długość 1 sekundy, dokładność ta jest o rząd wielkości wyższa niż rozdzielczość widmowa (1 Hz), więc składowa asymetrii znajduje się na poziomie 17 Hz.



Rysunek 3.3: Przykładowa sygnatura wibracji przekładni wokół GMFx1 [Dzięki uprzejmości AMC Vibro®]

### 3.2.2 Łożyska toczne (REB)

Wadliwe łożysko (z defektem lokalnym) wzbudza strukturę ze współczynnikiem powtarzalności określonym przez charakterystyczną częstotliwość pojedynczego trybu uszkodzenia. Taki sygnał charakteryzuje się więc dwoma odrębnymi zjawiskami. Pierwszym zjawiskiem jest pojedynczy impuls, który odzwierciedla charakterystykę modalną obiektu. Tutaj zazwyczaj określany jest zakres częstotliwości strukturalnych, co oznacza, że fala nośna jest nietrywialnym, niejednorodnym, wielopasmowym szumem. Dla praktycznej analizy, powszechną obserwacją jest, że dla poszczególnych maszyn pasma widmowe indukowane przez wadliwe REB są znacząco różne.

W tym sensie pojedynczą wadliwą odpowiedź indukowaną REB można uznać za pojedyncze wzbudzenie struktury impulsowej. Drugie zjawisko dotyczy szybkości, z jaką te wzbudzenia mają miejsce. Ten wskaźnik jest zwykle nazywany „częstotliwością charakterystyczną” łożyska. Klasycznie, charakterystyczna częstotliwość bezpośrednio odzwierciedla powszechnie znane tryby awaryjne REB (BPFI, BPFO, FTF i BSF).

Równania do obliczania częstotliwości charakterystycznych (właściwie rzędów) łożyska tocznego są powszechnie dostępne [2.1] i [2.4]; dlatego tutaj podane są tylko wybrane

informacje. REB to element, który zawsze umieszczany jest na jakimś wale. Dlatego najpierw należy obliczyć względny rząd tego wału. Stanowi to dodatkowe wyzwanie na przykład dla REB-ów trzymających wał planety. Porównując łożyska ze stałym pierścieniem zewnętrznym (najpopularniejszy) i stałym pierścieniem wewnętrznym, wszystkie częstotliwości oprócz częstotliwości koszyka (zwanej również „Fundamental Train Frequency” - FTF) są obliczane w ten sam sposób. Częstotliwość wirowania elementu wirującego jest czasami nazywana BSF (od Ball Spin Frequency) lub 2xBSF. Dzieje się tak, ponieważ niektórzy naukowcy zakładają, że sygnał jest przekazywany do akcelerometru tylko przez bieżnię zewnętrzną REB, podczas gdy inni twierdzą, że skoro wadliwy element toczy uderza zarówno w bieżnię zewnętrzną, jak i wewnętrzną, częstotliwość ta jest podwojona.

Na koniec warto wspomnieć, że powszechnie znane równania do obliczania BPFO, BPFI, FTF i BSF wywodzą się z jednego ogólnego równania, które można również wykorzystać do obliczenia częstotliwości charakterystycznych REB przy obrocie obu bieżni.

### 3.3 Komponenty harmoniczne

#### 3.3.1 Dlaczego w widmie generowane są składowe harmoniczne?

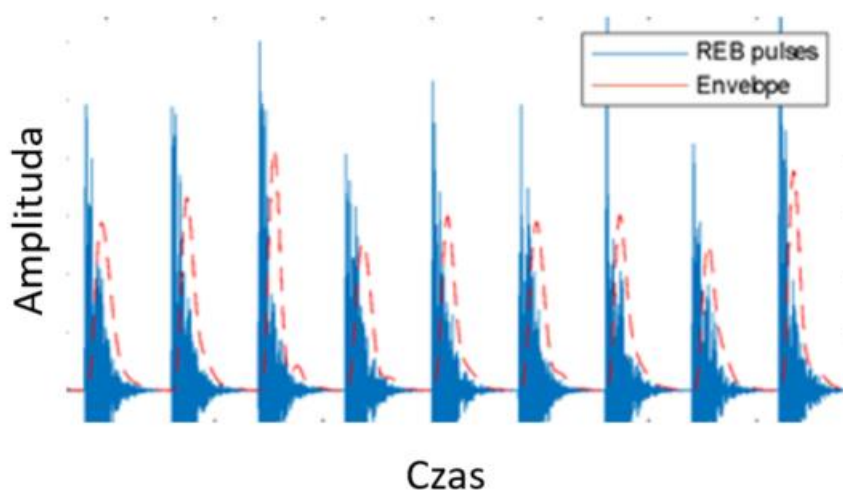
Aby odpowiedzieć na to pytanie, należy przywołać dwa podstawowe przykłady analizy Fouriera. Pierwszym przykładem jest transformata Fouriera czystej fali sinusoidalnej, która jest pojedynczą składową widmową, zwykle reprezentowaną w widmie jako pojedyncza linia pionowa. Drugim powszechnym przykładem rozpatrywanym w odwrotnej kolejności jest pozioma linia w widmie, która jest generowana jako transformata Fouriera funkcji delta Diraca. Można zatem zadać pytanie: co mogłoby generować wzór spektralny, który jest gdzieś „pomiędzy” pojedynczą składową deterministyczną widmową, a spektralną linią poziomą. Logicznie biorąc, biorąc pod uwagę powyższe, źródłowy sygnał w dziedzinie czasu musi być czymś okresowym „pomiędzy” deltą Diraca, a czystym sygnałem sinusoidalnym. Rzeczywiście, każdy sygnał, który jest deterministyczny i okresowy, ale nie jest zdefiniowany jako pojedyncza składowa sinusoidalna, generuje rodzinę składowych harmonicznych. Dla przykładu, jeśli przekładnia generuje GMF z harmonicznymi, oznacza to, że generowany sygnał jest okresowy, ale nie jest to składowa czysto sinusoidalna (tutaj ze względu na okresową modulację fazy wynikającą z ugięcia stykających się zębów). Dodatkowo, analizując komponenty sygnału drgań z przekładni należy rozpatrywać je w kategoriach systemów nieliniowych.

Ogólnie rzecz biorąc, obecność jakichkolwiek składowych harmonicznych nigdy nie jest pożądana z punktu widzenia monitorowania stanu. Jednak w przypadku niektórych maszyn,

takich jak maszyny tłokowe, takie elementy są nieuniknione, ponieważ są naturalnie wytwarzane przez zdrowe maszyny.

### 3.3.2 Dlaczego składowe harmoniczne są generowane w widmie obwiedni?

Pierwsza sekcja dotyczy spektralnych harmonicznych składowych deterministycznych. W tej części rozważane są składowe deterministyczne w widmie obwiedni. Kiedy REB doświadcza pojedynczego lokalnego uszkodzenia, wzbudza konstrukcję, tj. łożysko i jego obudowę) za każdym razem, gdy element toczny „wchodzi” w miejsce uszkodzenia. Zazwyczaj ta struktura ma wiele wąskopasmowych rezonansów strukturalnych mierzonych w kHz. Dlatego przebieg czasowy obwiedni impulsów jest okresowy, ale nie jest to pojedyncza składowa sinusoidalna, jak pokazano na rysunku 3.4.



Rysunek 3.4: Sygnał wysokiej częstotliwości REB z obwiednią

Zgodnie z teorią szeregów Fouriera, taki sygnał okresowy, jak również inne popularne kształty okresowe (jak fala kwadratowa czy trójkątna), wymagają wygenerowania szeregu harmonicznych. Dlatego z reguły wzrost amplitudy i liczby harmonicznych częstotliwości charakterystycznej REB jest oznaką rozwoju zwarcia REB.

### 3.4 Dodatkowe modulacje sygnałów

Modulacja opisuje zjawisko, w którym parametry jednego sygnału zmieniają się zgodnie z parametrami innego sygnału. Dla niektórych maszyn modulacje są naturalne, podczas gdy dla innych wskazują na usterkę maszyny. W analizie opartej na drganiach brane są pod uwagę zazwyczaj trzy typy modulacji, a mianowicie modulacja amplitudy (AM), modulacja częstotliwości (FM) i modulacja fazy (PM). Modulacje te mogą dotyczyć zarówno składowych deterministycznych, jak i losowych. Na przykład REB z uszkodzeniem bieżni

wewnętrznej (i obracającą się bieżnią wewnętrzną) generuje jednocześnie AM okresowo indukowanych losowych rezonansów strukturalnych i deterministycznego kształtu fali obwiedni z okresem równym prędkości wału, co skutkuje widmowymi pasmami bocznymi wokół BPFI. Podobnie uszkodzona przekładnia może indukować impulsy lub modyfikować GMF. W pierwszym przypadku częstotliwość powtarzalności byłaby obserwowalna tylko w widmie obwiedniowym, w drugim natomiast będzie dodatkowo widoczna na zwykłym widmie jako wstęgi boczne GMF.

Niektóre modulacje amplitudy, takie jak modulacje generowane przez planety EG przechodzące pod czujnikiem, są wyraźnie widoczne na surowym przebiegu czasowym; inne nie i wymagają dodatkowego przetwarzania w celu wykrycia. Jeśli chodzi o FM i PM to warto zdać sobie sprawę, że w wielu przypadkach mogą być one używane zamiennie, ponieważ FM odnosi się do prędkości kątowej, PM do kąta, a pochodna kąta (różnica względem czasu) wynosi prędkość kątowa, więc często jest to tylko kwestia notacji.

## 4 Wymagania funkcjonalne dla nowoczesnego systemu monitorowania

### 4.1 Metodologia badawcza

W niniejszym rozdziale przedstawiono metodologię, która została zastosowana w przeprowadzonych badaniach dotyczących analizy różnych aspektów mechatronicznych zaprojektowanych oraz badanych układów. Celem tego rozdziału jest opisanie kroków i narzędzi, które zostały wykorzystane do zbierania danych oraz analizy zebranych informacji. Metodologia ta odegrała kluczową rolę w realizacji celów badawczych i pozwoliła na uzyskanie wiarygodnych wyników.

W analizie mechatronicznych układów zaprojektowanych w kontekście systemów monitorowania i diagnostyki maszyn, istotne jest zastosowanie odpowiednich narzędzi i technik, które umożliwią kompleksową ocenę ich charakterystyk implementacyjnych. Jednym z takich narzędzi, które odgrywa szczególnie istotną rolę, są tablice morfologiczne.

Analiza morfologiczna jest metodą, która łączy elementy intuicji i analizy, tworząc kombinatoryczne narzędzie, które służy jako symulator wyobraźni. Metoda ta została opracowana przez Fritza Zwicky'ego (1898-1974), szwajcarskiego naukowca. Tablice morfologiczne stanowią strukturalne narzędzie, które umożliwia systematyczne analizowanie różnych aspektów badanych układów mechatronicznych. Są one szczególnie przydatne w przypadku złożonych systemów, gdzie analiza poszczególnych cech i parametrów może być skomplikowana. Tablice morfologiczne umożliwiają wizualizację i porównanie różnych wariantów rozwiązań, a także identyfikację istotnych zależności między poszczególnymi aspektami badanego systemu.

W kontekście systemów monitorowania i diagnostyki maszyn, zastosowanie tablic morfologicznych jest szczególnie istotne, ponieważ umożliwia analizę wielu ważnych aspektów takich jak wydajność, niezawodność, koszty implementacji, dostępność części zamiennych, warunki eksploatacyjne i wiele innych. Dzięki nim można dokonać kompleksowej oceny różnych możliwości projektowych i wybrać optymalne rozwiązanie z uwzględnieniem wielu czynników.

Metodologia prowadzonych badań opierała się na systematycznym zbieraniu danych dotyczących charakterystyk implementacyjnych mechatronicznych układów w systemach monitorowania i diagnostyki maszyn. Następnie, zastosowano tablice morfologiczne, które

umożliwiły analizę i porównanie różnych aspektów tych układów. Dzięki temu można było dokładnie ocenić ich potencjał, identyfikować mocne i słabe strony oraz wskazać najlepsze rozwiązania projektowe.

W rozdziale 4.2 przedstawione zostaną szczegóły dotyczące zebranych danych, opisane zostaną użyte narzędzia oraz przedstawione zostaną wyniki i wnioski z analizy tablic morfologicznych. Przy uwzględnieniu charakterystyki implementacyjnej w systemach CMS, zastosowanie tablic morfologicznych było niezwykle istotne dla przeprowadzenia wszechstronnej i dogłębnej analizy mechatronicznych układów zaprojektowanych w tym kontekście.

Celem rozdziałów 4.3 oraz 4.4 jest przedstawienie procesu wykonania prac związanych z pozyskaniem układów developerskich dla wybranych platform wbudowanych oraz przeprowadzeniem testów i badań dotyczących ich wydajności, zużycia energii oraz właściwości protokołów komunikacyjnych.

W ramach realizacji badań dotyczących platform wbudowanych, pierwszym etapem było pozyskanie odpowiednich układów deweloperskich. Wybór konkretnych platform był oparty na analizie wymagań projektowych oraz potrzeb związanych z monitorowaniem i diagnostyką maszyn. Układy developerskie stanowiły narzędzia umożliwiające rozwijanie i testowanie aplikacji w środowisku wbudowanym. Ich odpowiedni dobór był kluczowy dla późniejszych etapów badań. Kolejnym etapem było przeprowadzenie testów wydajnościowych układów obliczeniowych oraz pamięciowych. Testy te miały na celu ocenę możliwości i ograniczeń poszczególnych platform w zakresie przetwarzania danych oraz wydajności obliczeniowej. W ramach tych testów analizowano zarówno aspekty czysto obliczeniowe, jak i zarządzanie pamięcią. Ponadto, uwzględniano również zużycie energii elektrycznej, jako istotny czynnik w kontekście wbudowanych systemów zasilanych bateryjnie. Kolejnym istotnym obszarem badań było badanie właściwości protokołów komunikacyjnych. Skupiano się na ocenie zużycia energii oraz możliwości transferu danych w kontekście komunikacji między różnymi komponentami systemów monitorowania i diagnostyki maszyn. Protokoły komunikacyjne odgrywają kluczową rolę w skutecznym przesyłaniu informacji wewnątrz systemów mechatronicznych. Przeprowadzenie odpowiednich badań miało na celu identyfikację protokołów optymalnych pod względem efektywnego wykorzystania energii oraz przepustowości transmisji danych.

## 4.2 Tablice morfologiczne poszczególnych części składowych systemu

Generowanie alternatywnych rozwiązań projektowych dla produktów z wieloma komponentami może nastroczać trudności, zwłaszcza gdy oczekuje się, że komponenty będą spełniać skoordynowane funkcje w projektowanym systemie techniczno-fizycznym. Ściśle powiązane podfunkcje produktu mogą prowadzić do problemów z eksploracją potencjału projektu. Literatura oferuje wiele metod generowania pomysłów służących do poszerzenia przestrzeni rozwiązań dla inżynierii i projektowania produktów, a jedną z nich jest metoda wykresów morfologicznych.

Metoda wykresu morfologicznego opiera się na metodzie Ogólnej Analizy Morfologicznej (GMA - General Morphological Analysis) opracowanej przez Fritza Zwicky'ego w celu badania niekwantyfikowalnych kompleksów problemów [4.1 - 4.2]. Metoda ta jest wykorzystywana do podziału problemu na pod-funkcje (dekompozycja problemu), generowania dla nich licznych pod-rozwiązań (rozbieżność projektu) oraz wyboru i łączenia odpowiednich pod-rozwiązań w alternatywne rozwiązania ogólne (zbieżność projektu) [4.2 - 4.7]. Metoda ta jest zalecana do stosowania szczególnie na etapach opracowywania pomysłów, gdzie przestrzeń rozwiązań jest rozszerzana w poszukiwaniu wszystkich możliwych środków, które mogą działać jako rozwiązanie dla podfunkcji; obejmuje to poszukiwanie formy, jak sugeruje nazwa (morph-). Richardson, Summers i Mocko podsumowują korzyści płynące z metody wykresu morfologicznego jako: powiększenie przestrzeni projektowej do zbadania, generowanie nowych koncepcji, które w przeciwnym razie nie byłyby brane pod uwagę, oraz reprezentowanie szerokiego zakresu koncepcji umożliwiających rozważenie nieoczekiwanego dopasowania komponentów [4.8].

Poszerzenie przestrzeni rozwiązań polega na określeniu granic w obrębie, których będzie odbywać się eksploracja projektu, tak aby można było rozważyć kilka różnych możliwości. Obejmuje to dywergencję projektową, która polega na generowaniu szerokiej gamy alternatywnych pomysłów, które przyczynią się do tej eksploracji. Im bardziej świadomy będzie ten proces, tym dokładniej zostaną wyznaczone granice przestrzeni rozwiązań i tym lepiej ugruntowana będzie ta eksploracja. Dogłębna analiza problemu jest zatem ważna w przygotowaniu do tej eksploracji.

Tablice morfologiczne są narzędziem, które pomagają w systematycznym zbieraniu, analizowaniu i organizowaniu informacji na temat różnych elementów składowych lub aspektów danego problemu lub systemu. Są one wykorzystywane w procesie planowania

nowych urządzeń mechatronicznych w celu zdefiniowania i zrozumienia różnych możliwości oraz kombinacji, które można uwzględnić podczas projektowania.

Tablica morfologiczna składa się z dwóch lub więcej kolumn, z których każda reprezentuje inny aspekt lub element składowy problemu. W każdej kolumnie zawarte są różne opcje lub atrybuty dotyczące danego aspektu. Następnie, poprzez łączenie różnych opcji z różnych kolumn, można utworzyć kombinacje, które potencjalnie mogą prowadzić do rozwiązania problemu.

*Tabela 4.1: Przykład tabeli morfologicznej*

<b>Moduł mechaniczny</b>	<b>Moduł elektroniczny</b>	<b>Moduł oprogramowania</b>
Napęd	Procesor	Algorytmy sterowania
Rama	Sensory	Interfejs użytkownika
Przekładnia	Aktuatory	Komunikacja

Tabela 4.1 prezentuje przykład tablicy morfologicznej dla procesu planowania nowego urządzenia mechatronicznego.

W tym przykładzie, trzy różne aspekty modułów składowych urządzenia mechatronicznego zostały podzielone na trzy kolumny: moduł mechaniczny, moduł elektroniczny i moduł oprogramowania. Każda kolumna zawiera różne opcje lub elementy składowe, które można uwzględnić w odpowiednich modułach.

Stosowanie tablic morfologicznych w procesie planowania nowych urządzeń mechatronicznych ma kilka korzyści:

- 1. Systematyczne zbieranie informacji:** Tablice morfologiczne pomagają w zbieraniu i organizowaniu istotnych informacji dotyczących różnych aspektów projektowanego urządzenia. Umożliwiają one pełne uwzględnienie różnych elementów składowych i ich kombinacji.
- 2. Generowanie pomysłów:** Poprzez połączenie różnych opcji z różnych kolumn, tablice morfologiczne pozwalają na generowanie nowych pomysłów i kreatywne podejście do projektowania urządzenia mechatronicznego.
- 3. Wybór optymalnych rozwiązań:** Analiza kombinacji w tablicy morfologicznej umożliwia identyfikację najlepszych i najbardziej optymalnych rozwiązań dla poszczególnych modułów urządzenia mechatronicznego.

4. **Zapobieganie pominięciu elementów:** Dzięki strukturalnemu podejściu, tablice morfologiczne pomagają zapobiec pom

Przed przystąpieniem do prac nad zaawansowanym systemem akwizycji należy wziąć pod uwagę najważniejsze aspekty wpływające na jego funkcjonalność. System rozważany w tej rozprawie powinien być bardzo wydajny, tani i niezawodny; powinien spełniać podstawową funkcjonalność profesjonalnych systemów CMS. Na początku procesu tworzenia oraz rozwoju nowego produktu należy zadać sobie wiele kluczowych pytań, z którymi trzeba się następnie zmierzyć. Pytania te są związane z odpowiednim doбором zasilania całego systemu, odnalezieniem układu ADC spełniającego wymagania przetwarzania sygnału cyfrowego, ze znalezieniem procesora o właściwych parametrach pozwalających w kolejnych krokach na implementację niezbędnej funkcjonalności, należy również przewidzieć co w przyszłości może się okazać kluczowe dla całego systemu – np. dostępność dodatkowych funkcji jak slot na kartę SD, połączenie WiFi, moduł Bluetooth czy port Ethernet itp.



Rysunek 4.1: Przykłady architektury urządzenia wbudowanego, mającej kluczowy wpływ na przyszłą funkcjonalność CMS

Rysunek 4.1 przedstawia kilka przykładów rozważanej architektury mającej kluczowy wpływ na możliwości przetwornika, a w tabeli 4.2 przedstawiono uproszczoną tablicę morfologiczną, pozwalającą poznać najistotniejsze elementy systemu, która była podstawą do dalszej analizy oraz sformułowania wniosków własnych.

*Tabela 4.2: Tablica morfologiczna przedstawiająca wszystkie najważniejsze elementy składowe rozpatrywanego CMS*

Zasilanie	Układ ADC	Miejsce na dane	Komunikacja wew.	Komunikacja zew.	Moc obliczeniowa
230V	Wew.	Karta SD	Obsługa Wejść/Wyjść	Wi-Fi	Częstotliwość procesora
Baterie AA/AAA	Zew.	Pamięć RAM	Interfejs UART	Ethernet	Ilość rdzeni
Power-Bank	-	Wew. flash	Interfejs SPI	Bluetooth	-
Bateria wbudowana	-	Baza Online	Interfejs I2C	-	-
Port USB	-	-	-	-	-

Tabela 4.2 przedstawia różne opcje dotyczące zasilania, układu ADC, miejsca na dane, komunikacji wewnętrznej, komunikacji zewnętrznej oraz mocy obliczeniowej w kontekście planowania urządzenia mechatronicznego. Oto opis poszczególnych kolumn:

- **Zasilanie:** Ta kolumna przedstawia różne opcje zasilania urządzenia. Możliwe opcje to zasilanie sieciowe 230V, baterie AA/AAA, power-bank, bateria wbudowana oraz port USB.
- **Układ ADC:** Kolumna ta odnosi się do układu konwersji analogowo-cyfrowej (ADC). Możliwe opcje to układ ADC wewnętrzny (wbudowany) lub zewnętrzny.
- **Miejsce na dane:** Ta kolumna wskazuje, gdzie są przechowywane dane w urządzeniu. Opcje obejmują kartę SD, pamięć RAM, wewnętrzną pamięć flash oraz bazę danych online.
- **Komunikacja wewnętrzna:** Kolumna ta odnosi się do sposobu komunikacji między różnymi częściami urządzenia. Możliwe opcje to obsługa wejść/wyjść, interfejs UART, interfejs SPI oraz interfejs I2C.
- **Komunikacja zewnętrzna:** Ta kolumna odnosi się do sposobu komunikacji urządzenia z innymi urządzeniami lub sieciami. Możliwe opcje to Wi-Fi, Ethernet oraz Bluetooth.
- **Moc obliczeniowa:** Kolumna ta wskazuje moc obliczeniową urządzenia. Opcje to częstotliwość procesora, ilość rdzeni.

Tabela umożliwia dokładne zdefiniowanie różnych opcji związanych z zasilaniem, komunikacją i innymi aspektami projektowanego urządzenia mechatronicznego. Może być wykorzystana do analizy i wyboru optymalnych kombinacji dla poszczególnych elementów składowych urządzenia.

#### 4.2.1 Układy zasilania

Każda żywa istota, niezależnie od tego, czy mówimy o roślinach, czy o zwierzętach, potrzebuje pożywienia i wody, aby zaopatrywać się w energię, aby wykonać jakąkolwiek pracę. Analogicznie, tak jak każda nieożywiona rzecz potrzebuje zewnętrznego impulsu, aby działała, tak każde urządzenie cyfrowe potrzebuje źródła zasilania, aby wykonało pracę, której potrzebujemy.

Układy zasilania w urządzeniach mechatronicznych są odpowiedzialne za dostarczanie energii elektrycznej potrzebnej do prawidłowego działania urządzenia. Ich rola jest niezwykle istotna, ponieważ bez odpowiedniego zasilania urządzenie nie będzie w stanie funkcjonować poprawnie lub w ogóle. W systemach elektronicznych źródło zasilania ma postać zasilacza (lub baterii), który zasila wymagane urządzenie odpowiednią formą energii elektrycznej, która jest wymagana do wykonania pracy. Układy zasilania zapewniają przekształcenie i dostosowanie energii elektrycznej z zewnętrznego źródła zasilania (np. sieci elektrycznej, baterii, power banku) do odpowiedniego poziomu napięcia, prądu i częstotliwości, które są wymagane przez poszczególne komponenty mechatronicznego systemu.

Możliwe typy zastosowanych źródeł zasilania w urządzeniach mechatronicznych obejmują:

- Zasilanie sieciowe (np. 230V): W przypadku urządzeń podłączanych do sieci elektrycznej, układ zasilania przetwarza napięcie i prąd dostarczane z sieci na wartości zgodne z wymaganiami urządzenia. Ten typ zasilania jest popularny w domowych i przemysłowych aplikacjach mechatronicznych.
- Baterie: Baterie są powszechnie stosowanym źródłem zasilania w urządzeniach przenośnych, które wymagają mobilności. Mogą to być standardowe baterie alkaliczne (np. AA, AAA) lub specjalistyczne baterie, takie jak litowo-jonowe (Li-Ion) czy litowo-polimerowe (Li-Po).

- Power Bank: Power banki są przenośnymi urządzeniami, które przechowują energię elektryczną i umożliwiają jej dostarczanie do innych urządzeń. Są one szczególnie przydatne w sytuacjach, gdy brakuje dostępu do tradycyjnych źródeł zasilania, a urządzenie mechatroniczne wymaga mobilnego zasilania.
- Baterie wbudowane: Niektóre urządzenia mechatroniczne posiadają wbudowane baterie, które są ładowane poprzez zasilanie sieciowe lub inne źródła zasilania. To rozwiązanie umożliwia przenośność i niezależność od stałego podłączenia do sieci elektrycznej.
- Port USB: W niektórych przypadkach, urządzenia mechatroniczne mogą być zasilane za pomocą portu USB, który dostarcza energię elektryczną z innych urządzeń, takich jak komputery lub ładowarki.

Wybór odpowiedniego źródła zasilania zależy od konkretnych wymagań projektowanego urządzenia mechatronicznego, takich jak mobilność, czas pracy na baterii, dostępność zasilania sieciowego i inne czynniki specyficzne dla danej aplikacji.

*Tabela 4.3: Analiza dostępnych źródeł zasilania*

<b>Źródło zasilania</b>	<b>Rodzaj</b>	<b>Pojemność</b>	<b>Możliwość ładowania</b>
Sieciowe	Przemienny	Wysoka	Brak
Baterie	Stały	Niska	Uzależniona od typu
Power-Bank	Stały	Średnia	Uzależniona od typu
Bateria wbudowana	Stały	Niska	Uzależniona od typu
Port USB	Stały	Niska	Ograniczona

W tablicy morfologicznej 4.3 przedstawione są cztery różne źródła zasilania: zasilanie sieciowe, baterie, power banki oraz baterie wbudowane. Dodatkowo, wyróżniono również port USB jako potencjalne źródło zasilania. Każda kolumna reprezentuje inny atrybut lub element składowy układu zasilania:

- Źródło zasilania: W tej kolumnie wskazane są różne źródła zasilania, takie jak zasilanie sieciowe, baterie, power banki, baterie wbudowane oraz port USB.
- Rodzaj: Kolumna ta określa rodzaj prądu dostarczanego przez dane źródło zasilania, czy jest to prąd stały (DC) lub prąd przemienny (AC).
- Pojemność: Ta kolumna przedstawia pojemność energii dostarczanej przez dane źródło zasilania, która może być wysoka, średnia lub niska.

- Ładowność: W tej kolumnie wskazane są informacje dotyczące możliwości ładowania danego źródła zasilania, czy jest to brak możliwości ładowania, zależność od typu baterii, pojemności power banku lub ograniczenie w przypadku portu USB.

Analiza powyższej tabeli morfologicznej układów zasilania w urządzeniach mechatronicznych może wyglądać następująco:

- Zasilanie sieciowe:
  - Prąd przemienny: Podobnie jak prąd stały, zapewnia wysoką pojemność energii, ale w przypadku tego rodzaju prądu może być konieczne zastosowanie dodatkowego przetwornika AC/DC w celu przekształcenia go na prąd stały.
- Baterie:
  - Prąd stały: Zapewnia niską pojemność energii, co może być ograniczeniem czasu pracy urządzenia. Wybór baterii zależy od konkretnego typu, np. baterie alkaliczne (AA, AAA), litowo-jonowe (Li-Ion) lub litowo-polimerowe (Li-Po). Wymagają ładowania lub wymiany po wyczerpaniu się energii.
- Power Bank:
  - Prąd stały: Oferuje średnią pojemność energii i jest przenośnym źródłem zasilania, które można ładować przed użyciem. Jest przydatny w sytuacjach, gdy brakuje dostępu do zasilania sieciowego lub baterii.
- Baterie wbudowane:
  - Prąd stały: Zapewnia niską pojemność energii, a baterie wbudowane są zazwyczaj ładowane przez zasilanie sieciowe lub inne źródła zasilania. Mogą być stosowane w urządzeniach wymagających mobilności i niezależności od stałego podłączenia do zasilania sieciowego.
- Port USB:
  - Prąd stały: Oferuje niską pojemność energii i zasilanie poprzez port USB jest zazwyczaj ograniczone. Jest wykorzystywany głównie w celu ładowania urządzeń niskiego poboru mocy lub dostarczania niewielkiej ilości energii.

Najlepszym źródłem zasilania dla mobilnych czujników bezprzewodowych są baterie i power banki. Baterie są idealne ze względu na ich przenośność i możliwość bezprzewodowego działania, co jest istotne dla czujników, które muszą być łatwo przenoszone i komunikować się z innymi urządzeniami. Dodatkowo, łatwość wymiany baterii pozwala na ciągłe działanie urządzenia bez potrzeby podłączania go do zasilania sieciowego. Z drugiej strony, power banki oferują przenośne ładowanie, co oznacza, że czujniki mogą być ładowane w terenie, nawet w sytuacjach, gdy brak dostępu do zasilania sieciowego. W przypadku mobilnych czujników bezprzewodowych, te dwa źródła zasilania są najbardziej popularne i odpowiednie do zapewnienia niezależności od stałego zasilania i umożliwienia swobodnego działania w różnych miejscach.

#### 4.2.2 Przetworniki analogowo-cyfrowe

W świecie rzeczywistym, sygnały analogowe mają ciągle zmieniające się wartości, które pochodzą z różnych źródeł i mogą one być ciągłe i dostarczać nieskończenie wiele różnych wartości. Natomiast obwody cyfrowe pracują z sygnałem binarnym, który ma tylko dwa stany dyskretne, logiczne „1” (WYSOKI) lub logiczne „0” (NISKI). Z tego względu konieczne jest posiadanie specjalnego obwodu elektronicznego, który może dokonywać konwersji pomiędzy tymi dwiema różnymi domenami ciągle zmieniających się sygnałów analogowych i dyskretnych sygnałów cyfrowych. I to właśnie tutaj pojawiają się przetworniki analogowo-cyfrowe [4.9 – 4.11].

Przetworniki analogowo-cyfrowe (ADC) są istotnymi komponentami w nowoczesnej elektronice, które mają za zadanie przekształcać analogowe sygnały na formę cyfrową. Służą one konwersji ciągłych sygnałów analogowych, takich jak napięcie czy prąd, na równoważne wartości cyfrowe, które można przetwarzać za pomocą cyfrowych układów i algorytmów. ADC odgrywają kluczową rolę w różnych dziedzinach elektroniki, takich jak telekomunikacja, przemysł, medycyna czy automatyka. W telekomunikacji są wykorzystywane do konwersji sygnałów audio czy wideo na format cyfrowy, co umożliwia ich przesyłanie i przetwarzanie w cyfrowej postaci. W przemyśle ADC są używane do pomiaru i kontroli różnych parametrów, takich jak temperatura, ciśnienie czy poziom sygnałów elektrycznych. Kontekst historyczny przetworników analogowo-cyfrowych sięga początków rozwoju elektroniki. Wraz z postępem technologicznym i rosnącymi potrzebami przetwarzania danych, powstała potrzeba konwersji sygnałów analogowych na cyfrowe. Pierwsze próby w tym zakresie miały miejsce w latach 50. i 60. XX wieku, gdzie stosowano

proste układy konwersji. Jednak rozwój technologii mikroelektronicznych i cyfrowych umożliwił powstanie bardziej zaawansowanych i precyzyjnych ADC.

Wraz z upowszechnieniem się komputerów i cyfrowych systemów przetwarzania danych, ADC stały się nieodzownym elementem wielu aplikacji. Ich rola w elektronice jest niezwykle istotna, umożliwiając cyfrową obróbkę i analizę sygnałów, a także integrację z innymi cyfrowymi układami. Wraz z postępowaniem technologii, rozwijane są coraz bardziej zaawansowane i wydajne przetworniki analogowo-cyfrowe, które pozwalają na precyzyjne i szybkie konwersje sygnałów analogowych w szerokim zakresie aplikacji elektronicznych.

*Tabela 4.4: Analiza dostępnych przetworników analogowo-cyfrowych*

Typ przetwornika ADC	
<b>Przetwornik wbudowany</b>	Nie trzeba inwestować pieniędzy w zakup zewnętrznego ADC, znacznie mniejszy rozmiar płytki, łatwiejsze ustawienie odpowiednich parametrów adc, łatwiejsza synchronizacja, wsparcie.
<b>Przetwornik Zewnętrzny</b>	Konieczność zainwestowania pieniędzy w wybór odpowiedniego przetwornika, możliwość wyboru przetwornika o wyższej rozdzielczości

Tabela 4.4 przedstawia różnice między przetwornikiem ADC wbudowanym a zewnętrznym. Przetwornik wbudowany ma wiele korzyści, takich jak brak potrzeby inwestowania dodatkowych środków finansowych w zakup zewnętrznego ADC. Dodatkowo, przetwornik wbudowany jest znacznie mniejszy pod względem rozmiaru płytki, co może być istotne w przypadku projektów, gdzie przestrzeń jest ograniczona. Ustawienie odpowiednich parametrów ADC jest również łatwiejsze, a synchronizacja z innymi komponentami układu jest bardziej intuicyjna. Przetwornik wbudowany może również oferować wsparcie ze strony producenta, co może ułatwić proces projektowania i implementacji.

Z drugiej strony, wybór przetwornika ADC zewnętrznego wiąże się z koniecznością inwestycji finansowej w celu zakupu odpowiedniego przetwornika. Jednak taka opcja daje możliwość wyboru przetwornika o wyższej rozdzielczości, co może być istotne w aplikacjach, gdzie precyzja jest kluczowa. Przetworniki ADC zewnętrzne często oferują większą elastyczność w dostosowaniu do konkretnych wymagań projektowych.

Ostateczny wybór między przetwornikiem ADC wbudowanym a zewnętrznym zależy od konkretnej sytuacji i potrzeb projektowych. Każdy z tych typów ma swoje zalety i ograniczenia, dlatego należy uwzględnić takie czynniki jak koszt, rozmiar, precyzję i dostosowanie do konkretnych wymagań, aby podjąć właściwą decyzję.

### 4.2.3 Moduły przechowywania danych

Moduł lub komponent do przechowywania danych cyfrowych w nowoczesnej elektronice to urządzenie lub układ, który umożliwia gromadzenie i przechowywanie informacji w postaci cyfrowej. Służy on do przechowywania danych, takich jak pliki, programy, ustawienia czy dane użytkownika, w sposób trwały i nieulotny. Kontekst historyczny modułów do przechowywania danych sięga początków rozwoju technologii komputerowych. Pierwsze próby przechowywania danych cyfrowych miały miejsce w latach 40. i 50. XX wieku, gdzie stosowano m.in. perforowane karty lub taśmy magnetyczne. Jednak rozwój elektroniki i mikroprocesorów spowodował pojawienie się bardziej zaawansowanych i kompaktowych rozwiązań. Wraz z postępowaniem technologicznym, pojawiły się różne rodzaje modułów do przechowywania danych cyfrowych. Przykłady to:

- Dyski twarde (HDD): Wykorzystują magnetyczne dyski do przechowywania danych, oferując duże pojemności i stosunkowo niskie koszty.
- Dyski SSD: Opierają się na pamięciach flash, zapewniając wysoką prędkość odczytu i zapisu danych oraz większą odporność na wstrząsy, przy mniejszym poborze mocy.
- Karty pamięci: Niewielkie, przenośne karty, takie jak karty SD lub microSD, które są powszechnie używane w urządzeniach mobilnych i aparatach fotograficznych.
- Pamięć flash: Szeroko stosowana pamięć nieulotna, oferująca szybki dostęp do danych i trwałe przechowywanie.

Moduły do przechowywania danych są nieodzownymi elementami nowoczesnej elektroniki, umożliwiając przechowywanie i dostęp do informacji w sposób wygodny i efektywny. Dzięki nim użytkownicy mogą przechowywać i przenosić dane na różnych urządzeniach, a także zabezpieczać je przed utratą czy uszkodzeniem. Wraz z postępowaniem technologicznym, moduły do przechowywania danych stają się coraz bardziej pojemne, szybsze i niezawodne, spełniając rosnące potrzeby przechowywania i dostępu do danych w dzisiejszym świecie cyfrowym.

Tabela 4.5: Analiza możliwości przechowywania danych pomiarowych

Przechowywanie danych pomiarowych	
Karta SD/Pendrive (zew. Pamięć flash)	Możliwość przechowywania dużej ilości danych
Pamięć RAM	Więcej danych może być przechowywanych wewnątrz mikrokontrolera, bez zewnętrznych modułów
Wbudowana pamięć flash	Potrzeba zainwestowania większej ilości pieniędzy, aby wybrać lepszą płytkę, można wyeliminować dodatkową zewnętrzną pamięć flash
Baza danych online	Przechowywanie danych w chmurze, wymaga połączenia z Internetem, więc potrzebne jest Wi-Fi lub Ethernet

Tabela 4.5 przedstawia różne metody przechowywania danych pomiarowych. Karta SD, pendrive oraz zewnętrzna pamięć flash to popularne rozwiązania, które umożliwiają przechowywanie dużej ilości danych. Są one wygodne w użyciu, ponieważ można je łatwo podłączyć do urządzenia i przenieść dane na inne systemy. Pamięć RAM jest często stosowana wewnątrz mikrokontrolerów, umożliwiając przechowywanie większej ilości danych bez konieczności korzystania z zewnętrznych nośników. Jest to korzystne, zwłaszcza w przypadku ograniczonej przestrzeni lub gdy konieczne jest szybki dostęp do danych. Wbudowana pamięć flash jest dostępna na niektórych płytach elektronicznych. Może to wyeliminować konieczność zakupu dodatkowego nośnika flash, ale może wiązać się z wyższym kosztem zakupu lepszej płytki. Baza danych online umożliwia przechowywanie danych w chmurze. Jest to wygodna metoda, ponieważ dane są dostępne z dowolnego miejsca, pod warunkiem posiadania połączenia internetowego. Wymaga to zastosowania technologii takich jak Wi-Fi lub Ethernet, aby umożliwić przesyłanie danych do i z bazy danych online.

Wybór metody przechowywania danych pomiarowych zależy od konkretnych wymagań projektowych i preferencji użytkownika. Należy wziąć pod uwagę czynniki takie jak pojemność, koszt, dostępność, łatwość obsługi i wymagania dotyczące dostępu do danych w celu podjęcia optymalnej decyzji.

#### **4.2.4** **Możliwości komunikacyjne – wewnętrzne**

Wewnętrzne protokoły komunikacyjne, takie jak UART, SPI, I2C oraz obsługa wejść/wyjść, odgrywają istotną rolę w nowoczesnej elektronice. Służą one do przesyłania danych między różnymi komponentami układu elektronicznego lub mikrokontrolerem. Kontekst historyczny tych protokołów sięga lat 70. i 80. XX wieku, kiedy to zostały opracowane i wprowadzone do użytku. UART był jednym z pierwszych protokołów komunikacyjnych wykorzystywanych w komputerach osobistych, umożliwiając transmisję danych między komputerem a zewnętrznymi urządzeniami. SPI i I2C zostały wprowadzone jako protokoły komunikacyjne dla urządzeń peryferyjnych, umożliwiając podłączanie ich do mikrokontrolerów w sposób efektywny i niezawodny.

UART (Universal Asynchronous Receiver-Transmitter) to interfejs asynchronicznej transmisji szeregowej, który umożliwia komunikację między dwoma urządzeniami. Jest

powszechnie wykorzystywany do komunikacji między mikrokontrolerem a innymi urządzeniami, takimi jak moduły Bluetooth czy konwertery USB-serial.

SPI (Serial Peripheral Interface) jest protokołem synchronicznej transmisji szeregowej, który pozwala na komunikację między mikrokontrolerem a innymi urządzeniami peryferyjnymi, takimi jak czujniki, wyświetlacze czy pamięci. Wykorzystuje on interfejs zegarowy i linie danych do przesyłania informacji.

I2C (Inter-Integrated Circuit) to interfejs komunikacyjny, który umożliwia komunikację między urządzeniami poprzez dwie linie: linię danych (SDA) i linię zegara (SCL). Jest wykorzystywany do podłączania różnych urządzeń, takich jak czujniki, wyświetlacze, pamięci EEPROM czy układy zarządzania zasilaniem.

Obsługa wejść/wyjść odnosi się do interakcji między mikrokontrolerem a zewnętrznymi urządzeniami poprzez różne porty wejścia/wyjścia. Mikrokontroler może odczytywać dane z zewnętrznych czujników, przekazywać sygnały sterujące do elementów wykonawczych lub komunikować się z innymi układami za pomocą tych portów.

Współcześnie te wewnętrzne protokoły komunikacyjne są szeroko stosowane w nowoczesnej elektronice, umożliwiając interakcję i wymianę danych między różnymi komponentami układu elektronicznego. Oferują one efektywną i niezawodną komunikację, umożliwiając integrację różnorodnych urządzeń i rozszerzanie ich funkcjonalności.

*Tabela 4.6: Analiza możliwości komunikacji wewnętrznej pomiędzy elementami przetwornika*

<b>Możliwości komunikacji pomiędzy elementami przetwornika</b>	
<b>Obsługa Wejść/Wyjść</b>	Im większa liczba GPIO, tym więcej dodatkowych urządzeń można podłączyć do płyty
<b>Interfejs UART</b>	Możliwość podłączenia konwertera RS232, jeśli nie jest zapewniony, możliwość podłączenia modułu Bluetooth do bezprzewodowego sterowania systemem akwizycji
<b>Interfejs SPI</b>	Możliwość podłączenia zewnętrznego przetwornika ADC, pamięci FLASH, zewnętrznego modułu RTC, wyświetlacza LCD
<b>Interfejs I2C</b>	Możliwość podłączenia zewnętrznego przetwornika ADC, pamięci FLASH, zewnętrznego modułu RTC, wyświetlacza LCD

Tabela 4.6 przedstawia różne możliwości komunikacji pomiędzy poszczególnymi elementami przetwornika będącego elementem omawianego systemu. Obsługa wejść/wyjść (GPIO) odnosi się do liczby ogólnych portów wejścia/wyjścia dostępnych na płycie. Im większa liczba GPIO, tym więcej dodatkowych urządzeń można podłączyć do płyty, co zwiększa elastyczność i funkcjonalność systemu. Interfejs UART umożliwia komunikację

szeregową między przetwornikiem a innymi urządzeniami. Jeśli przetwornik nie zapewnia bezpośredniego wsparcia dla interfejsu RS232, można podłączyć konwerter RS232 w celu obsługi urządzeń zgodnych z tym standardem. Dodatkowo, możliwe jest również podłączenie modułu Bluetooth, co umożliwia bezprzewodowe sterowanie systemem akwizycji danych. Interfejs SPI (Serial Peripheral Interface) umożliwia komunikację między przetwornikiem a zewnętrznymi urządzeniami peryferyjnymi. Dzięki temu można podłączyć zewnętrzne przetworniki ADC (Analog-to-Digital Converter), pamięci flash, moduły RTC (Real-Time Clock) oraz wyświetlacze LCD, co rozszerza możliwości przetwarzania, przechowywania i wizualizacji danych. Interfejs I2C (Inter-Integrated Circuit) również umożliwia komunikację z zewnętrznymi urządzeniami. Podobnie jak w przypadku protokołu SPI, można podłączyć zewnętrzne przetworniki ADC, pamięci flash, moduły RTC oraz wyświetlacze LCD, co umożliwia większą elastyczność w projektowaniu systemu.

Dzięki tym różnym protokołom komunikacyjnym możliwe jest rozszerzenie funkcjonalności przetwornika poprzez podłączanie różnych urządzeń peryferyjnych. Wybór odpowiedniego protokołu zależy od konkretnych wymagań projektowych i dostępności zewnętrznych urządzeń, które mają być zintegrowane z przetwornikiem.

#### 4.2.5 Transmisja danych

Zewnętrzne protokoły komunikacyjne, takie jak Wi-Fi, Ethernet i Bluetooth, odgrywają istotną rolę w nowoczesnej elektronice, umożliwiając bezprzewodową transmisję danych między urządzeniami. Kontekst historyczny tych protokołów jest związany z rozwojem technologii komunikacyjnych. Wi-Fi zostało wprowadzone na szeroką skalę w latach 90. jako standard bezprzewodowej komunikacji, a obecnie jest powszechnie stosowane w wielu urządzeniach elektronicznych. Ethernet, z kolei, pojawił się już w latach 70. jako interfejs do komunikacji w sieciach lokalnych. Bluetooth zostało wprowadzone w latach 90. jako technologia umożliwiająca łatwe połączenia między urządzeniami mobilnymi.

Wi-Fi jest popularnym protokołem, który umożliwia bezprzewodowe połączenie z siecią internetową, co pozwala na przesyłanie danych, dostęp do zasobów sieciowych i interakcję z aplikacjami online.

Ethernet jest protokołem przewodowym wykorzystywanym do budowania lokalnych sieci komputerowych. Pozwala na szybką i niezawodną transmisję danych w sieciach przewodowych, umożliwiając komunikację między różnymi urządzeniami elektronicznymi, takimi jak komputery, drukarki, routery i inne.

Bluetooth to bezprzewodowy interfejs komunikacyjny, który umożliwia niskonapięciową transmisję danych na krótkim dystansie. Jest szeroko stosowany w urządzeniach mobilnych, takich jak smartfony, tablety, słuchawki bezprzewodowe, głośniki, a także w systemach samochodowych. Bluetooth umożliwia łatwe i wygodne połączenie między różnymi urządzeniami, co pozwala na przesyłanie danych, odtwarzanie muzyki, prowadzenie rozmów telefonicznych i wiele innych zastosowań.

Dzięki tym zewnętrznym protokołom komunikacyjnym, urządzenia elektroniczne mogą być połączone ze sobą, tworząc sieci i umożliwiając przesyłanie danych, udostępnianie zasobów, sterowanie i interakcję. Ich rola jest kluczowa w dzisiejszym świecie, gdzie coraz więcej urządzeń jest wzajemnie połączonych i współpracuje w ramach inteligentnych systemów.

*Tabela 4.7: Analiza możliwości transmisji danych pomiędzy przetwornikiem, a urządzeniem odbiorczym*

<b>Komunikacja przetwornika z urządzeniem odbiorczym</b>	
<b>Wi-Fi</b>	Bezprzewodowa komunikacja z urządzeniem przez Internet, może wysyłać dane bezpośrednio do chmury,
<b>Ethernet</b>	Przewodowa komunikacja z urządzeniem przez Internet, możliwość wysyłania danych bezpośrednio do chmury, konieczność podłączenia urządzenia do sieci za pomocą złącza RJ45
<b>Bluetooth</b>	Komunikacja bezprzewodowa do około 10 metrów, ograniczony transfer danych w porównaniu do powyższych

Komunikacja przetwornika z urządzeniem odbiorczym może odbywać się za pomocą różnych protokołów, z których trzy ważne są wymienione w tabeli 4.7. Wi-Fi umożliwia bezprzewodową komunikację z urządzeniem przez Internet. Przetwornik może wysyłać dane bezpośrednio do chmury, co umożliwia łatwy dostęp do danych z dowolnego miejsca. Jest to szczególnie przydatne w przypadku monitorowania zdalnego, zdalnej kontroli i analizy danych. Ethernet zapewnia przewodową komunikację z urządzeniem przez Internet. Przetwornik może wysyłać dane bezpośrednio do chmury, co umożliwia szybką i niezawodną transmisję danych. Aby skorzystać z tej formy komunikacji, konieczne jest podłączenie urządzenia do sieci za pomocą złącza RJ45, które jest powszechnie stosowane w sieciach przewodowych. Bluetooth umożliwia bezprzewodową komunikację na krótki dystans, zazwyczaj do około 10 metrów. Przetwornik może być sparowany z urządzeniem odbiorczym, co umożliwia przesyłanie danych między nimi. Jednak przepustowość danych Bluetooth jest zwykle niższa w porównaniu do Wi-Fi i Ethernet, co może wpływać na szybkość transmisji.

Wybór odpowiedniego protokołu komunikacyjnego zależy od konkretnych wymagań aplikacji i warunków środowiskowych. Wi-Fi i Ethernet są bardziej odpowiednie w przypadku komunikacji na większe odległości i wymagają dostępu do sieci internetowej, podczas gdy Bluetooth jest bardziej odpowiedni do krótkodystansowej komunikacji między urządzeniami. Ważne jest uwzględnienie tych różnych protokołów komunikacyjnych podczas projektowania systemu mechatronicznego, aby zapewnić optymalną komunikację przetwornika z urządzeniem odbiorczym zgodnie z potrzebami i ograniczeniami aplikacji.

#### 4.2.6 Moc obliczeniowa

Moc obliczeniowa w procesorach, mikroprocesorach i mikrokontrolerach odnosi się do zdolności tych urządzeń do wykonywania operacji obliczeniowych i przetwarzania danych. Oznacza to zdolność do wykonywania skomplikowanych instrukcji, algorytmów i operacji matematycznych w określonym czasie. Moc obliczeniowa jest miarą wydajności danego układu elektronicznego i zależy od liczby i organizacji rdzeni procesora, taktowania zegara oraz innych czynników. Im większa moc obliczeniowa, tym większa zdolność urządzenia do wykonywania bardziej zaawansowanych zadań, przetwarzania dużej ilości danych i obsługi bardziej wymagających aplikacji. Kontekst historyczny powstania moc obliczeniowej w procesorach, mikroprocesorach i mikrokontrolerach sięga rozwoju technologii komputerowych i elektroniki. Procesory zaczęły ewoluować od prostych układów wykonujących podstawowe operacje matematyczne do coraz bardziej zaawansowanych struktur, umożliwiających wykonanie bardziej skomplikowanych operacji i przetwarzanie większych ilości danych. Wraz z postępem technologicznym, procesory, mikroprocesory i mikrokontrolery stawały się coraz bardziej zaawansowane i wydajne. Rosła liczba rdzeni w procesorach, wzrastało taktowanie zegara, wprowadzano nowe technologie wykonawcze i poprawiano architekturę, co przekładało się na wzrost mocy obliczeniowej.

W dzisiejszej nowoczesnej elektronice moc obliczeniowa jest kluczowym czynnikiem przy projektowaniu systemów mechatronicznych, urządzeń IoT, smartfonów, komputerów i wielu innych aplikacji. Wysoka moc obliczeniowa umożliwia wykonywanie zaawansowanych algorytmów sztucznej inteligencji, przetwarzanie obrazów, analizę danych w czasie rzeczywistym i wiele innych zadań, które wymagają dużej mocy obliczeniowej. Współczesne procesory, mikroprocesory i mikrokontrolery są wyposażone w zaawansowane jednostki obliczeniowe, pamięć cache, specjalizowane instrukcje i inne funkcje, które umożliwiają wydajne przetwarzanie danych. Również rozwój architektury wielordzeniowej umożliwia równoległe wykonywanie wielu zadań, co jeszcze bardziej zwiększa moc obliczeniową.

Wniosek jest taki, że moc obliczeniowa jest kluczowym aspektem nowoczesnej elektroniki, umożliwiającym wykonywanie skomplikowanych operacji i przetwarzanie dużej ilości danych. Jej rozwój był wynikiem ewolucji procesorów i mikrokontrolerów, które stały się coraz bardziej wydajne i zaawansowane technologicznie. Dzięki wysokiej mocy obliczeniowej możliwe jest realizowanie zaawansowanych aplikacji i zadań w dzisiejszych systemach elektronicznych.

*Tabela 4.8: Analiza mocy obliczeniowej procesora*

<b>Moc obliczeniowa</b>	
<b>Częstotliwość pracy</b>	Im więcej obliczeń można wykonać w ograniczonym czasie, tym bardziej złożone może być przetwarzanie danych
<b>Ilość rdzeni</b>	Wiele operacji może być przetwarzanych w tym samym czasie, wiele obliczeń wstępnego przetwarzania na surowych danych może być wykonywanych w połączeniu z gromadzeniem danych w tym samym momencie
<b>Pamięć cache</b>	Wydajniejsza i większa pamięć cache w procesorze pozwala na szybszy dostęp do danych i instrukcji, co z kolei przyspiesza operacje obliczeniowe.
<b>Instrukcje specjalizowane</b>	Dodanie specjalizowanych instrukcji lub zestawów instrukcji do procesora, które są optymalizowane pod kątem konkretnych zadań, może zwiększyć moc obliczeniową w tych obszarach.

Moc obliczeniowa jest kluczowym czynnikiem wydajności procesora, a tabela 4.8 przedstawia różne elementy, które wpływają na tę moc.

Częstotliwość pracy odzwierciedla liczbę obliczeń, które można wykonać w określonym czasie. Im wyższa częstotliwość, tym bardziej zaawansowane i złożone zadania obliczeniowe można przetwarzać. Ilość rdzeni odnosi się do zdolności procesora do jednoczesnego wykonywania wielu operacji. Większa liczba rdzeni pozwala na równoległe przetwarzanie zadań, co przekłada się na większą moc obliczeniową. Pamięć cache pełni istotną rolę, umożliwiając szybki dostęp do danych i instrukcji. Wydajniejsza i większa pamięć cache przyspiesza operacje obliczeniowe, ponieważ procesor ma łatwiejszy i szybszy dostęp do niezbędnych informacji. Instrukcje specjalizowane są dedykowane do konkretnych zadań i zoptymalizowane pod kątem wydajności. Dodanie tych instrukcji do procesora umożliwia wykonywanie specjalistycznych operacji w bardziej efektywny sposób, co zwiększa moc obliczeniową w tych obszarach.

Te czynniki wpływają na ogólną moc obliczeniową procesora, a ich poprawa przekłada się na lepszą wydajność i zdolność do przetwarzania bardziej zaawansowanych zadań. W nowoczesnej elektronice dąży się do ciągłego doskonalenia tych elementów, aby zapewnić coraz większą moc obliczeniową i efektywność systemów przy jednoczesnym ograniczeniu zużycia energii elektrycznej.

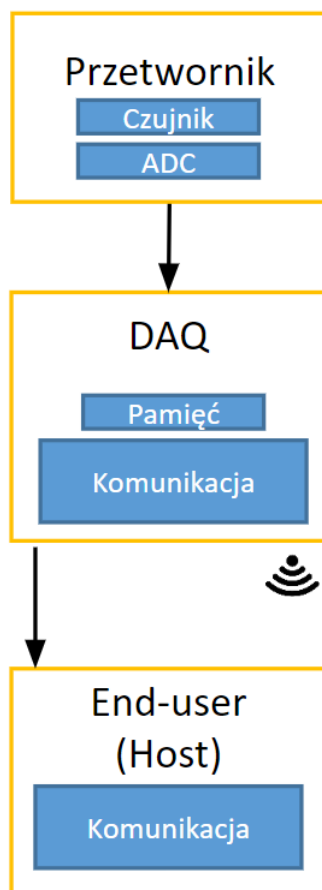
### 4.3 Protokoły

Urządzenie wbudowane to specjalizowane urządzenie, które jest zaprojektowane do pełnienia określonych funkcji lub realizacji konkretnego zadania. Jest ono osadzone lub zawarte w innym obiekcie lub systemie i służy większym celom niż samo urządzenie. Charakteryzuje się wysoką specjalizacją i zoptymalizowaną konstrukcją, co umożliwia efektywne i wydajne działanie w określonym kontekście.

Jednym z przykładów urządzenia wbudowanego jest jednostka akwizycji danych (DAQ). Urządzenie DAQ pełni rolę interfejsu między komputerem (użytkownikiem końcowym lub hostem) a sygnałami pochodzącymi z rozważanej maszyny lub układu. Jego głównym zadaniem jest przekształcenie analogowych sygnałów wejściowych na postać cyfrową, aby umożliwić ich dalszą interpretację i analizę przez komputer. W ramach tego procesu, urządzenia DAQ mogą również zawierać funkcje automatyzacji systemów pomiarowych i procesów.

Protokoły komunikacyjne odgrywają kluczową rolę w urządzeniach wbudowanych, umożliwiając wymianę danych między różnymi komponentami lub systemami. Protokół komunikacyjny określa reguły i procedury, które muszą być przestrzegane w celu poprawnej transmisji i odbioru informacji. Zapewniają one standardy komunikacji, które umożliwiają skuteczną współpracę między urządzeniami. Wybór odpowiedniego protokołu komunikacyjnego zależy od specyfiki urządzenia wbudowanego oraz wymagań dotyczących prędkości, niezawodności, kosztów czy odległości transmisji danych. Protokoły komunikacyjne mogą różnić się pod względem fizycznych warstw transmisji, takich jak połączenia szeregowo czy równoległe, oraz metody przesyłania i odbioru danych, takich jak transmisja asynchroniczna, synchroniczna, czy też bezprzewodowa. W nowoczesnej elektronice zastosowanie protokołów komunikacyjnych jest niezwykle powszechne. Dzięki nim możliwa jest komunikacja między różnymi urządzeniami, które są ze sobą połączone w sieć. Przykładowymi zastosowaniami protokołów komunikacyjnych są systemy IoT (Internet of Things), gdzie urządzenia wbudowane komunikują się ze sobą i wymieniają informacje w celu monitorowania i sterowania różnymi aspektami środowiska. W kontekście historycznym, rozwój protokołów komunikacyjnych był kluczowy dla rozwoju urządzeń wbudowanych. Wcześniej, komunikacja między urządzeniami była ograniczona i wymagała dedykowanych rozwiązań dla każdej aplikacji. Wraz z pojawieniem się standardowych protokołów komunikacyjnych, możliwe stało się tworzenie bardziej uniwersalnych i interoperacyjnych systemów wbudowanych.

Obecnie, moc obliczeniowa i możliwości protokołów komunikacyjnych w urządzeniach wbudowanych stale się rozwijają. Wprowadzane są coraz bardziej zaawansowane protokoły, umożliwiające szybką i niezawodną transmisję danych w różnych dziedzinach, takich jak przemysł, medycyna, transport czy energetyka. Wraz z postępem technologicznym, protokoły komunikacyjne stają się coraz bardziej efektywne, elastyczne i dostosowane do specyficznych wymagań aplikacji, co przyczynia się do dalszego rozwoju urządzeń wbudowanych i ich rosnącej roli we współczesnym świecie.



Rysunek 4.2: Schemat wbudowanego DAQ jako autonomicznego urządzenia w lokalnym systemie monitoringu

Rysunek 4.2 przedstawia proces akwizycji danych, w którym głównymi elementami są przetwornik, jednostka akwizycji danych (DAQ) oraz połączenia pomiędzy DAQ a użytkownikiem końcowym. Na początku mamy przetwornik, który składa się z czujnika i modułu ADC. Czujnik jest odpowiedzialny za pomiar i odczytanie wartości fizycznego parametru. Następnie moduł ADC przekształca ten sygnał analogowy na postać cyfrową, która może być dalej przetwarzana przez komputer. Dane z przetwornika trafiają do jednostki

akwizycji danych, która pełni rolę interfejsu między przetwornikiem a użytkownikiem końcowym. Jego główną funkcją jest przekształcenie danych analogowych na cyfrowe i dostarczenie ich do odpowiedniego medium komunikacyjnego.

W przypadku połączenia przewodowego, jednostka akwizycji danych jest połączona przewodem z użytkownikiem końcowym. Przewód może być różnego rodzaju, takiego jak USB, RS-232, Ethernet itp. Dzięki temu użytkownik końcowy ma możliwość odczytania i interpretacji zgromadzonych danych na swoim komputerze lub urządzeniu. Alternatywnie, jednostka akwizycji danych może być również połączona bezprzewodowo z użytkownikiem końcowym. To oznacza, że dane są przesyłane bez użycia kabli za pomocą technologii bezprzewodowych, takich jak Wi-Fi, Bluetooth lub innych protokołów radiowych. Dzięki temu użytkownik końcowy może zdalnie odczytywać dane z DAQ, co zapewnia większą elastyczność i wygodę w monitorowaniu i analizie zebranych informacji.

W ten sposób, schemat przedstawia kompletny proces akwizycji danych, począwszy od pomiaru sygnału za pomocą przetwornika, poprzez przekształcenie go w postać cyfrową przez moduł ADC, aż do dostarczenia danych do użytkownika końcowego za pośrednictwem jednostki akwizycji danych (DAQ). Zarówno połączenia przewodowe, jak i bezprzewodowe dają możliwość efektywnej transmisji zgromadzonych danych do użytkownika końcowego w celu analizy, monitorowania lub dalszego przetwarzania [4.12-4.17].

### **4.3.1 Przetwornik drgań**

Rozdział ten skupia się na opisie protokołów komunikacyjnych niezbędnych do obsługi przetwornika drgań mechanicznych. Każdy z tych protokołów pełni istotną rolę w zapewnieniu skutecznej transmisji danych z przetwornika do innych urządzeń. Przedstawienie tych protokołów pozwoli zrozumieć ich działanie oraz wybór odpowiedniego protokołu zależnie od wymagań i charakterystyki przetwornika drgań [4.12-4.14].

#### **4.3.1.1 Ograniczenia protokołu komunikacyjnego**

Przy wyborze przetwornika drgań mechanicznych, zarówno analogowego, jak i cyfrowego, istotne jest uwzględnienie potencjalnych ograniczeń związanych z dostępnymi protokołami komunikacyjnymi.

W przypadku przetworników analogowych, ograniczenia związane z protokołami komunikacyjnymi wynikają głównie z ich charakterystyki. Sygnał analogowy jest ciągłym, nieliniowym sygnałem, który musi być odpowiednio próbkowany i przetworzony. Istotne jest,

aby protokół komunikacyjny był w stanie przesyłać sygnał analogowy bez zniekształceń i utraty precyzji. W przypadku przetworników analogowych, często wykorzystuje się interfejsy takie jak napięciowy, prądowy lub interfejs generujący sygnał analogowy w postaci falowej.

Natomiast w przypadku przetworników cyfrowych, ograniczenia związane z protokołami komunikacyjnymi dotyczą głównie przetwarzania i przesyłania danych w postaci cyfrowej. Przetwornik cyfrowy generuje sygnał cyfrowy, który musi być odpowiednio zakodowany i przekazany do urządzenia odbiorczego. Ważne jest, aby protokół komunikacyjny zapewniał odpowiednią przepustowość i niezawodność przesyłania danych cyfrowych. W przypadku przetworników cyfrowych, popularne protokoły komunikacyjne obejmują UART, SPI, I2C, Ethernet, USB i wiele innych.

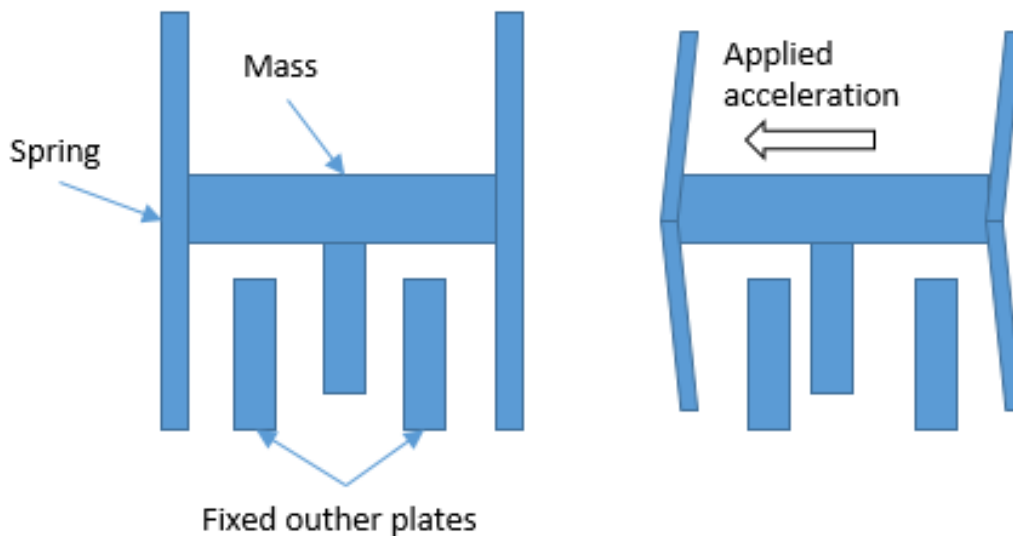
Ograniczenia dotyczące protokołów komunikacyjnych mogą również wynikać z odległości pomiędzy przetwornikiem drgań mechanicznych a urządzeniem docelowym. Niektóre protokoły komunikacyjne mają ograniczenia dotyczące maksymalnego zasięgu transmisji, co może być istotne w przypadku rozległych systemów, gdzie przetwornik drgań mechanicznych jest oddalony od urządzenia odbiorczego. W takich przypadkach konieczne może być zastosowanie protokołów komunikacyjnych, które oferują większy zasięg lub dodatkowe urządzenia wzmacniające sygnał. Ważnym aspektem przy wyborze protokołu komunikacyjnego jest również zgodność z istniejącymi systemami lub platformami, z którymi przetwornik drgań mechanicznych będzie współpracować. Odpowiedni protokół powinien być kompatybilny z danym systemem, umożliwiając prawidłowe odczytywanie, przetwarzanie i interpretację danych z przetwornika.

Ostatecznie, wybór odpowiedniego protokołu komunikacyjnego dla przetwornika drgań mechanicznych, niezależnie od tego, czy jest on analogowy czy cyfrowy, jest kluczowy dla zapewnienia efektywnej i niezawodnej transmisji danych. Warto rozważyć parametry przetwornika, takie jak zakres danych, precyzja, prędkość transmisji, zasięg i zgodność z istniejącymi systemami, aby dokonać odpowiedniego wyboru protokołu komunikacyjnego, który najlepiej spełni wymagania i zapewni niezawodność w przesyłaniu danych z przetwornika drgań mechanicznych do urządzenia odbiorczego.

Akcelerometry MEMS (Micro-Electro-Mechanical Systems) są powszechnie stosowanymi czujnikami przyspieszenia w nowoczesnych urządzeniach elektronicznych. Podstawowym elementem typowego akcelerometru MEMS jest ruchoma konstrukcja belki, składająca się z dwóch zestawów palców. Pierwszy zestaw palców jest przymocowany do

solidnej płaszczyzny uziemienia na podłożu, podczas gdy drugi zestaw palców jest przymocowany do znanej masy zamontowanej na sprężynach. Ta masa może poruszać się w odpowiedzi na przyłożone przyspieszenie. Przyspieszenie to może być przyspieszeniem liniowym lub siłą grawitacyjną. Podczas, gdy akcelerometr MEMS jest poddawany przyspieszeniu, masa na sprężynach przemieszcza się, co powoduje zginanie ruchomej konstrukcji belki. W wyniku tego zginania, odległość pomiędzy stałymi a ruchomymi palcami belki ulega zmianie. Zmiana tej odległości wpływa na pojemność pomiędzy stałymi a ruchomymi palcami belki. Dlatego akcelerometry MEMS wykorzystują zjawisko zmiany pojemności w odpowiedzi na przyspieszenie, które jest nazywane efektem pojemnościowym. Aby dokładnie mierzyć te zmiany pojemności, akcelerometry MEMS wykorzystują technologię mikroskalowej elektromechaniki. Znajdujące się na palcach belki elektrody pojemnościowe tworzą kondensatory, których pojemność zmienia się wraz z ruchem ruchomej konstrukcji. Do pomiaru zmiany pojemności i przetworzenia jej na sygnał elektryczny, akcelerometry MEMS wykorzystują układy elektroniczne wbudowane w chipie. Te układy elektroniczne wykonują odpowiednie przetwarzanie sygnałów, takie jak wzmacnianie, filtracja i konwersja analogowo-cyfrowa, aby uzyskać precyzyjne odczyty przyspieszenia. Otrzymany sygnał przyspieszenia może być dalej wykorzystywany w różnych zastosowaniach, takich jak detekcja kroków, monitorowanie ruchu, stabilizacja obrazu, wykrywanie kolizji czy nawigacja inercyjna.

Dzięki swojej małej wielkości, niskim kosztom produkcji i niskiemu poborowi mocy, akcelerometry MEMS są powszechnie stosowane w wielu dziedzinach, w tym w smartfonach, tabletach, kamerach, nawigacji GPS, samochodach, robotach i wielu innych urządzeniach, które wymagają pomiaru przyspieszenia.



Rysunek 4.3: Schemat akcelerometru MEMS

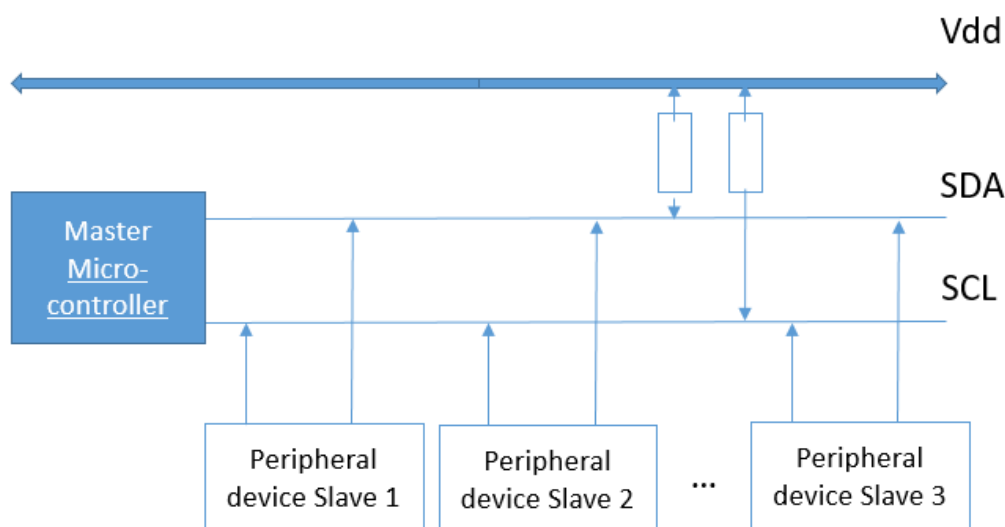
Protokół komunikacji pomiędzy przetwornikiem a SoC jest określany bezpośrednio przez wybrany typ akcelerometru lub ADC. Poniższe podrozdziały opisują szczegółowo możliwe protokoły.

#### 4.3.1.2 Protokół I<sup>2</sup>C

I<sup>2</sup>C (Inter-Integrated Circuit) to popularny protokół komunikacyjny stosowany do połączenia peryferyjnych układów scalonych o niższej prędkości, takich jak sensory, pamięci, przetworniki, ekrany, z procesorami i mikrokontrolerami na krótkich dystansach. Często używany jest do komunikacji wewnątrzpokładowej w różnych systemach elektronicznych. I<sup>2</sup>C charakteryzuje się minimalnym zestawem sygnałów, wykorzystując tylko dwie dwukierunkowe linie: szeregową linię danych (SDA) oraz szeregową linię zegarową (SCL). Obie linie są „podciągnięte rezystorami”. Linia SDA przenosi dane, natomiast linia SCL służy do przesyłania sygnału zegarowego, który synchronizuje przesyłanie danych. Protokół I<sup>2</sup>C umożliwia komunikację między urządzeniami poprzez transmisję i odbiór pakietów danych w formacie bitowym. Transmisja danych odbywa się wzdłuż linii SDA przy pomocy impulsów zegarowych generowanych na linii SCL. Każde urządzenie w sieci I<sup>2</sup>C ma unikalny adres, który identyfikuje je podczas komunikacji. Pozwala to na komunikację wielu urządzeń przy użyciu tej samej magistrali. Typowe napięcia stosowane w protokole I<sup>2</sup>C to +5 V lub +3,3 V, chociaż możliwe są również inne wartości napięć. Ważne jest, aby wszystkie urządzenia podłączone do magistrali I<sup>2</sup>C były zgodne z tym samym poziomem napięcia. Typowe

stosowane napięcia to +5 V lub +3,3 V, chociaż dopuszczalne są systemy z innymi napięciami.

Konstrukcja referencyjna I<sup>2</sup>C ma przestrzeń adresową 7-bitową lub 10-bitową - w zależności od zastosowanego urządzenia. Typowe prędkości magistrali to: 100 kbit/s w trybie standardowym i 10 kbit/s w trybie niskiej prędkości. Ostatnie wersje I<sup>2</sup>C mogą obsługiwać więcej węzłów i działać z większymi prędkościami (tryb szybki 400 kb/s, tryb szybki 1 Mbit/s plus i tryb wysokiej prędkości 3,4 Mb/s). Te wyższe prędkości są szerzej stosowane w nowoczesne systemy wbudowane, w których może być konieczne przesyłanie dużej ilości danych. Warto zauważyć, że szybkości transmisji bitów są podawane dla transakcji między urządzeniem nadrzędnym i podrzędnym, bez wydłużania czasu pracy lub innych narzutów sprzętowych. Narzuty protokołu obejmują adres urządzenia podrzędnego i być może adres rejestru w urządzeniu podrzędnym, a także bity potwierdzone/niepotwierdzone. W związku z tym rzeczywista szybkość przesyłania danych użytkownika jest niższa niż wynikałoby to z samych szczytowych szybkości transmisji bitów. Rysunek 4.4 przedstawia schemat połączenia pomiędzy urządzeniami za pomocą protokołu I<sup>2</sup>C.



Rysunek 4.4: Schemat protokołu komunikacyjnego I<sup>2</sup>C

Schemat składa się z dwóch głównych linii: szeregowej linii danych (SDA) oraz szeregowej linii zegarowej (SCL). Linia SDA przenosi dane, podczas gdy linia SCL generuje sygnały zegarowe służące do synchronizacji przesyłania danych. Obie linie są połączone do wszystkich urządzeń w magistrali I<sup>2</sup>C. Na schemacie można zobaczyć, że linie SDA i SCL są połączone do każdego urządzenia na magistrali. Każde urządzenie ma przypisaną unikalną

adres, który identyfikuje je podczas komunikacji. Adresy urządzeń mogą być 7-bitowe lub 10-bitowe, w zależności od zastosowanej implementacji, co daje maksymalną liczbę węzłów 112, które mogą komunikować się na tej samej magistrali. Oznacza to, że każde urządzenie I<sup>2</sup>C jest rozpoznawane przez unikalny 7-bitowy adres. Należy zauważyć, że maksymalna liczba węzłów jest ograniczona przestrzenią adresową, a także całkowitą pojemnością magistrali, która wynosi 400 węzłów.

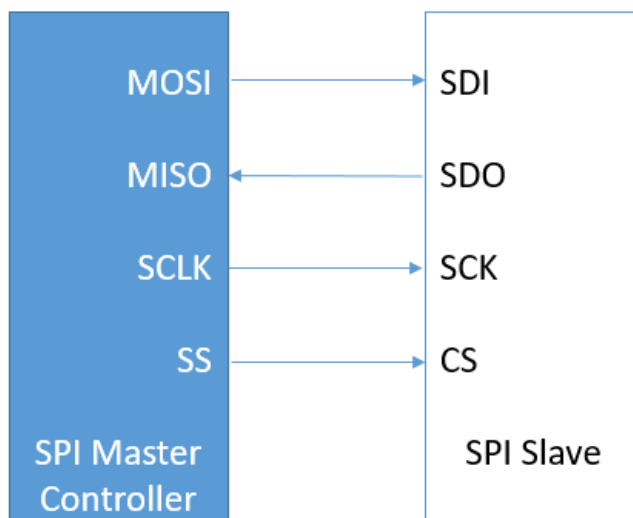
#### *4.3.1.3 Protokół (SPI)*

Protokół SPI (Serial Peripheral Interface) to synchroniczny interfejs komunikacji szeregowej wykorzystywany do komunikacji na niewielkie odległości w systemach wbudowanych. Na podstawie przedstawionego opisu, można wyróżnić kilka istotnych cech protokołu SPI. Jedną z kluczowych cech protokołu SPI jest jego tryb pełnego duplexu, co oznacza, że zarówno urządzenie nadrzędne (master), jak i urządzenia podrzędne (slave), mogą komunikować się ze sobą jednocześnie. Taka architektura umożliwia dwukierunkową transmisję danych, co przyspiesza wymianę informacji między urządzeniami.

W protokole SPI stosowana jest architektura master-slave, gdzie jedno urządzenie pełni rolę nadrzędną (master), a pozostałe urządzenia są podrzędne (slave). Urządzenie nadrzędne kontroluje całą komunikację i generuje ramki do czytania i pisania danych. Ramka składa się z określonej liczby bitów i zawiera informacje, które mają zostać przesłane. W protokole SPI wykorzystywane są cztery linie komunikacyjne: linia zegarowa (SCK), linie danych wyjściowych (MOSI) i wejściowych (MISO) oraz linia wyboru urządzenia (SS). Linia zegarowa (SCK) jest generowana przez urządzenie nadrzędne i synchronizuje przesył danych między urządzeniami. Linie MOSI i MISO służą do transmisji danych w obie strony, odpowiednio z urządzenia nadrzędnego do podrzędnego i vice versa. Linia wyboru urządzenia (SS) pozwala na wybór konkretnego urządzenia podrzędnego, gdy w systemie istnieje ich więcej niż jedno.

Protokół SPI jest popularny w wielu dziedzinach elektroniki, szczególnie w systemach wbudowanych, gdzie występuje potrzeba szybkiej i niezawodnej komunikacji na krótkie odległości. Jest szeroko stosowany w urządzeniach takich jak mikrokontrolery, czujniki, przetworniki danych czy karty pamięci. Jego prostota, szybkość transmisji oraz możliwość jednoczesnej komunikacji w trybie pełnego duplexu czynią go atrakcyjnym rozwiązaniem w wielu zastosowaniach. Warto zauważyć, że protokół SPI może różnić się w zależności od konkretnych implementacji i wymagań aplikacji. Istnieje wiele wariantów protokołu SPI,

takich jak tryb zegarowy, format ramki czy szybkość transmisji, które mogą być dostosowane do konkretnych potrzeb projektu.



Rysunek 4.5: Schemat protokołu komunikacyjnego SPI

Na rysunku 4.5 przedstawiono schemat protokołu komunikacyjnego SPI. Linia zegarowa (SCK) jest generowana przez urządzenie nadrzędne i służy do synchronizacji transmisji między masterem a urządzeniami podrzędnymi. Linia MOSI (Master Output Slave Input) służy do przesyłania danych z mastera do slawów, natomiast linia MISO (Master Input Slave Output) służy do przesyłania danych z slawów do mastera. Linia SS (Slave Select) jest używana do wyboru konkretnego slawa, gdy w systemie istnieje wiele urządzeń podrzędnych.

W komunikacji SPI, master inicjuje transmisję poprzez wysłanie sygnału SS, który wybiera docelowego slawa. Następnie, master generuje sygnał zegarowy (SCK), który synchronizuje przesył danych. W czasie trwania jednego cyklu zegarowego, master wysyła dane na linii MOSI, a slave na linii MISO. Komunikacja odbywa się równocześnie w obie strony, dzięki czemu protokół SPI działa w trybie pełnego duplexu, umożliwiając jednoczesne przesyłanie i odbieranie danych.

Wiele urządzeń podrzędnych jest obsługiwanych poprzez wybór za pomocą indywidualnych linii wyboru urządzenia podrzędnego (SS). Magistrala SPI może działać z jednym urządzeniem nadrzędnym oraz z jednym lub większą liczbą urządzeń podrzędnych. Jeśli używane jest pojedyncze urządzenie podrzędne, pin SS może być ustawiony na stan logiczny niski, jeśli urządzenie podrzędne na to pozwala. W przypadku wielu urządzeń

podrzędnych wymagany jest niezależny sygnał SS z urządzenia nadrzędnego dla każdego urządzenia podrzędnego. Większość urządzeń podrzędnych ma wyjścia z trzema stanami, więc ich sygnał MISO (Master Input Slave Output) ma wysoką impedancję (logicznie rozłączony), gdy urządzenie nie jest wybrane. Urządzenia bez wyjść z potrójnym stanem nie mogą współdzielić segmentów magistrali SPI z innymi urządzeniami; tylko jeden taki slave może komunikować się z węzłem głównym. Spośród tych czterech sygnałów logicznych, dwa z nich MOSI i MISO można zgrupować jako linie danych, a pozostałe dwa SS i SCLK jako linie sterujące.

### 4.3.2 Jednostka akwizycji drgań

W tym rozdziale skoncentrujemy się na opisie elementów i protokołów niezbędnych do obsługi jednostki akwizycji danych. Każdy z nich odgrywa istotną rolę w zapewnieniu skutecznej transmisji bądź przechowywania danych [4.15-4.17].

#### 4.3.2.1 Protokół SPI

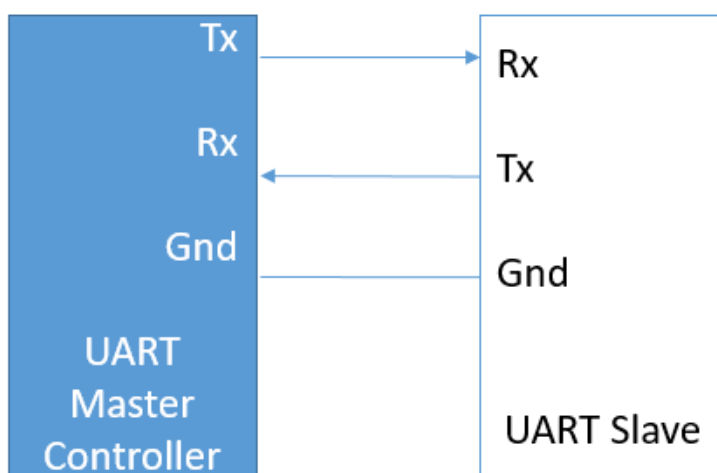
Protokół komunikacyjny SPI został opisany w poprzednim rozdziale.

#### 4.3.2.2 Protokół UART

Universal Asynchronous Receiver-Transmitter (UART) jest powszechnie stosowanym protokołem komunikacyjnym w dziedzinie elektroniki i telekomunikacji. Protokół UART umożliwia przesyłanie danych asynchronicznie, co oznacza, że dane są przekazywane w postaci ciągu bitów, bez zegara sygnalizującego rozpoczęcie i zakończenie transmisji. Jest to jeden z najprostszych i najpopularniejszych protokołów komunikacyjnych wykorzystywanych w układach mikrokontrolerów, mikroprocesorach, modułach komunikacyjnych i innych urządzeniach elektronicznych. Protokół UART wykorzystuje dwa przewody komunikacyjne - jeden dla transmisji danych (TX) i drugi dla odbioru danych (RX). Dane są przekazywane w postaci kolejnych bitów, z których każdy jest poprzedzony bitem startu i zakończony bitem stopu.

Podstawową cechą protokołu UART jest jego prostota. Nie wymaga on złożonego sprzętu ani zegara synchronizującego. Dzięki temu protokół UART jest łatwy w implementacji i znajduje szerokie zastosowanie w różnych dziedzinach, w tym w telekomunikacji, systemach wbudowanych, komunikacji szeregowej między urządzeniami oraz w interfejsach komputerowych. Protokół UART definiuje parametry transmisji, takie jak prędkość transmisji (baud rate), liczba bitów danych, bity parzystości, bity stopu i konfiguracja struktury ramki danych. Prędkość transmisji określa liczbę bitów, które są przesyłane w ciągu jednej sekundy.

Jest to istotne, ponieważ wpływa na szybkość przesyłania danych między urządzeniami. Liczba bitów danych określa ilość informacji przesyłanych w ramce danych. Bity parzystości są używane do wykrywania i korekcji błędów transmisji. Bity stopu służą do sygnalizowania zakończenia transmisji i synchronizacji między urządzeniami. Protokół UART umożliwia jednokierunkową lub dwukierunkową komunikację między urządzeniami. W jednokierunkowej komunikacji, jedno urządzenie pełni rolę nadawcy (Transmitter), a drugie urządzenie pełni rolę odbiornika (Receiver). W dwukierunkowej komunikacji, oba urządzenia mogą zarówno nadawać, jak i odbierać dane.



Rysunek 4.6: Schemat protokołu komunikacyjnego UART

Rysunek 4.6 przedstawia schemat protokołu komunikacyjnego UART, który jest szeroko stosowany w dziedzinie elektroniki i telekomunikacji. Protokół UART umożliwia asynchroniczną transmisję danych między dwoma urządzeniami za pomocą dwóch linii komunikacyjnych - TX (Transmit) i RX (Receive). Na rysunku widoczne są trzy główne elementy protokołu UART: nadajnik (Master Controller), odbiornik (UART Slave) oraz linie komunikacyjne TX i RX oraz linia uziemienia Gnd. Nadajnik jest odpowiedzialny za przesyłanie danych, podczas gdy odbiornik zajmuje się ich odbiorem. Linia TX jest używana do przesyłania danych z nadajnika do odbiornika, natomiast linia RX służy do odbierania danych przez odbiornik.

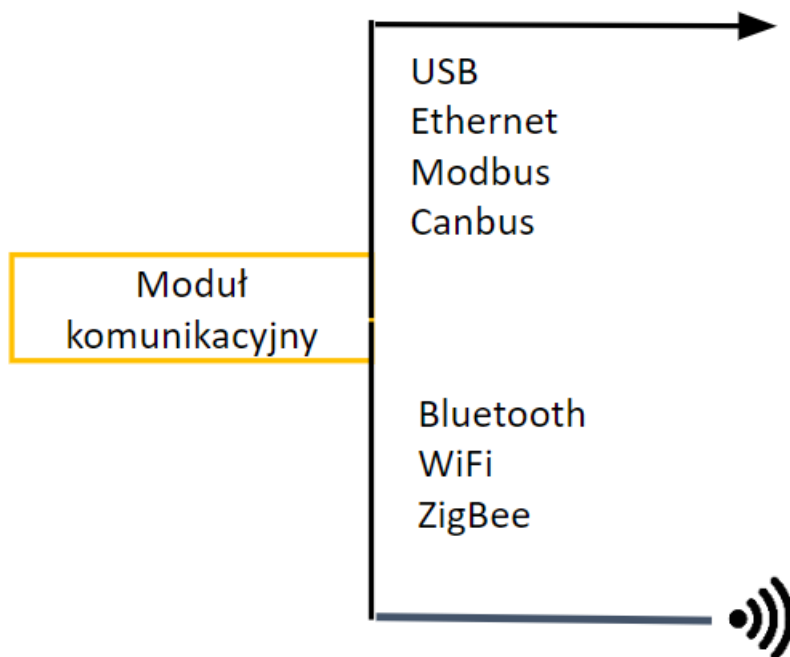
Protokół UART jest kompatybilny z różnymi interfejsami, takimi jak RS-232, RS-485, TTL i inne. Może być stosowany w różnych aplikacjach, takich jak komunikacja między mikrokontrolerami, podłączanie modułów komunikacyjnych (np. Bluetooth, Wi-Fi) do

mikrokontrolerów, komunikacja z urządzeniami peryferyjnymi, takimi jak klawiatury, myszy, drukarki, czujniki i wiele innych. Jedną z zalet protokołu UART jest jego wszechstronność i łatwość integracji. Może być stosowany w różnych konfiguracjach sprzętowych i oprogramowaniu, zgodnie z potrzebami projektu. Dodatkowo, protokół UART umożliwia prostą i niezawodną komunikację między urządzeniami, niezależnie od ich architektury czy platformy. Ważnym aspektem protokołu UART jest również jego niskie zużycie energii. Dzięki temu protokół ten jest często wykorzystywany w urządzeniach mobilnych, które działają na bateriach, oraz w systemach, w których energia jest czynnikiem ograniczającym.

Podsumowując, protokół UART jest popularnym protokołem komunikacyjnym stosowanym w dziedzinie elektroniki i telekomunikacji. Jego prostota, wszechstronność, łatwość integracji i niskie zużycie energii czynią go atrakcyjnym wyborem dla wielu zastosowań. Dzięki protokołowi UART możliwa jest niezawodna i efektywna komunikacja między urządzeniami, przesyłanie danych oraz sterowanie różnymi urządzeniami peryferyjnymi.

### **4.3.3 Uniwersalny system nadawania – połączenia z hostem, użytkownikiem końcowym**

W tym rozdziale skupimy się na opisie kluczowych protokołów komunikacyjnych niezbędnych do połączenia modułu akwizycji i przechowywania danych z modułem końcowym, którym jest komputer użytkownika. Każdy z tych protokołów odgrywa istotną rolę w efektywnej transmisji danych między tymi dwoma urządzeniami.



*Rysunek 4.7: Schemat interfejsów w module komunikacyjnym*

Brama sieciowa to komputer lub inne urządzenie podłączone do sieci komputerowej. Może oferować zasoby informacyjne, usługi i aplikacje użytkownikom lub innym węzłom w sieci. Brama sieciowa to węzeł sieci, któremu przypisano adres hosta warstwy sieciowej. Komputery uczestniczące w sieciach korzystających z pakietu Internet Protocol Suite mogą być również nazywane hostami IP. W szczególności komputery uczestniczące w Internecie nazywane są hostami internetowymi, czasami węzłami internetowymi. Hosty internetowe i inne hosty IP mają jeden lub więcej adresów IP przypisanych do ich interfejsów sieciowych. Adresy są konfigurowane ręcznie przez administratora, automatycznie podczas uruchamiania lub bezstanowymi metodami autokonfiguracji adresów. Każdy host sieciowy jest fizycznym węzłem sieciowym (tj. urządzeniem sieciowym), ale nie każdy fizyczny węzeł sieciowy jest hostem. Urządzenia sieciowe, takie jak modemy, koncentratory i przełączniki sieciowe, nie mają przypisanych adresów hostów (z wyjątkiem czasami do celów administracyjnych) i w związku z tym nie są uważane za hosty sieciowe. Urządzenia, takie jak drukarki sieciowe i routery sprzętowe, mają adresy IP, ale ponieważ nie są to komputery ogólnego przeznaczenia, czasami nie są uważane za hosty.

Hosty sieciowe, które uczestniczą w aplikacjach wykorzystujących model przetwarzania klient-serwer, są klasyfikowane jako systemy serwerowe lub klienckie. Hosty

sieciowe mogą również działać jako węzły w aplikacjach peer-to-peer, w których wszystkie węzły współdzielą i zużywają zasoby w sposób równoważny. Host sieciowy służy jako centrum kontroli systemu DAQ, może on pełnić następujące funkcje:

- Kontrola DAQ,
- Zapewnić graficzny interfejs użytkownika (GUI) do obsługi systemu (z wykorzystaniem web gui opartego na protokole http)
- Przechowywać pozyskane informacje na urządzeniach pamięci masowych
- Zapewnić narzędzia do wyświetlania i/lub analizowania zebranych danych w czasie rzeczywistym lub nie;

#### 4.3.4 Bezprzewodowe protokoły komunikacyjne

##### 4.3.4.1 Protokół Bluetooth

Bluetooth to powszechnie stosowany standard technologii bezprzewodowej, umożliwiający wymianę danych na krótkie odległości za pomocą fal radiowych o częstotliwościach ultrahigh frequency (UHF) w paśmie ISM od 2,4 do 2,485 GHz. Został wprowadzony na rynek w 1994 roku przez firmę Ericsson jako rozwiązanie bezprzewodowe zastępujące tradycyjne przewody RS-232. Protokół Bluetooth umożliwia komunikację między różnymi urządzeniami, zarówno stacjonarnymi, jak i mobilnymi, oraz tworzenie sieci osobistych (PAN), które pozwalają na bezprzewodowe połączenie urządzeń wokół użytkownika. Bluetooth dzieli przesyłane dane na pakiety i przesyła każdy pakiet na jednym z 79 wyznaczonych kanałów Bluetooth. Każdy kanał ma szerokość pasma 1 MHz. Zwykle wykonuje 800 przeskoków na sekundę, z włączoną funkcją Adaptive Frequency-Hopping (AFH). Niskoenergetyczny Bluetooth wykorzystuje odstęp 2 MHz, który obsługuje 40 kanałów.

Główną zaletą Bluetooth jest jego zdolność do równoczesnego połączenia się z siedmioma różnymi urządzeniami, eliminując ograniczenia starszych technologii, które miały trudności z łączeniem się między sobą. Protokół ten jest powszechnie stosowany w urządzeniach takich jak smartfony, tablety, głośniki, słuchawki, klawiatury, myszy i wiele innych, umożliwiając im bezprzewodową komunikację i wymianę danych.

Ważnym aspektem protokołu Bluetooth jest także jego niskie zużycie energii, co pozwala na długotrwałe działanie urządzeń zasilanych bateryjnie. Dodatkowo, Bluetooth jest stosunkowo łatwy w implementacji i obsłudze, co przyczynia się do jego popularności i szerokiego zastosowania w dziedzinie urządzeń mobilnych, inteligentnych domów, systemów

audio i innych dziedzinach, gdzie istnieje potrzeba bezprzewodowej komunikacji na krótkie odległości.

Bluetooth jest powszechnie stosowany w różnych dziedzinach, takich jak motoryzacja, zdrowie, telekomunikacja, audio i wiele innych. Przykładowe zastosowania obejmują bezprzewodowe słuchawki i głośniki, połączenia między smartfonem a samochodem, kontrolery bezprzewodowe do gier wideo oraz monitoring zdrowia i fitness. Wraz z rozwojem technologii, Bluetooth ewoluuje, wprowadzając nowe wersje i ulepszenia. Na przykład Bluetooth 5.0 oferuje większy zasięg, wyższą prędkość przesyłania danych i bardziej efektywne zużycie energii. Dodatkowo, Bluetooth Low Energy (BLE) to wariant protokołu Bluetooth, który jest zoptymalizowany pod kątem minimalnego zużycia energii, co czyni go idealnym rozwiązaniem dla urządzeń zasilanych bateryjnie, takich jak inteligentne zegarki czy czujniki IoT.

Dzięki swojej wszechstronności, wygodzie i niskiemu zużyciu energii, Bluetooth odgrywa ważną rolę w dzisiejszym świecie technologii bezprzewodowych, zapewniając użytkownikom łatwą i niezawodną komunikację między urządzeniami.

#### *4.3.4.2 Protokół WiFi*

WiFi, znane również jako Wireless Fidelity, jest powszechnie stosowanym protokołem bezprzewodowej komunikacji sieciowej. Jest oparty na standardach IEEE 802.11, które określają zasady i specyfikacje techniczne dla bezprzewodowych sieci lokalnych (WLAN). Protokół WiFi umożliwia przesyłanie danych między urządzeniami poprzez fale radiowe w paśmie ISM (Industrial, Scientific, and Medical) o częstotliwościach 2,4 GHz i 5 GHz. WiFi znajduje zastosowanie w różnych typach urządzeń, takich jak komputery osobiste, laptopy, smartfony, tablety, telewizory inteligentne, drukarki i wiele innych. Dzięki technologii WiFi, te urządzenia mogą bezprzewodowo łączyć się z siecią lokalną i uzyskiwać dostęp do Internetu.

Podstawowym elementem architektury WiFi jest punkt dostępowy (Access Point - AP), który działa jako centralny węzeł sieci WLAN. Punkt dostępowy jest odpowiedzialny za zarządzanie i kontrolę połączeń między urządzeniami w sieci oraz przekazywanie danych do sieci zewnętrznej, np. do routera. Punkty dostępowe są zazwyczaj zainstalowane w miejscach publicznych, biurach, domach i innych lokalizacjach, gdzie wymagane jest bezprzewodowe połączenie z Internetem. Podczas przesyłania danych w sieci WiFi, urządzenia korzystają z różnych technik modulacji, takich jak OFDM (Orthogonal Frequency-Division Multiplexing)

i DSSS (Direct Sequence Spread Spectrum), które pozwalają na efektywną transmisję danych przy minimalnych zakłóceniach. Szybkość przesyłania danych w sieci WiFi może wynosić kilka megabitów na sekundę do kilkuset megabitów na sekundę, w zależności od standardu WiFi i sprzętu.

W celu zapewnienia bezpieczeństwa w sieci WiFi, stosowane są różne mechanizmy zabezpieczeń. Wprowadzono protokoły takie jak WEP (Wired Equivalent Privacy), WPA (Wi-Fi Protected Access) i WPA2, które wykorzystują algorytmy szyfrowania do zabezpieczenia transmisji danych. Większość sieci WiFi jest chroniona hasłem, które musi być podane podczas połączenia z siecią w celu uwierzytelnienia i uzyskania dostępu. WiFi oferuje również funkcje takie jak roaming, które umożliwiają urządzeniom przechodzenie z jednego punktu dostępowego do drugiego, zapewniając ciągłość połączenia. Ponadto, technologia WiFi umożliwia tworzenie sieci ad hoc, gdzie urządzenia mogą bezpośrednio ze sobą komunikować się i wymieniać dane bez potrzeby punktu dostępowego.

Wraz z rozwojem technologii, pojawiają się nowe standardy WiFi, takie jak WiFi 6 (802.11ax), które oferują jeszcze większą przepustowość i efektywność energetyczną. WiFi jest niezwykle powszechnym i wygodnym rozwiązaniem, umożliwiającym użytkownikom łatwy dostęp do Internetu i wymianę danych bez konieczności stosowania kabli. Dzięki swojej popularności i rozpowszechnieniu, WiFi jest obecne w wielu środowiskach, takich jak domy, biura, kawiarnie, lotniska, hotele i miejsca publiczne, zapewniając elastyczność i mobilność w komunikacji sieciowej.

### **4.3.5 Przewodowe protokoły komunikacyjne**

#### **4.3.5.1 Protokół Modbus**

Modbus to popularny protokół komunikacyjny wykorzystywany w przemyśle do transmisji danych między urządzeniami w systemach automatyzacji i sterowania. Protokół Modbus został opracowany w latach 70. przez Modicon (obecnie Schneider Electric) i jest szeroko stosowany w różnych branżach, takich jak energetyka, produkcja, budownictwo czy transport.

Protokół Modbus umożliwia komunikację między urządzeniami tzw. master-slave, gdzie jedno urządzenie pełni rolę urządzenia nadrzędnego (master), które inicjuje i kontroluje komunikację, a pozostałe urządzenia działają jako urządzenia podrzędne (slave), odpowiadające na zapytania i dostarczające dane. Komunikacja odbywa się za pośrednictwem różnych interfejsów, takich jak RS-485, Ethernet czy TCP/IP. Protokół Modbus jest prosty i

oparty na prostych strukturach danych, co sprawia, że jest łatwy w implementacji i obsłudze. Składa się z kilku formatów ramki, w tym ramki zapytania i odpowiedzi, które zawierają informacje o adresie urządzenia, funkcji, danych do przesłania i sumie kontrolnej. Komunikacja odbywa się w sposób synchroniczny, gdzie urządzenie nadrzędne wysyła zapytanie do urządzenia podrzędnego, oczekuje na odpowiedź i podejmuje dalsze działania na podstawie otrzymanych danych.

Protokół Modbus oferuje różne funkcje, które umożliwiają odczyt i zapis danych, jak również sterowanie urządzeniami zdalnymi. Na przykład, funkcja odczytu rejestrów (Read Holding Registers) pozwala na odczyt wartości zarejestrowanych danych, a funkcja zapisu rejestrów (Write Single Register) umożliwia zmianę wartości rejestru w urządzeniu podrzędnym. Istnieją także funkcje do sterowania urządzeniami, takie jak uruchamianie i zatrzymywanie procesów.

Protokół Modbus ma wiele zalet, takich jak skalowalność, interoperacyjność i dostępność na różnych platformach i systemach operacyjnych. Ponadto, istnieje wiele urządzeń i systemów, które są kompatybilne z protokołem Modbus, co ułatwia integrację i rozbudowę istniejących systemów. Jednakże, protokół Modbus ma pewne ograniczenia. Przede wszystkim, nie posiada wbudowanych mechanizmów bezpieczeństwa, co może stanowić ryzyko w przypadku transmisji wrażliwych danych. Ponadto, protokół ten ma ograniczoną przepustowość i nie jest odpowiedni do transmisji dużych ilości danych w czasie rzeczywistym. Mimo tych ograniczeń, protokół Modbus pozostaje popularnym wyborem w przemyśle ze względu na swoją prostotę, niezawodność i szerokie wsparcie ze strony producentów. Wraz z rozwojem technologii i wprowadzeniem nowszych wariantów protokołu, takich jak Modbus TCP, Modbus RTU czy Modbus over Ethernet, protokół ten nadal odgrywa kluczową rolę w komunikacji między urządzeniami w systemach automatyzacji przemysłowej.

#### **4.3.5.2 Protokół CAN bus**

Controller Area Network (CAN) to popularny protokół komunikacyjny stosowany w przemyśle motoryzacyjnym oraz w systemach sterowania i automatyzacji. Został opracowany w latach 80. przez firmę Bosch i jest szeroko stosowany ze względu na swoją niezawodność, skalowalność i zdolność do obsługi wielu urządzeń w sieci. Protokół CAN jest zaprojektowany do transmisji danych w czasie rzeczywistym i umożliwia komunikację między różnymi elementami systemu, takimi jak sterowniki, czujniki, aktywatory i inne urządzenia. Opiera się na architekturze master-slave, gdzie istnieje jeden nadzorujący

urządzenie (master), zwane kontrolerem, które inicjuje i zarządza komunikacją, oraz wiele urządzeń podrzędnych (slave), które odpowiadają na żądania i przekazują dane.

W protokole CAN wykorzystuje się dwie linie transmisyjne: CAN High (CAN-H) i CAN Low (CAN-L). Dane są przesyłane w postaci sekwencji bitów, gdzie każde urządzenie w sieci może odczytywać i zapisywać dane. Transmisja danych odbywa się za pomocą metody multipleksowania czasowego, co oznacza, że wiele urządzeń może korzystać z tych samych linii w określonym czasie. Protokół CAN oferuje również zaawansowane mechanizmy kontroli błędów. Wykorzystuje się sumę kontrolną cykliczną (CRC), która pozwala na wykrywanie błędów transmisji danych i gwarantuje ich integralność. Dodatkowo, protokół CAN umożliwia retransmisję danych w przypadku wystąpienia błędów, co zwiększa niezawodność komunikacji.

Ważną cechą protokołu CAN jest jego zdolność do obsługi rozległych sieci, które mogą zawierać wiele urządzeń. Sieci CAN mogą być tworzone zarówno w małych systemach, jak i w dużych kompleksach przemysłowych. Ponadto, protokół CAN jest bardzo elastyczny i umożliwia łatwą rozbudowę i modyfikację systemu poprzez dodawanie nowych urządzeń i konfigurację sieci. Wśród zalet protokołu CAN warto wymienić również jego odporność na zakłócenia elektromagnetyczne i zakłócenia, co jest niezwykle istotne w środowiskach przemysłowych. Ponadto, protokół ten jest energooszczędny, co ma znaczenie w przypadku systemów zasilanych bateryjnie.

Protokół CAN jest szeroko stosowany w branży motoryzacyjnej, gdzie znajduje zastosowanie w systemach zarządzania silnikiem, kontroli trakcji, układach bezpieczeństwa i wielu innych. Ponadto, protokół CAN jest także wykorzystywany w automatyce przemysłowej, monitoringu i systemach sterowania, gdzie umożliwia komunikację między urządzeniami i zdalne zarządzanie procesami. Protokół CAN to niezwykle istotny standard komunikacyjny, który umożliwia efektywną i niezawodną komunikację między urządzeniami w różnych aplikacjach. Jego zalety, takie jak niezawodność, skalowalność, odporność na zakłócenia i zdolność do obsługi rozległych sieci, sprawiają, że jest on szeroko stosowany w przemyśle motoryzacyjnym, automatyce przemysłowej i wielu innych dziedzinach. Protokół CAN zapewnia szybką transmisję danych w czasie rzeczywistym, umożliwiając monitorowanie, sterowanie i zarządzanie systemami z wykorzystaniem wielu urządzeń. Dzięki jego elastyczności i łatwości w rozbudowie, protokół CAN staje się nieodłączną częścią zaawansowanych systemów komunikacyjnych, przyczyniając się do poprawy wydajności, niezawodności i funkcjonalności aplikacji.

### 4.3.5.3 Protokół USB

USB (Universal Serial Bus) jest globalnym standardem komunikacyjnym powszechnie wykorzystywanym do podłączania różnorodnych urządzeń peryferyjnych do komputerów i innych elektronicznych urządzeń. Zapewnia prosty i uniwersalny interfejs, umożliwiający szybką transmisję danych, zasilanie urządzeń oraz inne funkcje. Szyna USB składa się z dwóch głównych komponentów: hosta i urządzenia. Hostem jest zazwyczaj komputer lub inny system zarządzający, który kontroluje komunikację z podłączonymi urządzeniami. Urządzeniem może być szeroki zakres urządzeń peryferyjnych, takich jak drukarki, klawiatury, myszki, kamery, pendrive'y i wiele innych.

Główne cechy i korzyści płynące z użycia interfejsu USB to: uniwersalność, łatwość w obsłudze, plug-and-play (podłącz i korzystaj), obsługa hot-swap (możliwość podłączania i odłączania urządzeń w czasie pracy systemu), wsparcie dla różnych prędkości transmisji danych (od niskich do bardzo wysokich), dostępność zasilania urządzeń peryferyjnych, możliwość przesyłania zarówno danych cyfrowych, jak i zasilania przez ten sam interfejs.

USB wykorzystuje różne rodzaje kabli i złączy w zależności od generacji i typu urządzeń. Najpopularniejsze złącza to USB Type-A (standardowe złącze hosta), USB Type-B (złącze urządzenia), USB Micro-B (często stosowane w starszych smartfonach i innych urządzeniach przenośnych) oraz USB Type-C (nowoczesne, dwustronne i odwracalne złącze).

Protokół komunikacyjny USB obejmuje różne warstwy, takie jak warstwa fizyczna, warstwa łącza danych, warstwa protokołu i warstwa aplikacji. Wykorzystuje się również różne rodzaje pakietów danych, takie jak pakiety kontrolne, pakiety danych i pakiety zgłoszeniowe, które są odpowiedzialne za przekazywanie informacji i zarządzanie komunikacją. USB oferuje również wsparcie dla różnych prędkości transmisji danych. Pierwotnie pojawiła się wersja USB 1.0 z prędkością 1,5 Mbps i 12 Mbps. Następnie wprowadzono USB 2.0, który oferował prędkość do 480 Mbps. Obecnie najnowsza generacja to USB 3.2, która zapewnia prędkości do 20 Gbps.

USB jest szeroko stosowanym standardem w przemyśle elektronicznym i jest obsługiwany przez większość systemów operacyjnych, co czyni go niezwykle popularnym wśród producentów urządzeń i użytkowników końcowych. Jego wszechstronność, prostota obsługi i szerokie zastosowanie czynią go nieodłączną częścią dzisiejszych technologii komunikacyjnych i multimedialnych.

#### 4.3.6 Analiza zbiorcza

W tabeli 4.9 przedstawiono porównanie właściwości dostępnych protokołów połączeniowych, które można wykorzystać w celu podłączenia urządzeń peryferyjnych do DAQ. Zapewnia porównanie tych interfejsów w oparciu o różne czynniki, które obejmują możliwą odległość, rodzaj komunikacji, zegar, złożoność sprzętu i oprogramowania, zalety i wady. W dzisiejszych systemach wbudowanych i urządzeniach elektronicznych, te protokoły odgrywają kluczową rolę w efektywnej wymianie danych między różnymi komponentami systemu. Celem tej analizy jest zgłębienie charakterystyk, funkcjonalności, mocnych stron i potencjalnych ograniczeń każdego z tych protokołów. Przeanalizowanie tych trzech protokołów komunikacyjnych pozwoli nam na lepsze zrozumienie ich charakterystyk, możliwości i ograniczeń, co z kolei pomoże nam w wyborze odpowiedniego protokołu do konkretnych zastosowań w projektowaniu systemów wbudowanych i elektroniki użytkowej.

*Tabela 4.9: Porównanie protokołów połączeń CMS stosowanych w części systemu wbudowanego*

	<b>UART</b>	<b>SPI</b>	<b>I2C</b>
<b>Szybkość przesyłania danych</b>	Obsługuje od około 230 kb/s do 460 kb/s.	Obsługa od około 10 Mb/s do 20 Mb/s	Obsługuje prędkości 100 kb/s, 400 kb/s, 3,4 Mb/s
<b>Odległość</b>	Niska	Największa	Średnia
<b>Typ komunikacji</b>	Asynchroniczna	Synchroniczna	Synchroniczna
<b>Ilość urządzeń master</b>	N/A	Jedno	Jedno lub wiele
<b>Sygnal zegara</b>	Nie jest używany wspólny sygnał zegara	Jeden wspólny sygnał zegara szeregowego	Wspólny sygnał zegara
<b>Sposób adresowania</b>	N/A	'n' linii wyboru slave na urządzeniu master dla 'n' urządzeń sl	Do 27 urządzeń podrzędnych
<b>Zalety</b>	Prosty protokół	Komunikacja w pełnym duplexie, mechanizm Push-pull, SPI zużywa mniej energii niż I2C	Więcej niż jeden master, proste adresowanie, elastyczność napięcia magistrali, wykorzystuje kontrolę przepływu
<b>Wady</b>	Komunikacja tylko między dwoma urządzeniami, początkowo ustawiona stała szybkość transmisji danych	Złożoność sprzętowa, brak możliwości zmiany relacji master i slave, brak kontroli przepływu w SPI	Zwiększa złożoność obwodu, gdy wzrasta liczba urządzeń podrzędnych i nadrzędnych, Interfejs półduplexowy, Koszty przetwarzania sygnałów

W celu doboru odpowiedniego protokołu komunikacji pomiędzy elementami nowoczesnej jednostki akwizycji danych do wykorzystania w Systemie Monitorowania Stanu należy przeanalizować tabelę 4.9, gdyż wybór protokołu może mieć wpływ na pracę

urządzenia, a zwłaszcza na wydajność pomiaru danych i transfer danych. Pod względem szybkości przesyłania danych, UART obsługuje prędkości od około 230 kb/s do 460 kb/s, podczas gdy SPI osiąga prędkości od około 10 Mb/s do 20 Mb/s. Natomiast I2C oferuje prędkości transmisji wynoszące 100 kb/s, 400 kb/s i 3,4 Mb/s. Jeśli chodzi o odległość komunikacji, UART ma niską zasięg, SPI zapewnia największą odległość, a I2C mieści się w średnim zakresie. Pod względem typu komunikacji, UART działa w trybie asynchronicznym, podczas gdy zarówno SPI, jak i I2C korzystają z synchronicznej komunikacji. Jeśli chodzi o ilość urządzeń master, UART nie ma tej funkcji, a zarówno SPI, jak i I2C obsługują jedno lub wiele urządzeń master. Pod względem sygnału zegara, UART nie korzysta z wspólnego sygnału zegara, podczas gdy zarówno SPI, jak i I2C wykorzystują wspólny sygnał zegara. Sposób adresowania różni się w tych protokołach. UART nie posiada specyficznego sposobu adresowania, SPI korzysta z linii wyboru slave, które są sterowane przez urządzenie master, a I2C wykorzystuje adresy dla urządzeń podrzędnych, z możliwością obsługi do 27 urządzeń podrzędnych. W tabeli wymienione są również zalety i wady każdego z tych protokołów. UART jest prosty w implementacji, jednak komunikuje się tylko między dwoma urządzeniami. SPI oferuje komunikację w pełnym duplexie, zużywa mniej energii niż I2C, ale wymaga większej złożoności sprzętowej. I2C umożliwia więcej niż jeden master, proste adresowanie i elastyczność napięcia magistrali, ale zwiększa złożoność obwodu przy większej liczbie urządzeń. Jeśli chodzi o wady, UART ma ograniczenia dotyczące liczby urządzeń i stałej szybkości transmisji danych. SPI jest bardziej skomplikowany pod względem sprzętowym, brakuje możliwości zmiany relacji master-slave i kontroli przepływu. I2C zwiększa złożoność obwodu przy większej liczbie urządzeń i ma interfejs półduplexowy, a koszty przetwarzania sygnałów mogą być wyższe.

*Tabela 4.10: Porównanie protokołów połączeń CMS używanych między systemem wbudowanym a bramą sieciową (lub użytkownikiem końcowym)*

	<b>Bluetooth</b>	<b>Wifi</b>	<b>Canbus</b>	<b>Modbus</b>
<b>Przesył danych</b>	do 1 Mb/s	do 54 Mb/s	High Speed: do 1 Mbit/s Low Speed: do 125 Kb/s	do 1Gb/s
<b>Częstotliwość pracy</b>	2400 MHz	2400 MHz,5000 MHz	-	-
<b>Zasięg</b>	10m	100m	High speed: 40m Low speed: 500m	1000m
<b>Pasmo</b>	1 MHz	22 MHz	-	-
<b>Rozmiar sieci</b>	8	2000	110 / 253	254

<b>Zalety</b>	Prosty, Bezprzewodowy	Bezprzewodowy, duży zasięg, duży transfer	Niezawodność i odporność na zakłócenia, prędkość transmisji, skalowalność	Duży zasięg, duży transfer
<b>Wady</b>	Mały zasięg, Ograniczona ilość urządzeń	Zużycie energii	Koszty, złożoność	Brak wsparcia dla złożonych struktur, Ilość urządzeń, brak mechanizmów do synchronizacji czasu

Ponieważ wprowadzenie protokołu komunikacji bezprzewodowej między urządzeniem wbudowanym, a urządzeniem użytkownika końcowego może być konieczne, należy wziąć pod uwagę parametry szybkości transmisji danych, nominalny zakres protokołu i wydajność energetyczną, a zatem przeanalizować tabelę 4.10 podsumowującą porównanie opisanych interfejsów, w tym protokoły komunikacji bezprzewodowej i przewodowej.

**Bluetooth:** Protokół Bluetooth oferuje przesył danych o maksymalnej prędkości do 1 Mb/s. Ma zasięg do 10 metrów, co czyni go idealnym do krótkich dystansów, takich jak połączenia między urządzeniami osobistymi. Jest prosty w użyciu i zapewnia bezprzewodową komunikację.

**WiFi:** Technologia WiFi zapewnia znacznie wyższe prędkości transmisji danych, sięgające do 54 Mb/s. Ma większy zasięg do 100 metrów i szerokie pasmo pracy wynoszące 22 MHz. Jest powszechnie stosowana w sieciach bezprzewodowych, oferując duży transfer danych i długą zasięgowość.

**CANbus:** Protokół CANbus jest szeroko stosowany w systemach samochodowych i przemysłowych. Posiada dwa tryby pracy: High Speed, który obsługuje prędkość do 1 Mbit/s, oraz Low Speed, z prędkością do 125 Kb/s. Ma zasięg do 500 metrów w trybie Low Speed i 40 metrów w trybie High Speed. Jest znany z niezawodności, odporności na zakłócenia i skalowalności.

**Modbus:** Protokół Modbus jest często wykorzystywany w systemach sterowania i monitoringu. Oferuje prędkość transmisji do 1 Gb/s w trybie High Speed i do 1000 metrów zasięgu. Ma ograniczoną liczbę urządzeń w sieci, wynoszącą 254. Protokół Modbus jest ceniony ze względu na duży zasięg i transfer danych, ale nie obsługuje złożonych struktur ani mechanizmów synchronizacji czasu.

Wady i zalety każdego protokołu są związane z jego specyfiką. Bluetooth ma ograniczony zasięg i ilość urządzeń, WiFi zużywa więcej energii, CANbus może być kosztowny i skomplikowany, a Modbus nie obsługuje złożonych struktur danych i wymaga dodatkowych mechanizmów synchronizacji czasu. Wybór protokołu zależy od konkretnych wymagań aplikacji i warunków pracy.

Podsumowując, komunikacja pomiędzy komponentami w systemach wbudowanych jest kluczowym aspektem projektowania, mającym wpływ na wydajność, niezawodność i efektywność całego systemu. Istnieje wiele protokołów komunikacyjnych, które zostały opracowane w celu umożliwienia płynnej i efektywnej wymiany danych pomiędzy różnymi elementami systemu. Trzy z najpopularniejszych protokołów komunikacyjnych wykorzystywanych w systemach to SPI (Serial Peripheral Interface), I2C (Inter-Integrated Circuit) oraz UART (Universal Asynchronous Receiver/Transmitter).

SPI jest jednym z najszybszych protokołów komunikacyjnych i jest często stosowany do komunikacji z urządzeniami peryferyjnymi, takimi jak czujniki, konwertery ADC (Analog-to-Digital Converter), układy FPGA czy pamięci flash. Jest to protokół synchroniczny, który wymaga jednej linii zegara i dwóch linii danych (MOSI - Master Out Slave In i MISO - Master In Slave Out). SPI jest szczególnie przydatny w aplikacjach, gdzie prędkość komunikacji jest kluczowym czynnikiem, a odległość pomiędzy urządzeniami nie jest duża. I2C jest protokołem szeregowym, który umożliwia komunikację pomiędzy wieloma urządzeniami za pomocą tylko dwóch linii - SDA (Serial Data Line) i SCL (Serial Clock Line). Jest to protokół stosowany do komunikacji z układami peryferyjnymi, takimi jak czujniki, ekrany LCD, pamięci EEPROM i wiele innych. I2C charakteryzuje się prostotą i oszczędnością zasobów, dlatego często jest stosowany w systemach, gdzie liczba połączonych urządzeń jest dość duża. UART to popularny protokół komunikacyjny asynchroniczny, który jest często używany do komunikacji pomiędzy mikrokontrolerami i innymi urządzeniami peryferyjnymi. W protokole UART dane są przesyłane w postaci pakietów, a nie w formie ramki, co sprawia, że jest on stosunkowo prosty do implementacji. UART jest często stosowany w systemach wbudowanych, gdzie prostota i niski koszt są priorytetem. Dla potrzeb komunikacji bezprzewodowej w systemach, WiFi i Bluetooth są najczęściej stosowanymi protokołami. WiFi zapewnia szybką komunikację na większe odległości i jest szczególnie przydatne w aplikacjach wymagających transmisji danych w czasie rzeczywistym. Bluetooth natomiast jest często używany do krótkiego zasięgu, jak np. w systemach IoT, które wymagają komunikacji pomiędzy urządzeniami w pobliżu siebie. W

przypadku, gdy konieczne jest komunikowanie się z urządzeniami z różnych producentów, stosuje się protokół Modbus. Jest to protokół komunikacyjny, który umożliwia przesyłanie danych pomiędzy urządzeniami z różnych producentów poprzez różne interfejsy komunikacyjne, takie jak RS-232, RS-485 czy Ethernet.

Wybór odpowiedniego protokołu komunikacyjnego w systemach wbudowanych zależy od wielu czynników, takich jak prędkość transmisji danych, odległość komunikacji, prostota implementacji oraz specyficzne wymagania projektu. Świadome i odpowiednie dopasowanie protokołu komunikacyjnego pozwala na efektywną i niezawodną wymianę danych pomiędzy różnymi elementami systemu.

#### 4.4 Przegląd i porównanie najpopularniejszych systemów wbudowanych

Procesor wbudowany to jednostka centralna systemu wbudowanego, która steruje podstawowymi funkcjami systemu. Jest on rodzajem mikroprocesora i posiada zazwyczaj ograniczone funkcje, zużywając przy tym zauważalnie mniej mocy w stosunku do procesorów tradycyjnych. Wbudowane procesory są dostarczane tylko z procesorem w chipie (SoC). Do działania wymagają pamięci zewnętrznej oraz innych zewnętrznych komponentów i urządzeń peryferyjnych. Mikrokontrolery te mają wszystko, co potrzebne do wykonania zadania — w tym pamięć, rejestry sygnałów i porty wejścia/wyjścia — wbudowane w układ. Ich główną zaletą jest to, że zużywają mniej energii i ostatecznie mogą być tańszym rozwiązaniem niż procesor. Podstawową wadą wyboru SoC jest to, że producenci, którzy z nich korzystają, mają ograniczoną elastyczność w zmianie systemu. Na przykład w przypadku procesora i pamięci zewnętrznej producent może zdecydować o zmianie lub dodaniu pamięci zewnętrznej. W przypadku SoC nie jest to możliwe. Taka zmiana wymusza całkowitą wymianę SoC, co może wiązać się ze znaczną modyfikacją całego systemu wbudowanego.

Decydując się na odpowiedni procesor lub układ SoC dla systemu wbudowanego, należy wziąć pod uwagę wiele czynników. Najważniejsze z nich to oczywiście: ogólna wydajność, zapotrzebowanie na pamięć, potrzeby związane z urządzeniami peryferyjnymi oraz koszty.

Oto kluczowe kwestie, które należy wziąć pod uwagę przy wyborze procesora lub SoC:

- Wydajność: najważniejszym czynnikiem jest oczywiście ogólna wydajność i szybkość procesora, które określa architektura procesora lub SoC oraz konstrukcja układu scalonego.
- Pamięć: wybierając mikrokontroler lub SoC, należy upewnić się, że ma on możliwości pamięci, których potrzebuje przyszłe oprogramowanie.

- Pobór mocy: wysoka wydajność procesora oznacza jednocześnie większe zużycie energii.
- Urządzenia peryferyjne: właściwa ich obsługa w systemie wbudowanym jest niezbędna. W przypadku SoC urządzenia peryferyjne często znajdują się w chipie.
- Napięcia robocze: w ostatnich latach chipy komputerowe zmniejszyły zużycie energii procesora poprzez zmniejszenie napięcia roboczego chipa, jednakże należy brać pod uwagę, iż ma to bezpośredni wpływ na wydajność całego układu
- Koszt: koszt procesora lub SoC będzie oczywiście kluczowym czynnikiem w procesie produkcji.

#### 4.4.1 ESP32

ESP32 jest wyjątkową płytą rozwojową, idealną do budowy platformy wbudowanej dedykowanej akwizycji danych z czujników drgań mechanicznych. Dzięki swojej wszechstronności i zaawansowanej architekturze, ESP32 oferuje wiele funkcji i możliwości, które umożliwiają skuteczną i precyzyjną analizę oraz monitorowanie drgań mechanicznych [4.18-4.21].

Płyta ESP32 jest wyposażona w dwurdzeniowy procesor taktowany zegarem o częstotliwości do 240 MHz, co zapewnia wystarczającą moc obliczeniową do przetwarzania danych z czujników. Dodatkowo, ESP32 posiada wbudowaną pamięć Flash i SRAM, co umożliwia przechowywanie programów oraz danych bezpośrednio na płycie. ESP32 oferuje bogaty zestaw interfejsów komunikacyjnych, w tym UART, SPI i I2C. Dzięki nim można bezproblemowo połączyć czujniki drgań mechanicznych z płytą i przesyłać dane pomiarowe w sposób efektywny i niezawodny. Interfejsy te umożliwiają również integrację z innymi peryferiami i modułami, takimi jak wyświetlacze, karty SD czy moduły komunikacji bezprzewodowej. ESP32 obsługuje również technologie bezprzewodowe, w tym WiFi i Bluetooth. To otwiera możliwość przesyłania danych pomiarowych bezprzewodowo do innych urządzeń lub do chmury, co znacznie zwiększa elastyczność i dostępność danych z czujników drgań mechanicznych. Możliwość komunikacji przez Bluetooth pozwala również na sterowanie platformą zdalnie z urządzeń mobilnych.

Jedną z największych zalet ESP32 jest również jego obszerna społeczność programistów i dostępność różnych narzędzi programistycznych oraz bibliotek. Dzięki temu programowanie platformy wbudowanej opartej na ESP32 jest znacznie ułatwione, a tworzenie aplikacji związanych z akwizycją danych z czujników drgań mechanicznych staje się prostsze i bardziej efektywne.

Podsumowując, ESP32 stanowi doskonałą podstawę do budowy platformy wbudowanej przeznaczonej do akwizycji danych z czujników drgań mechanicznych. Dzięki swojej wszechstronności, wydajności i obszernej społeczności programistów, ESP32 umożliwia precyzyjną analizę i monitorowanie drgań mechanicznych oraz dostarcza niezawodne rozwiązania w obszarze akwizycji danych.

*Tabela 4.11: Parametry techniczne układu ESP32*

	<b>ESP32</b>
<b>CPU</b>	CoreMark 2-core 32-bit
<b>CPU FREQ</b>	204 Mhz
<b>RAM</b>	520 KB
<b>FLASH</b>	4 MB
<b>I/O</b>	34
<b>ADC</b>	2x 8bit
<b>Timer</b>	2x 64bit
<b>DMA</b>	1x12
<b>USART</b>	2
<b>SPI</b>	4
<b>I2C</b>	2
<b>Wi-Fi</b>	802.11n(2.4GHz) do 150Mbps
<b>Bluetooth</b>	V4.2 oraz BLE
<b>Ethernet</b>	Brak
<b>SD Slot</b>	0
<b>Zużycie energii</b>	95-240 uA

W tabeli 4.11 przedstawiono parametry techniczne układy wbudowanego ESP32. Układ ten jest popularnym wyborem do budowy platform wbudowanych, charakteryzuje się szeregiem kluczowych parametrów. Jest to dwurdzeniowy procesor 32-bitowy, który oferuje wsparcie dla wielu funkcji i zapewnia odpowiednią moc obliczeniową. Częstotliwość procesora wynosi 204 MHz, co przekłada się na wydajność działania. Układ ten posiada 520 KB pamięci RAM, umożliwiającej przechowywanie danych w trakcie działania. Dodatkowo, wyposażony jest w 4 MB pamięci Flash, służącej do przechowywania programów i danych. ESP32 oferuje 34 piny I/O, co umożliwia podłączenie różnych urządzeń i czujników. W zakresie akwizycji danych, posiada dwa wejścia analogowo-cyfrowe (ADC) o rozdzielczości 8 bitów. Dzięki temu jest w stanie mierzyć przyspieszenie, nachylenie oraz wibracje lub wstrząsy, co czyni go idealnym rozwiązaniem w aplikacjach związanych z czujnikami drgań mechanicznych. Układ ESP32 jest wyposażony w dwa 64-bitowe liczniki czasu (timery), które mogą być wykorzystane do precyzyjnego pomiaru czasu. Posiada również jeden kanał DMA, umożliwiający efektywny transfer danych między różnymi urządzeniami. W zakresie komunikacji, oferuje dwa interfejsy I2C, dwa interfejsy USART oraz cztery interfejsy SPI, które umożliwiają komunikację z innymi urządzeniami. Jeśli chodzi o bezprzewodową

komunikację, układ ESP32 posiada moduł Wi-Fi z obsługą standardu 802.11n, co umożliwia komunikację z innymi urządzeniami i sieciami w trybie bezprzewodowym, o maksymalnej prędkości do 150 Mbps. Dodatkowo, jest wyposażony w moduł Bluetooth w wersji 4.2 oraz obsługę BLE (Bluetooth Low Energy), co umożliwia bezprzewodową komunikację w krótkim zasięgu. Należy zauważyć, że układ ESP32 nie posiada wbudowanego modułu Ethernet oraz gniazda na kartę SD. Niewątpliwie jednym z kluczowych atutów układu ESP32 jest jego niskie zużycie energii, oscylujące w zakresie od 95 do 240 uA. Dzięki temu jest on idealnym wyborem do aplikacji, które wymagają oszczędności energii i działania przy niskich mocach.

Podsumowując, układ ESP32 jest bardzo wszechstronnym i uniwersalnym rozwiązaniem do budowy platform wbudowanych, szczególnie w przypadku aplikacji związanych z akwizycją danych z czujników drgań mechanicznych. Dzięki swoim parametrom, takim jak dwurdzeniowy procesor, pamięć RAM i Flash, interfejsy komunikacyjne oraz moduły Wi-Fi i Bluetooth, zapewnia on niezbędne narzędzia do stworzenia efektywnych i energooszczędnych systemów wbudowanych. Z tego też powodu jest doskonałą podstawą do budowy platformy wbudowanej dedykowanej akwizycji danych z czujników drgań mechanicznych.

#### 4.4.2 STM32

Płytką STM32 stanowi solidną podstawę do budowy platformy wbudowanej dedykowanej akwizycji danych z czujników drgań mechanicznych. Wyposażona w zaawansowany mikrokontroler z rodziny STM32, oferuje wiele funkcji i możliwości, które umożliwiają precyzyjne monitorowanie oraz analizę drgań mechanicznych. Mikrokontrolery STM32 są znane z wysokiej wydajności i elastyczności. Posiadają różne warianty, które różnią się m.in. liczbą pinów, pamięcią i mocą obliczeniową. Dzięki temu można dostosować płytę STM32 do konkretnych wymagań aplikacji związanych z akwizycją danych z czujników drgań mechanicznych [4.22-4.25].

STM32 oferuje rozbudowane interfejsy komunikacyjne, takie jak UART, SPI i I2C. Pozwala to na bezproblemowe podłączenie czujników drgań mechanicznych do płytki i przesyłanie danych pomiarowych w sposób niezawodny. Interfejsy te umożliwiają również komunikację z innymi urządzeniami peryferyjnymi, co zwiększa funkcjonalność platformy wbudowanej. Płytką STM32 obsługuje również technologie bezprzewodowe, takie jak WiFi i Bluetooth. Dzięki temu dane pomiarowe z czujników drgań mechanicznych mogą być przesyłane bezprzewodowo do innych urządzeń lub do chmury, co zwiększa dostępność i elastyczność systemu. Dodatkowo, istnieje wiele gotowych bibliotek i narzędzi programistycznych, które ułatwiają implementację komunikacji bezprzewodowej.

STM32 oferuje zaawansowane funkcje przerwań, które pozwalają na szybką i precyzyjną reakcję na zdarzenia wywołane przez czujniki drgań mechanicznych. Można programować mikrokontroler w języku C lub C++, co daje dużą swobodę w tworzeniu aplikacji i algorytmów analizy danych. Płytki STM32 jest również wyposażona w różne złącza i interfejsy, takie jak GPIO, ADC czy Timery, które umożliwiają bezpośrednie podłączenie czujników drgań mechanicznych oraz innych modułów peryferyjnych. To zapewnia szerokie możliwości rozwoju i rozbudowy platformy wbudowanej.

Tabela 4.12: Porównanie różnych modeli STM32 z różnymi rdzeniami arm cortex M

Series STM32	L1	L4	F1	F3	F2	F4	F7
Micro	L152RE	L476RG	F103RB	F303RE	F207ZG	F401RE	F767ZI
CPU	M3	M4	M3	M4	M3	M4	M7
CPU FREQ	32Mhz	80MHz	72MHz	72MHz	120MHz	84Mhz	216MHz
RAM	80KB	128KB	20KB	64KB	128KB	96KB	512KB
FLASH	512KB	1MB	128KB	512KB	1MB	512KB	2MB
I/O	116	114	80	115	140	81	168
ADC	1x 12bit 1Msps	3x 12bit 5Msps	2x12bit 1Msps	1x12bit 5Msps	1x12bit 2Msps	1x12bit 2.4Msps	3x12bit 2.4Msps
Timer	1x 32bit , 10x 16bit	2x32bit, 14x16bit	7x16bit	1x32bit 13x16bit	2x32bit 15x16bit	2x32bit 9x16bit	2x32bit 16x16bit
DMA	1x12	1x14	1x7	1x12	1x16	1x16	1x16
USART	5	6	3	5	4	3	4
SPI	8x 16Mbit/s	3x16 Mbit/s	2x18 Mbit/s	4x16 Mbit/s	3x30 Mbit/s	4x42 Mbit/s	6x54 Mbit/s
I2C	2	3	2	3	3	3	4
Wi-Fi	None	None	None	None	None	None	None
Bluetooth	None	None	None	None	None	None	None
Ethernet	None	None	None	None	None	None	None
SD Slot	None	None	None	None	None	None	None
Zużycie energii	195 uA/MHz	100 uA/MHz	N/A	N/A	N/A	146 uA/MHz	N/A

W tabeli 4.12 przedstawiono wybrane serie mikrokontrolerów STM32 wraz z ich parametrami. Series STM32: Tabela zawiera informacje o różnych seriach mikrokontrolerów STM32, takich jak L1, L4, F1, F3, F2, F4 i F7.

Kolumna "Micro" zawiera konkretne modele mikrokontrolerów w ramach każdej serii, takie jak L152RE, L476RG, F103RB, F303RE, F207ZG, F401RE i F767ZI. W kolumnie "CPU" znajdują się informacje o rodzaju procesora w mikrokontrolerach, takich jak M3 (Cortex-M3) i M4 (Cortex-M4). Natomiast w kolumnie "CPU FREQ" przedstawiona jest częstotliwość pracy procesora w megahercach (MHz) dla poszczególnych modeli. W kolumnie "RAM" podane są ilości pamięci RAM dostępnej w kilobajtach (KB) dla każdego

modelu mikrokontrolera. Kolumna "FLASH" przedstawia pojemność pamięci flash w kilobajtach (KB) lub megabajtach (MB) dla każdego modelu. Liczba dostępnych pinów wejścia/wyjścia (I/O) w mikrokontrolerach jest przedstawiona w kolumnie "I/O". Kolumna "ADC" zawiera informacje o liczbie konwerterów analogowo-cyfrowych (ADC) oraz ich rozdzielczości w bitach dla każdego modelu. W kolumnie "Timer" przedstawiona jest liczba timerów oraz ich rozmiar (32-bitowy lub 16-bitowy) dla każdego modelu mikrokontrolera. Natomiast w kolumnie "DMA" znajduje się informacja o liczbie kanałów Direct Memory Access (DMA) dostępnych w mikrokontrolerach. Liczba interfejsów USART (Universal Synchronous/Asynchronous Receiver/Transmitter) dla każdego modelu jest przedstawiona w kolumnie "USART". Kolumna "SPI" zawiera informacje o liczbie interfejsów SPI (Serial Peripheral Interface) oraz ich maksymalnej prędkości transmisji danych dla każdego modelu. W kolumnie "I2C" znajduje się liczba interfejsów I2C (Inter-Integrated Circuit) dla każdego modelu mikrokontrolera. Kolumny "Wi-Fi", "Bluetooth", "Ethernet" oraz "SD Slot" informują, czy dany model mikrokontrolera obsługuje odpowiednie technologie i czy posiada gniazdo na kartę SD. W kolumnie "Power need" znajdują się informacje o zapotrzebowaniu na moc w jednostkach mikroamperów na megaherc (uA/MHz) dla poszczególnych modeli.

Przedstawiona tabela dostarcza szczegółowe informacje o różnych modelach mikrokontrolerów STM32, co pozwala na wybór odpowiedniego modelu do projektu platformy wbudowanej związanej z akwizycją. Podsumowując, płytki STM32 stanowią solidną podstawę do budowy platformy wbudowanej przeznaczonej do akwizycji danych z czujników drgań mechanicznych. Dzięki swojej wydajności, elastyczności i bogactwu interfejsów komunikacyjnych, STM32 umożliwia precyzyjną analizę i monitorowanie drgań mechanicznych oraz dostarcza niezawodne rozwiązania w obszarze akwizycji danych.

#### **4.2.3. Raspberry Pi**

Raspberry Pi jest popularną płytą rozwojową, która doskonale nadaje się jako podstawa do budowy platformy wbudowanej przeznaczonej do akwizycji danych z czujników drgań mechanicznych. Płytki Raspberry Pi oferuje szerokie możliwości i wydajność, umożliwiając rozbudowane projekty związane z analizą i monitorowaniem drgań [4.26-4.27].

Pierwszym atutem Raspberry Pi jest jego moc obliczeniowa. Wyposażony w wydajny procesor ARM, Raspberry Pi oferuje wystarczającą moc do przetwarzania danych z czujników drgań w czasie rzeczywistym. Procesor ten jest również odpowiedzialny za obsługę systemu operacyjnego, co umożliwia wykorzystanie bogatej gamy narzędzi i bibliotek programistycznych. Płytki Raspberry Pi posiada również bogate złącza i interfejsy,

które umożliwiają łatwe podłączenie i komunikację z czujnikami drgań mechanicznych. Na płycie znajdziemy interfejsy GPIO (General Purpose Input/Output), które umożliwiają podłączenie czujników analogowych i cyfrowych. Dodatkowo, Raspberry Pi posiada złącza USB, Ethernet, Wi-Fi oraz Bluetooth, co daje możliwość łatwego podłączenia innych urządzeń i komunikację z nimi.

W przypadku akwizycji danych z czujników drgań mechanicznych, Raspberry Pi oferuje wiele opcji. Można podłączyć czujniki analogowe do interfejsu ADC (Analog-to-Digital Converter) lub skorzystać z interfejsów cyfrowych, takich jak I2C czy SPI, do komunikacji z czujnikami cyfrowymi. Dzięki temu można zbierać dane o drganiach w czasie rzeczywistym i przesyłać je do platformy analizującej.

Raspberry Pi działa na systemie operacyjnym Linux, co zapewnia stabilność i elastyczność platformy. Można zainstalować różne narzędzia i biblioteki programistyczne, które ułatwiają analizę danych drgań mechanicznych. Istnieją również dedykowane systemy operacyjne i narzędzia, które dostosowane są do specjalistycznych zastosowań związanych z drganiami i monitorowaniem strukturalnym. Dodatkowo, płyta Raspberry Pi oferuje możliwość rozbudowy poprzez podłączenie modułów rozszerzeń, takich jak kamery, wyświetlacze czy karty pamięci. Daje to większe możliwości integracji i rozwinięcia funkcjonalności platformy wbudowanej.

*Tabela 4.13: Porównanie modeli Raspberry Pi dostępnych na rynku*

Model	Pi 1 B	Pi1 B+	Pi 2 B	Pi 3 B	Pi Zero
<b>CPU ARM</b>	1176JZF-S	1176JZF-S	Cortex-A7	Cortex-A53	1176JZF
<b>CPU FREQ</b>	700MHz	700MHz	900Mhz	1.2GHz	1GHz
<b>RAM</b>	512MB	512MB	1GB	1GB	512MB
<b>I/O</b>	8	17	17	17	40
<b>ADC</b>	Brak	Brak	Brak	Brak	Brak
<b>UART</b>	1	1	1	1	1
<b>SPI</b>	1	1	1	1	1
<b>I2C</b>	1	1	1	1	1
<b>Wi-Fi</b>	Brak	Brak	Brak	Tak	Brak
<b>Bluetooth</b>	Brak	Brak	Brak	Tak	Brak
<b>Ethernet</b>	Tak	Tak	Tak	Tak	Brak
<b>SD slot</b>	SD	MicroSD	MicroSD	MicroSD	MicroSD
<b>USB port</b>	2	4	4	4	1 (microUSB)
<b>Zużycie energii</b>	700mA	600mA	900mA	2.5A	160mA

W tabeli 4.13 przedstawiono wybrane modele komputerów jednopłytkowych serii Raspberry Pi wraz z ich parametrami. Model Pi 1 B jest wyposażony w procesor ARM 1176JZF-S działający z częstotliwością 700MHz. Posiada 512MB pamięci RAM, 8 portów I/O, 1 port UART, 1 port SPI, 1 port I2C oraz obsługuje karty pamięci typu SD. Nie posiada

wbudowanego modułu Wi-Fi ani Bluetooth, ale posiada port Ethernet oraz 2 porty USB. Zużycie energii wynosi 700mA. Pi 1 B+ jest ulepszoną wersją modelu Pi 1 B. Ma identyczne parametry techniczne, ale posiada większą liczbę portów I/O, które wynosi 17. Posiada również 4 porty USB oraz obsługuje karty microSD. Zużycie energii zostało zmniejszone do 600mA. Pi 2 B jest wyposażony w bardziej wydajny procesor Cortex-A7 o częstotliwości 900MHz. Posiada 1GB pamięci RAM, 17 portów I/O, 1 port UART, 1 port SPI, 1 port I2C oraz obsługuje karty microSD. Nie posiada wbudowanego modułu Wi-Fi ani Bluetooth, ale posiada port Ethernet. Dodatkowo, model Pi 2 B posiada 4 porty USB. Zużycie energii wynosi 900mA. Pi 3 B to kolejny model z serii Raspberry Pi, który posiada jeszcze wydajniejszy procesor Cortex-A53 o częstotliwości 1.2GHz. Posiada 1GB pamięci RAM, 17 portów I/O, 1 port UART, 1 port SPI, 1 port I2C oraz obsługuje karty microSD. W przeciwieństwie do poprzednich modeli, Pi 3 B posiada wbudowany moduł Wi-Fi oraz Bluetooth. Posiada również port Ethernet i 4 porty USB. Zużycie energii wynosi 2.5A. Pi Zero to model wyposażony w procesor ARM 1176JZF o częstotliwości 1GHz. Posiada 512MB pamięci RAM, a liczba portów I/O wynosi 40. Posiada 1 port UART, 1 port SPI, 1 port I2C oraz obsługuje karty microSD. Pi Zero nie posiada wbudowanych modułów Wi-Fi ani Bluetooth, ani portu Ethernet. Zamiast tego, posiada 1 port micro USB. Zużycie energii tego modelu wynosi tylko 160mA.

Tabela przedstawia różne modele płyt Raspberry Pi, które oferują różne parametry techniczne i funkcjonalności. Modele z serii B i Zero są wyposażone w procesory ARM, różniące się taktowaniem i architekturą. Posiadają one różne ilości pamięci RAM, portów I/O, portów komunikacyjnych (UART, SPI, I2C) oraz wsparcie dla Wi-Fi, Bluetooth, Ethernet i slotów kart pamięci. Zużycie energii również różni się w zależności od modelu. Wybór odpowiedniego modelu zależy od konkretnych potrzeb projektowych i wymagań dotyczących mocy obliczeniowej, interfejsów komunikacyjnych i dostępnych funkcji. Można wnioskować, że Raspberry Pi jest dobrą podstawą do budowy platformy wbudowanej do akwizycji danych z czujników drgań mechanicznych. Dzięki swojej wydajności, interfejsom komunikacyjnym oraz elastycznemu systemowi operacyjnemu, płytka ta umożliwia rozwinięcie zaawansowanych projektów monitorowania drgań i analizy danych.

#### **4.2.4. NVIDIA Jetson**

Komputer Nvidia Jetson jest zaawansowaną platformą wbudowaną, idealną do akwizycji danych z czujników drgań mechanicznych. Jetson to rodzina komputerów jednopłytkowych opartych na architekturze ARM, zaprojektowanych specjalnie do zastosowań wbudowanych i

sztucznej inteligencji. Wśród najpopularniejszych modeli znajdują się Nvidia Jetson Nano, Jetson TX2, Jetson Xavier NX i Jetson AGX Xavier. Każdy z nich oferuje różne możliwości i moc obliczeniową, co pozwala na wybór odpowiedniego modelu do aplikacji. Komputery Nvidia Jetson są wyposażone w zaawansowane procesory graficzne Nvidia CUDA, które umożliwiają przyspieszenie obliczeń i wykorzystanie algorytmów sztucznej inteligencji. Dzięki temu można wykonywać zaawansowane obliczenia, analizować dane z czujników drgań mechanicznych i stosować różne algorytmy przetwarzania sygnałów [4.28-4.30].

Nvidia Jetson posiada wystarczającą ilość portów I/O, takich jak GPIO, UART, SPI i I2C, aby podłączyć czujniki drgań mechanicznych i inne urządzenia peryferyjne. Można również korzystać z interfejsów Ethernet, USB i HDMI w celu łatwej komunikacji z innymi urządzeniami i wyjściem wideo.

Komputery Nvidia Jetson są wyposażone w wystarczającą ilość pamięci RAM i pamięci masowej, aby przechowywać dane z czujników i wykonywać skomplikowane obliczenia. Dostępne są różne modele z różnymi pojemnościami pamięci. Co więcej, Nvidia Jetson oferuje wsparcie dla różnych narzędzi i frameworków programistycznych, takich jak CUDA, PyTorch i TensorFlow, co ułatwia rozwój aplikacji i implementację algorytmów sztucznej inteligencji.

*Tabela 4.14: Porównanie komputerów NVIDIA Jetson TK1 i TX1*

	<b>TK1</b>	<b>TX1</b>
<b>CPU ARM</b>	A15 64-bit	A57 64-bit
<b>CPU FREQ</b>	933MHz	998MHz
<b>RAM</b>	2GB	4GB
<b>I/O</b>	125	70
<b>ADC</b>	Brak	Brak
<b>UART</b>	1	3
<b>SPI</b>	1	3
<b>I2C</b>	3	4
<b>Wi-Fi</b>	Brak	Tak
<b>Bluetooth</b>	Brak	Tak
<b>Ethernet</b>	Tak	Tak
<b>SD slot</b>	Tak	Tak
<b>Zużycie energii</b>	N/A	N/A

Tabela 4.14 przedstawia porównanie dwóch modeli komputerów Nvidia Jetson: TK1 i TX1. Model TK1 jest wyposażony w 64-bitowy procesor ARM A15 o taktowaniu 933MHz, 2GB pamięci RAM i 125 portów I/O. Nie posiada obsługi konwertera analogowo-cyfrowego (ADC), a także nie ma wbudowanej obsługi Wi-Fi, Bluetooth ani slotu na kartę SD. Jednak ma jeden port UART, jeden port SPI i trzy interfejsy I2C. Obsługuje również Ethernet.

Niestety, informacja dotycząca zużycia energii nie jest dostępna. Natomiast model TX1 ma 64-bitowy procesor ARM A57 o taktowaniu 998MHz, 4GB pamięci RAM i 70 portów I/O. Podobnie jak TK1, nie posiada obsługi ADC, ale ma trzy porty UART, trzy porty SPI i cztery interfejsy I2C. Dodatkowo, ma wbudowaną obsługę Wi-Fi, Bluetooth, slot na kartę SD i obsługuje Ethernet. Informacja dotycząca zużycia energii również nie jest dostępna.

Podsumowując, model TX1 oferuje wyższą wydajność i większą ilość pamięci RAM niż TK1. Posiada również większą elastyczność dzięki większej liczbie portów UART, SPI i I2C. Dodatkowo, ma wbudowaną obsługę Wi-Fi, Bluetooth i slot na kartę SD. Oba modele obsługują Ethernet. Jednak żaden z nich nie obsługuje konwertera analogowo-cyfrowego (ADC), co może wymagać dodatkowych układów lub modułów w przypadku akwizycji danych z czujników drgań mechanicznych. Wybór między TK1 a TX1 zależy od konkretnych wymagań projektowych i zastosowania platformy wbudowanej. Na podstawie powyższych można stwierdzić, że komputery Nvidia Jetson są bardzo dobrą podstawą do budowy platformy wbudowanej do akwizycji danych z czujników drgań mechanicznych. Oferują wysoką moc obliczeniową, bogate interfejsy komunikacyjne, wystarczającą ilość pamięci i wsparcie dla zaawansowanych narzędzi programistycznych.

#### 4.5 Wybór platformy wbudowanej

W procesie rozwoju systemu akwizycji danych należało rozważyć wiele różnorodnych pomysłów oraz potencjalnych, możliwych do wykonania rozwiązań tego systemu. Aby przenieść koncepcje do rzeczywistości stworzono szkice przedstawiające moduły, mikrokontrolery i połączenia między nimi. Podczas burzy mózgów brano pod uwagę układy ESP, STM, Raspberry Pi i Nvidia Jetson. Różnią się one między sobą ceną oraz możliwościami oraz zapewniają wiele różnorodnych wbudowanych oraz dodatkowych modułów interfejsów.

*Tabela 4.15: Porównanie wybranych systemów wbudowanych*

	STM32F401re	Raspberry Pi 2 B	Nvidia Jetson TK1	ESP32
<b>CPU FREQ</b>	84Mhz	900 Mhz	933 MHz	204 Mhz
<b>RAM</b>	96KB	1 GB	2 GB	520 KB
<b>ADC</b>	12-bit 2.4 Msps	Brak	Brak	8-bit 27 Ksps
<b>Wi-Fi</b>	Brak	Brak	Tak	Tak
<b>Bluetooth</b>	Brak	Brak	Tak	Tak
<b>Ethernet</b>	Brak	Brak	Tak	Brak

<b>SD-slot</b>	Brak	Tak	Tak	Brak
<b>HDMI</b>	Brak	Tak	Tak	Brak
<b>Zużycie energii</b>	146 uA/Mhz max 12 mA	900mA	N/A	95-240 uA

W procesie dobierania odpowiednich modułów, interfejsów komunikacyjnych, odpowiednich mikrokontrolerów należy wziąć pod uwagę wiele istotnych czynników, takich jak szybkość i wydajność procesora, wbudowane funkcje, możliwości rozszerzenia platformy w przyszłości. Tabela 4.15 umożliwia porównanie wszystkich rozważanych platform.

STM32F401re to mikrokontroler, który ma częstotliwość procesora wynoszącą 84 Mhz. Posiada 96 KB pamięci RAM oraz konwerter analogowo-cyfrowy (ADC) o rozdzielczości 12-bitowej i szybkości próbkowania 2.4 Msps. Nie posiada wbudowanego modułu Wi-Fi ani Bluetooth, ani portu Ethernet czy gniazda na kartę SD. Zużycie energii wynosi maksymalnie 146 uA/Mhz, a maksymalne zużycie przy pełnej pracy wynosi 12 mA.

Raspberry Pi 2B to minikomputer, który posiada wydajny procesor o częstotliwości 900 Mhz. Jest wyposażony w 1 GB pamięci RAM oraz nie posiada wbudowanego modułu ADC. Nie ma również wbudowanego modułu Wi-Fi, Bluetooth, ani portu Ethernet, ale posiada gniazdo na kartę SD. Raspberry Pi 2B posiada natomiast port HDMI, który umożliwia podłączenie do monitora lub telewizora. Zużycie energii tego minikomputera wynosi 900mA.

Nvidia Jetson TK1 to platforma wbudowana oparta na procesorze o częstotliwości 933 MHz. Posiada 2 GB pamięci RAM oraz nie posiada wbudowanego modułu ADC. Jest wyposażony w moduły Wi-Fi, Bluetooth oraz port Ethernet, co umożliwia komunikację bezprzewodową i połączenie z siecią. Jetson TK1 posiada również gniazdo na kartę SD oraz port HDMI. Zużycie energii dla tego układu nie jest dostępne.

ESP32 to mikrokontroler charakteryzujący się częstotliwością pracy wynoszącą 204 Mhz. Posiada 520 KB pamięci RAM oraz konwerter analogowo-cyfrowy (ADC) o rozdzielczości 8-bitowej i szybkości próbkowania 27 Ksps. Esp32 posiada wbudowane moduły Wi-Fi i Bluetooth, które umożliwiają bezprzewodową komunikację. Nie ma natomiast wbudowanego modułu Ethernet ani gniazda na kartę SD. Zużycie energii tego mikrokontrolera wynosi od 95 do 240 uA w zależności od częstotliwości.

W porównaniu tych układów, jeśli zależy nam na niskim zużyciu energii, mikrokontroler STM32F401re wyróżnia się najniższym zużyciem, wynoszącym maksymalnie 146 uA/Mhz. Jeśli potrzebujemy większej mocy obliczeniowej i większej pamięci RAM, Raspberry Pi 2B z

procesorem o częstotliwości 900 Mhz i 1 GB pamięci RAM będzie bardziej odpowiedni. Jeśli natomiast potrzebujemy platformy wbudowanej z wbudowanym modułem Wi-Fi, Bluetooth i portem Ethernet, Nvidia Jetson TK1 spełnia te wymagania. W przypadku ESP32, jest to mikrokontroler, który oferuje zarówno moduły Wi-Fi, Bluetooth, jak i konwerter analogowo-cyfrowy ADC, ale nie posiada modułu Ethernet. Zużycie energii tego mikrokontrolera jest również stosunkowo niskie, wynoszące od 95 do 240 uA.

ESP32 jest bardzo dobrą platformą do budowy systemu akwizycji drgań mechanicznych z kilku powodów. Przede wszystkim, istotną cechą ESP32 jest jego niskie zużycie energii, co przekłada się na możliwość długiego czasu pracy na baterii. Dzięki temu, system akwizycji drgań oparty na ESP32 może działać przez długi czas bez konieczności częstej wymiany baterii lub ładowania. Kolejną zaletą ESP32 jest obecność dwóch rdzeni, co umożliwia wykonywanie operacji równoległe. Dzięki temu, system może jednocześnie zajmować się akwizycją danych z czujników drgań mechanicznych, przetwarzaniem tych danych i komunikacją bezprzewodową. Wykorzystanie obu rdzeni daje możliwość efektywnego wykorzystania zasobów sprzętowych i zapewnia płynne działanie systemu. ESP32 posiada wbudowane moduły Wi-Fi i Bluetooth, co stanowi istotną zaletę w przypadku systemu akwizycji drgań. Dzięki modułowi Wi-Fi, urządzenie może bezprzewodowo komunikować się z innymi urządzeniami, przysyłać dane do chmury lub odbierać instrukcje zdalnie. Moduł Bluetooth umożliwia natomiast łatwe połączenie z innymi urządzeniami Bluetooth, takimi jak smartfony czy tablety, co ułatwia konfigurację i kontrolę systemu akwizycji. ESP32 posiada również wbudowany konwerter analogowo-cyfrowy (ADC), który jest istotny w przypadku akwizycji danych z czujników drgań mechanicznych. Chociaż ADC w ESP32 jest 8-bitowy, co oznacza niższą rozdzielczość w porównaniu do 12-bitowego ADC w niektórych innych platformach, można zastosować dodatkowe techniki i algorytmy w celu poprawy precyzji pomiarów.

STM32 bazujące na architekturze Cortex-M jest jedną z najlepszych platform do budowy systemu akwizycji drgań mechanicznych z wielu powodów. Przede wszystkim, charakteryzuje się małym zużyciem energii, co ma kluczowe znaczenie dla systemów pracujących na baterii. Dzięki temu można osiągnąć znacznie dłuższy czas pracy urządzenia, co jest istotne zwłaszcza w przypadku aplikacji, w których trudno jest dostarczyć stałe zasilanie. Kolejnym atutem platformy STM32 jest dynamiczne zarządzanie prędkością procesora. Oznacza to, że procesor może dostosowywać swoją moc obliczeniową w zależności od aktualnego obciążenia. W sytuacjach, gdy wymagane są mniejsze obliczenia,

procesor może pracować w trybie niskiego poboru mocy, co przekłada się na jeszcze większą efektywność energetyczną systemu. Dzięki temu można zoptymalizować zużycie energii i przedłużyć czas działania na baterii. Ważnym elementem platformy STM32 jest również dostępność zaawansowanego przetwornika analogowo-cyfrowego (ADC) o rozdzielczości aż 12-bitowej. Ten wysokopoziomowy przetwornik pozwala na dokładny pomiar drgań mechanicznych z dużo większą precyzją niż w przypadku konkurencyjnych platform. Dzięki temu można uzyskać bardziej wiarygodne i precyzyjne dane diagnostyczne, co ma kluczowe znaczenie przy monitorowaniu stanu technicznego maszyn. Dodatkowo, platforma STM32 oferuje szeroki zakres narzędzi i bibliotek programistycznych, które ułatwiają rozwój systemu akwizycji drgań mechanicznych. Wspierane przez środowisko programistyczne, można tworzyć aplikacje w oparciu o popularne języki programowania, takie jak C/C++.

#### 4.6 Wybór akcelerometru

Akcelerometry są elektronicznymi urządzeniami, które służą do pomiaru przyspieszenia, nachylenia, wibracji lub wstrząsów. Są one niezwykle wszechstronne i znajdują zastosowanie w różnych dziedzinach, od urządzeń fitness po systemy stabilizacji platform przemysłowych. Oferują one możliwość precyzyjnego pomiaru i monitorowania ruchu oraz zmiany pozycji obiektów. Historia akcelerometrów sięga lat 70. XX wieku, kiedy to pierwsze prototypy tych urządzeń zostały opracowane przez NASA w celu monitorowania ruchów statków kosmicznych. Początkowo akcelerometry były stosowane głównie w przemyśle kosmicznym i lotniczym, jednak wraz z postępem technologicznym ich zastosowania poszerzyły się.

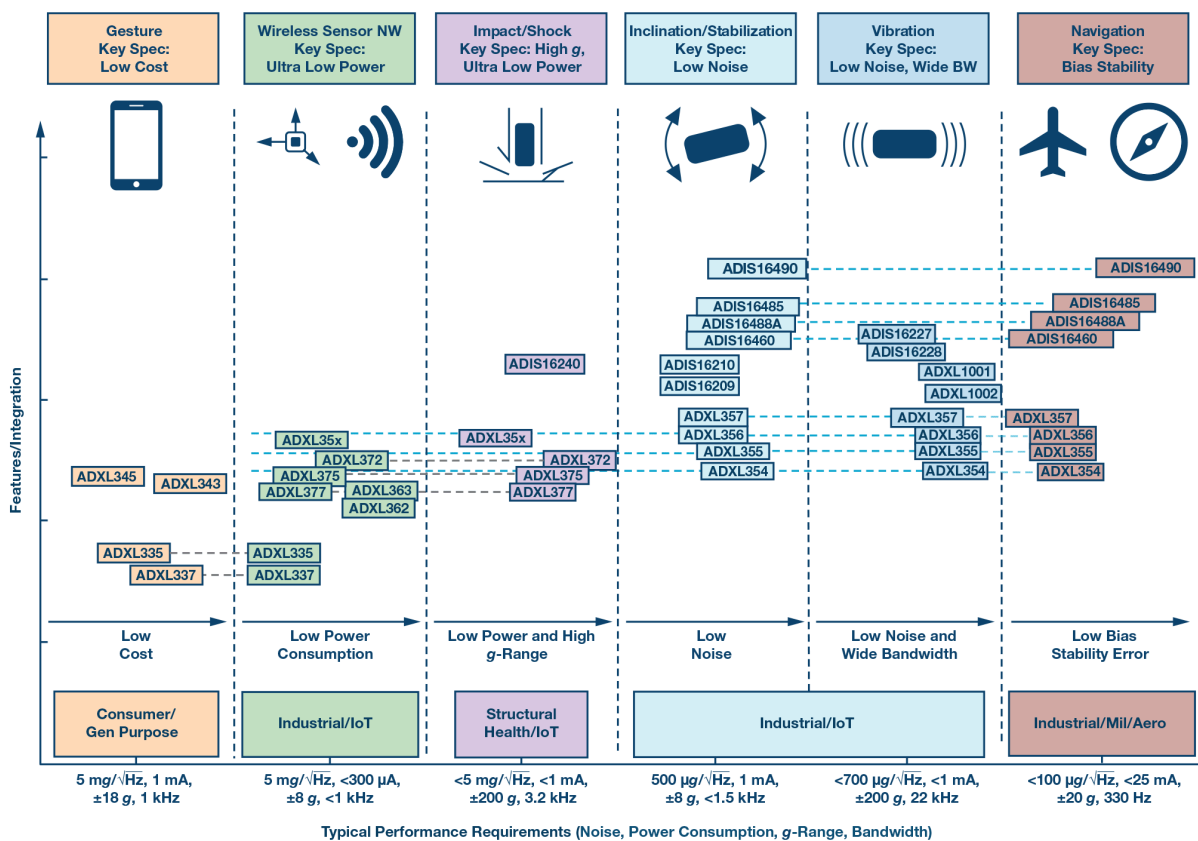
Obecnie na rynku istnieje wiele różnych akcelerometrów, oferujących różne parametry i cechy, aby sprostać różnorodnym potrzebom aplikacji. Istnieje szeroki wybór akcelerometrów o różnej wydajności i kosztach, co umożliwia projektantom wybór najlepszego rozwiązania dla swojego konkretnego projektu.

*Tabela 4.16: Klasa akcelerometru i typowy obszar zastosowania*

	<b>Główne zastosowanie</b>	<b>Pasma</b>	<b>Zakres</b>
<b>Konsument</b>	Wykrycie ruchu, przyspieszenie	0 Hz	1 g
<b>Motoryzacja</b>	Zderzenie, stabilność	100 Hz	< 200 g
<b>Przemysł</b>	Przechył, stabilność	5 – 500 Hz	25 g
<b>Wojsko</b>	Produkcja broni	< 1kHz	8 g
<b>Nawigacja</b>	Nawigacja pojazdów (samochody, łodzie, samoloty)	>300 Hz	15 g

Tabela 4.16 przedstawia wykorzystanie akcelerometrów podzielone na obszary w których znajdują one zastosowanie uwzględniając pasmo ich pracy oraz zakres pomiaru wyrażony w g. W sektorze konsumentów, akcelerometry są powszechnie stosowane do wykrywania ruchu oraz pomiaru przyspieszenia. Ich zakres pomiarowy wynosi od 0 Hz do 1 g, co pozwala na precyzyjne monitorowanie nawet niewielkich ruchów. W motoryzacji, akcelerometry są używane do wykrywania zderzeń oraz monitorowania stabilności pojazdów. Pasma pomiarowe w tym przypadku wynosi od 100 Hz do mniej niż 200 g, co umożliwia precyzyjne rejestrowanie siły działającej na pojazd w przypadku kolizji. W przemyśle, akcelerometry znajdują zastosowanie w pomiarze przechyłu oraz monitorowaniu stabilności różnego rodzaju platform przemysłowych. Zakres pomiarowy w tym sektorze to 5 Hz - 500 Hz przy maksymalnym przyspieszeniu do 25 g, co pozwala na precyzyjne śledzenie i kontrolę ruchów w środowisku przemysłowym. W dziedzinie wojskowej, akcelerometry są używane w produkcji broni. Pasma pomiarowe dla tych aplikacji to mniej niż 1 kHz, a maksymalne przyspieszenie wynosi 8 g. Akcelerometry zapewniają niezbędną precyzję i pomiar sił działających podczas wystrzałów i manewrów wojskowych. W dziedzinie nawigacji, akcelerometry są wykorzystywane w pojazdach takich jak samochody, łodzie i samoloty. W tym przypadku, pasmo pomiarowe przekracza 300 Hz, a maksymalne przyspieszenie wynosi 15 g. Akcelerometry umożliwiają precyzyjne monitorowanie ruchów pojazdów i dostarczają istotnych danych do systemów nawigacyjnych. Podsumowując, akcelerometry znajdują zastosowanie w różnych sektorach, w tym w sektorze konsumentów, motoryzacji, przemyśle, wojsku i nawigacji. Ich pasmo i zakres pomiarowy są dostosowane do specyficznych wymagań i zastosowań w tych obszarach, umożliwiając precyzyjne monitorowanie ruchu, przyspieszenia, przechyłu i stabilności w zależności od konkretnych potrzeb.

Ważne parametry i cechy, które powinien brać pod uwagę projektant, to zakres pomiarowy, rozdzielczość, dokładność, przetwarzanie danych, interfejs komunikacyjny, zużycie energii i wielkość. Zakres pomiarowy odnosi się do maksymalnego przyspieszenia, które może zostać zmierzone przez akcelerometr. Rozdzielczość odnosi się do najmniejszej zmiany przyspieszenia, którą urządzenie jest w stanie wykryć. Dokładność odnosi się do miary, w jakiej pomiar akcelerometru jest zgodny z rzeczywistym przyspieszeniem. Przetwarzanie danych odnosi się do sposobu, w jaki akcelerometr przekazuje dane pomiarowe do systemu. Interfejs komunikacyjny określa sposób, w jaki akcelerometr komunikuje się z innymi urządzeniami. Zużycie energii jest istotne, szczególnie w przypadku urządzeń zasilanych bateryjnie. Wielkość odnosi się do fizycznego rozmiaru akcelerometru, który może być ważny w przypadku aplikacji o ograniczonej przestrzeni.



Rysunek 4.8: Przegląd zastosowań dla wybranych akcelerometrów MEMS firmy Analog Devices Inc. [<https://www.analog.com/en/analog-dialogue/articles/choosing-the-most-suitable-mems-accelerometer-for-your-application-part-1.html>]

Rysunek 4.8 przedstawia migawkę szeregu akcelerometrów MEMS, prezentując różne czujniki i klasyfikując je na podstawie kluczowych wskaźników wydajności oraz poziomu inteligencji i integracji, dostosowanych do konkretnych zastosowań. Rysunek skupia się na akcelerometrach nowej generacji, które wykorzystują ulepszone struktury MEMS i zaawansowane techniki przetwarzania sygnałów. Te nowoczesne czujniki oferują stabilność i poziom szumów porównywalny z droższymi urządzeniami specjalistycznymi, przy jednoczesnym mniejszym zużyciu energii. Jest to szczególnie istotne zwłaszcza dla zastosowań, gdzie długotrwała praca na baterii jest wymagana, zapewniając wydłużony czas pracy urządzenia. Można wnioskować, że akcelerometry nowej generacji oparte na ulepszonych strukturach MEMS i technikach przetwarzania sygnałów stanowią obiecujące rozwiązanie dla szerokiej gamy zastosowań. Dostarczają one stabilności, precyzji i wydajności, przy jednoczesnym mniejszym zużyciu energii, co sprawia, że są atrakcyjnym wyborem dla projektantów i inżynierów.

Do budowy nowoczesnego systemu akwizycji drgań mechanicznych wybrano akcelerometry serii ADXL30x - wieloosiowe oraz akcelerometry z rodziny ADXL100x - jednoosiowe, o znacznie większym zakresie i szerszym paśmie. Wybór tych konkretnych modeli akcelerometrów jest podyktowany kilkoma istotnymi czynnikami. Pierwszym z tych czynników jest wieloosiowość akcelerometrów serii ADXL30x. Posiadanie wielu osi pomiarowych umożliwia rejestrowanie drgań w różnych kierunkach, co jest istotne w przypadku złożonych maszyn i struktur, gdzie drgania mogą występować w różnych płaszczyznach. Wieloosiowe akcelerometry ADXL30x pozwalają na precyzyjne monitorowanie drgań w trzech wymiarach, co jest niezwykle przydatne przy analizie i diagnozowaniu stanu technicznego maszyn. Z kolei akcelerometry z rodziny ADXL100x charakteryzują się jednoosiowością, ale oferują znacznie większy zakres i szersze pasmo. Dzięki temu są w stanie rejestrować drgania o większej amplitudzie i wyższej częstotliwości, co jest istotne w przypadku maszyn generujących intensywne drgania. Takie akcelerometry umożliwiają bardziej precyzyjne pomiar i analizę dynamicznych zjawisk mechanicznych, które mogą wpływać na wydajność i bezpieczeństwo maszyny. Wybór akcelerometrów serii ADXL30x i ADXL100x jest także wynikiem ich renomy i wysokiej jakości. Te modele zostały opracowane przez renomowanego producenta, który jest znany z dostarczania precyzyjnych i niezawodnych czujników pomiarowych. Wreszcie, zastosowanie akcelerometrów serii ADXL30x i ADXL100x pozwala na integrację z innymi elementami systemu akwizycji danych. Dzięki temu możliwa jest kompleksowa analiza i prezentacja danych pomiarowych.

Podsumowując, wybór akcelerometrów serii ADXL30x i ADXL100x do budowy nowoczesnego systemu akwizycji drgań mechanicznych jest uzasadniony ich wieloosiowością, większym zakresem, szerszym pasmem i wysoką jakością. Dzięki tym cechom system jest w stanie dostarczać precyzyjne i kompleksowe dane, umożliwiające skuteczną analizę i monitorowanie stanu technicznego maszyn.

## 5 Analiza właściwości mechanicznych toru przenoszenia sygnału drgań

### 5.1 Akcelerometry wykonane w technologii MEMS

Akcelerometry wykonane w technologii MEMS (Micro-Electro-Mechanical Systems) stanowią innowacyjne rozwiązanie w dziedzinie czujników pomiarowych. Te małe, precyzyjne urządzenia pomiarowe są szeroko stosowane w różnych dziedzinach, takich jak elektronika konsumencka, przemysł, motoryzacja, medycyna i wiele innych. Wykorzystując zaawansowane technologie mikroelektroniki i mikro mechaniki, akcelerometry MEMS oferują wiele korzyści i funkcjonalności. Jedną z głównych zalet akcelerometrów MEMS jest ich kompaktowa budowa. Są one wykonane w formie małych układów scalonych, co pozwala na łatwą integrację z innymi urządzeniami elektronicznymi. Dzięki temu akcelerometry MEMS można z powodzeniem zastosować w kompaktowych i przenośnych urządzeniach, takich jak smartfony, tablety czy zegarki inteligentne. Kolejną ważną cechą akcelerometrów MEMS jest ich wysoka precyzja i dokładność pomiaru. Dzięki zastosowaniu mikro mechanicznych struktur i zaawansowanych algorytmów kompensacyjnych, te czujniki są w stanie rejestrować nawet najmniejsze zmiany przyspieszenia. Dzięki temu możliwe jest precyzyjne monitorowanie ruchu, położenia i drgań w różnych aplikacjach. Akcelerometry MEMS charakteryzują się również szerokim zakresem pomiarowym. Mogą rejestrować przyspieszenie w różnych zakresach, od niskich wartości do bardzo wysokich. Dzięki temu są one elastycznym narzędziem, które można dostosować do konkretnych potrzeb pomiarowych w zależności od specyfiki aplikacji. Kolejną zaletą akcelerometrów MEMS jest ich niskie zużycie energii. Dzięki zastosowaniu zaawansowanych technik oszczędzania energii, te czujniki są wyjątkowo energooszczędne. Można je z powodzeniem stosować w urządzeniach zasilanych bateryjnie, gdzie długi czas pracy jest istotny.

Akcelerometry MEMS znajdują zastosowanie w wielu dziedzinach. W elektronice konsumenckiej są wykorzystywane do funkcji takich jak detekcja ruchu, stabilizacja obrazu czy gestykulacja. W przemyśle są używane do monitorowania drgań maszyn, diagnostyki uszkodzeń czy kontroli jakości. W motoryzacji znajdują zastosowanie w systemach kontroli stabilności, nawigacji i aktywnego zawieszenia. W medycynie są używane w aparatach ortopedycznych, systemach monitorowania pacjentów i wielu innych obszarach.

Obecnie techniki symulacyjne stanowią integralną część procesu projektowania i technologii wytwarzania. Obecna w przemyśle tendencja wymusza produkcję lżejszych, trwalszych, łatwiejszych w produkcji i tańszych urządzeń. Właśnie dlatego Analiza

Elementów Skończonych staje się kluczową metodą symulacji. Ideą MES jest rozbicie dużej struktury o wysokim stopniu złożoności na mniejsze i łatwiejsze w zarządzaniu sekcje, zwane elementami. Każdy element reprezentuje właściwości mechaniczne swojej domeny lokalnej. Dzieląc rozważaną strukturę na coraz mniejsze fragmenty, można zrozumieć, w jaki sposób większa struktura będzie reagowała na stymulację zewnętrzną lub wewnętrzną [5.1].

Technologia systemów mikro elektromechanicznych (MEMS) jest niezwykle wszechstronna i znajduje szerokie zastosowanie w produkcji różnorodnych mikro komponentów. Jest to technologia, która integruje elementy mikroskalowe, takie jak mikroczujniki, mikro siłowniki czy mikro filtry pasmowo przepustowe, w jednym układzie. Urządzenia oparte na MEMS są preferowane w projektowaniu czujników ze względu na ich małe rozmiary, minimalnie inwazyjną implantację, kompatybilność materiałową i niski pobór mocy [5.2]. Pierwszym aspektem, który przyciąga uwagę projektantów, jest niewątpliwie mały rozmiar urządzeń MEMS. Dzięki wykorzystaniu mikroskali, możliwe jest stworzenie kompaktowych czujników, które zajmują minimalną przestrzeń. Ta cecha jest niezwykle cenna, zwłaszcza w przypadku aplikacji, gdzie dostępna przestrzeń jest ograniczona. Czujniki MEMS mogą być łatwo zintegrowane w urządzeniach o mniejszych gabarytach, takich jak smartfony, smartwatche czy implanty medyczne. Kolejnym atutem technologii MEMS jest minimalnie inwazyjna implantacja. Oznacza to, że urządzenia oparte na tej technologii mogą być łatwo wprowadzane do środowiska, bez konieczności znacznego ingerowania w istniejącą strukturę. Przykładem mogą być mikroczujniki stosowane w medycynie, które mogą być implantowane w organizmach pacjentów, minimalizując ryzyko powikłań i przyspieszając proces gojenia. Kompatybilność materiałowa jest kolejnym ważnym aspektem czujników MEMS. Dzięki technologii MEMS możliwe jest wykonanie czujników z różnorodnych materiałów, co pozwala na dostosowanie ich do konkretnych zastosowań. Na przykład, w przypadku czujników ciśnienia, można użyć membrany wykonanej z materiału o odpowiedniej elastyczności i odporności na oddziaływanie czynników zewnętrznych. Niski pobór mocy jest kolejną zaletą urządzeń MEMS. Ze względu na ich mały rozmiar i zoptymalizowane struktury, czujniki MEMS zużywają znacznie mniej energii niż tradycyjne urządzenia. Jest to szczególnie ważne w przypadku urządzeń mobilnych, które działają na bateriach. Dzięki niskiemu poborowi mocy, urządzenia te mogą działać dłużej na jednym ładowaniu.

Nowoczesne systemy monitorowania stanu, oparte na czujnikach MEMS, mają potencjał do znacznego obniżenia kosztów instalacji na każdy węzeł. Przeprowadzone badania

wykazały, że dzięki wykorzystaniu czujników MEMS, koszt instalacji może spaść z 1000 do 100 USD na pojedynczy węzeł [5.3]. Ta znacząca redukcja kosztów jest niezwykle atrakcyjna dla różnych sektorów i branż, które wykorzystują systemy monitorowania stanu, takie jak przemysł, energetyka czy transport. Jednak korzyści płynące z czujników MEMS nie kończą się na aspekcie finansowym [5.4]. Cechują się one również niezwyklejmi atrybutami, które wpływają na jakość i wydajność systemów monitorowania. Pierwszym z tych atrybutów jest rozmiar. Czujniki MEMS charakteryzują się niewielkimi gabarytami, co pozwala na ich łatwą integrację w różnych aplikacjach. Dzięki temu można je bezproblemowo umieszczać w ograniczonej przestrzeni, a nawet integrować z innymi urządzeniami. Kolejnym istotnym atrybutem jest waga czujników MEMS. Ich lekkość sprawia, że są one idealne do zastosowań, gdzie waga ma kluczowe znaczenie, np. w branży lotniczej czy kosmicznej. Redukcja masy czujników ma bezpośredni wpływ na osiągi systemów monitorowania, umożliwiając bardziej precyzyjne i efektywne działanie. Niski pobór mocy to kolejny atut czujników MEMS. Dzięki zoptymalizowanym strukturom i wykorzystaniu nowoczesnych technologii, czujniki te zużywają minimalną ilość energii. Oznacza to, że mogą działać przez dłuższy czas na jednym ładowaniu lub bateryjce, co jest szczególnie istotne w aplikacjach, gdzie dostęp do zasilania jest utrudniony. Ostatnim, ale równie istotnym atrybutem czujników MEMS jest ich wysoki poziom funkcjonalności. Dzięki swojej konstrukcji i wykorzystaniu zaawansowanych technologii, czujniki te mogą mieć różnorodne interfejsy i możliwość łączenia się z innymi urządzeniami. Daje to projektantom i inżynierom większą swobodę w projektowaniu systemów monitorowania, umożliwiając bardziej elastyczną i kompleksową integrację.

W tej chwili można powiedzieć, że ludzkość jest w erze MEMS, w której, po fazie napędzanej głównie miniaturyzacją systemów, rozszerzone możliwości są dodawane do mikrosystemów; na przykład wieloosiowe systemy pomiarowe łączące akcelerometry, żyroskopy, czujniki ciśnienia i pola magnetycznego. W niedalekiej przyszłości możemy być świadkami nowej fazy; w którym mikrosystemy zostaną włączone do bezprzewodowych sieci czujników (Internet of Things) i będą miały dodatkowe możliwości i wydajności, takie jak autonomia energetyczna i wysoka niezawodność, nawet w ekstremalnie agresywnych środowiskach. Po okresie, w którym mikrosystemy były przedmiotem badań w laboratoriach akademickich, na rynku zaczęła pojawiać się branża MEMS. W ostatnim czasie produkcja MEMS rozszerzyła się, głównie ze względu na zapotrzebowanie na inercyjne MEMS (mikroakcelerometry i mikrożyroskopy) w produktach na rynku konsumenckim (głównie urządzenia inteligentne, jak smartfony) [5.5-5.8].

## 5.2 Procedura analizy modalnej

Analiza modalna jest kompleksowym procesem, który pozwala wyodrębnić istotne parametry modalne z danych dotyczących drgań. Istnieją dwie główne metody analizy modalnej: analiza modalna w dziedzinie częstotliwości oraz analiza modalna w dziedzinie czasu. Oba podejścia mają na celu uzyskanie informacji na temat częstotliwości drgań własnych, współczynników tłumienia i stałych modalnych. W przypadku analizy modalnej w dziedzinie częstotliwości, wykorzystuje się funkcje odpowiedzi częstotliwościowej (FRF), które są mierzonymi danymi. Proces polega na dopasowaniu krzywej do tych danych przy użyciu wcześniej ustalonego modelu matematycznego struktury poddawanej analizie. Model ten uwzględnia liczbę stopni swobody konstrukcji, rodzaj tłumienia oraz ewentualną liczbę postaci drgań w badanym zakresie częstotliwości. Na podstawie tych założeń można matematycznie wyrazić każdą krzywą FRF uzyskaną z pomiarów. Następnie, poprzez proces dopasowywania krzywej do danych pomiarowych, można wyprowadzić parametry modalne zawarte we wzorze matematycznym FRF. W przypadku analizy modalnej w dziedzinie czasu, wykorzystuje się odpowiedzi impulsowe obiektu poddawanego badaniom. Proces analizy polega na identyfikacji i wyodrębnieniu drgań własnych na podstawie odpowiedzi impulsowej. Za pomocą zaawansowanych technik matematycznych i statystycznych, takich jak transformata Fouriera czy metoda najmniejszych kwadratów, można obliczyć częstotliwość drgań własnych, współczynniki tłumienia oraz stałe modalne. Analiza modalna jest niezwykle ważna w dziedzinach inżynierii, takich jak budowa mostów, pojazdów, maszyn czy konstrukcji lotniczych. Pozwala ona na ocenę dynamicznych charakterystyk struktur, identyfikację problemów związanych z drganiami oraz optymalizację projektu. Dzięki analizie modalnej można uniknąć niebezpiecznych rezonansów, zoptymalizować działanie konstrukcji oraz poprawić jej wytrzymałość i trwałość.

Dokładność analizy modalnej jest kwestią wymagającą uwagi, ponieważ dopasowanie krzywej FRF do danych pomiarowych ma charakter matematyczny. Im bardziej precyzyjne są pomiary FRF, tym większe szanse na dokładniejsze dopasowanie krzywej do danych. W matematyce ocenę dokładności i skuteczności dopasowania krzywej można przeprowadzić poprzez definiowanie funkcji błędu i dążenie do minimalizacji tego błędu. Jednak podejście to jest poprawne tylko wtedy, gdy używany model matematyczny jest właściwy. Jeśli zostanie zastosowany nieprawidłowy model matematyczny, dopasowanie krzywej może zakończyć się nieprawidłowymi wynikami lub nawet całkowitym niepowodzeniem, nawet jeśli numerycznie minimalizuje się funkcję błędu. Ważne jest, aby dobrze zrozumieć charakterystykę badanej struktury i wybrać odpowiedni model matematyczny do analizy modalnej. Niepoprawne lub

niewłaściwie dobranie modelu może prowadzić do błędnych wniosków i niewłaściwych interpretacji wyników. Dlatego kluczowe jest posiadanie odpowiedniej wiedzy i doświadczenia w zakresie analizy modalnej oraz umiejętność wyboru właściwego modelu matematycznego. Dodatkowym czynnikiem wpływającym na dokładność analizy modalnej jest jakość danych pomiarowych. Dobrej jakości pomiary FRF, o wysokiej rozdzielczości i niskim poziomie zakłóceń, są istotne dla uzyskania dokładnych wyników analizy modalnej. W przypadku niskiej jakości danych pomiarowych, mogą wystąpić trudności w dopasowaniu krzywej FRF, co prowadzi do mniejszej dokładności parametrów modalnych. Ponadto, ważne jest uwzględnienie warunków granicznych i asumptów przy tworzeniu modelu matematycznego. Rzeczywiste struktury często są narażone na różne czynniki zewnętrzne, takie jak temperatury, obciążenia czy siły związane z ruchem. Właściwe uwzględnienie tych czynników w modelu matematycznym pozwala na bardziej realistyczną analizę modalną i dokładniejsze odzwierciedlenie zachowania struktury w rzeczywistych warunkach.

W przypadku analizy modalnej w dziedzinie częstotliwości, model matematyczny jest analitycznym wyrażeniem funkcji odpowiedzi częstotliwościowej, które jest rzeczywiście reprezentatywne dla danych FRF uzyskanych z pomiarów. Jednak pojawia się kilka istotnych pytań dotyczących tego modelu. Po pierwsze, struktura analizowana za pomocą analizy modalnej ma zazwyczaj nieskończoną liczbę stopni swobody. W rzeczywistości, zmierzone dane FRF zawierają informacje o modalnych trybach drgań zarówno w zakresie częstotliwości pomiaru, jak i poza nim. W celu uproszczenia analizy, można przyjąć, że strukturę można dyskretyzować jako układ składający się z „N” stopni swobody. Kolejnym pytaniem dotyczącym modelu matematycznego jest wybór odpowiedniego zbioru funkcji bazowych. W przypadku analizy modalnej, popularnym podejściem jest wykorzystanie funkcji sinusoidalnych, takich jak funkcje trygonometryczne jako bazy do reprezentacji modalnych drgań struktury. Wybór takiej bazy jest związany z teorią rozwoju w szereg Fouriera, która umożliwia rozbicie dowolnej funkcji na sumę funkcji trygonometrycznych o różnych częstotliwościach. Kolejne pytanie dotyczy właściwości tłumienia w modelu matematycznym. Tłumienie jest istotnym aspektem analizy modalnej, ponieważ wpływa na trwałość, stabilność i dynamiczne zachowanie struktury. Istnieje kilka sposobów modelowania tłumienia w analizie modalnej, takich jak tłumienie proporcjonalne czy tłumienie nieliniowe. Wybór odpowiedniej metody modelowania tłumienia jest zależny od charakterystyki badanej struktury i wymaga starannego rozważenia. Ponadto, model matematyczny powinien uwzględniać różne warunki graniczne, takie jak siły zewnętrzne, obciążenia, temperatury czy warunki brzegowe. Te czynniki mogą mieć istotny wpływ na

zachowanie struktury i jej modalne parametry. Dlatego ważne jest, aby model matematyczny był odporny na wpływ tych czynników i był w stanie odzwierciedlić rzeczywiste warunki działania struktury.

Test modalny i analiza modalna służą do określenia parametrów modalnych konstrukcji inżynierskich, takich jak częstotliwości rezonansowe, współczynniki tłumienia i kształty modów. Zmierzone dane pobudzenia i odpowiedzi (lub tylko odpowiedzi) są wykorzystywane w analizie modalnej, a następnie przetwarzana jest dynamiczna analiza sygnału i identyfikacja parametrów modalnych.

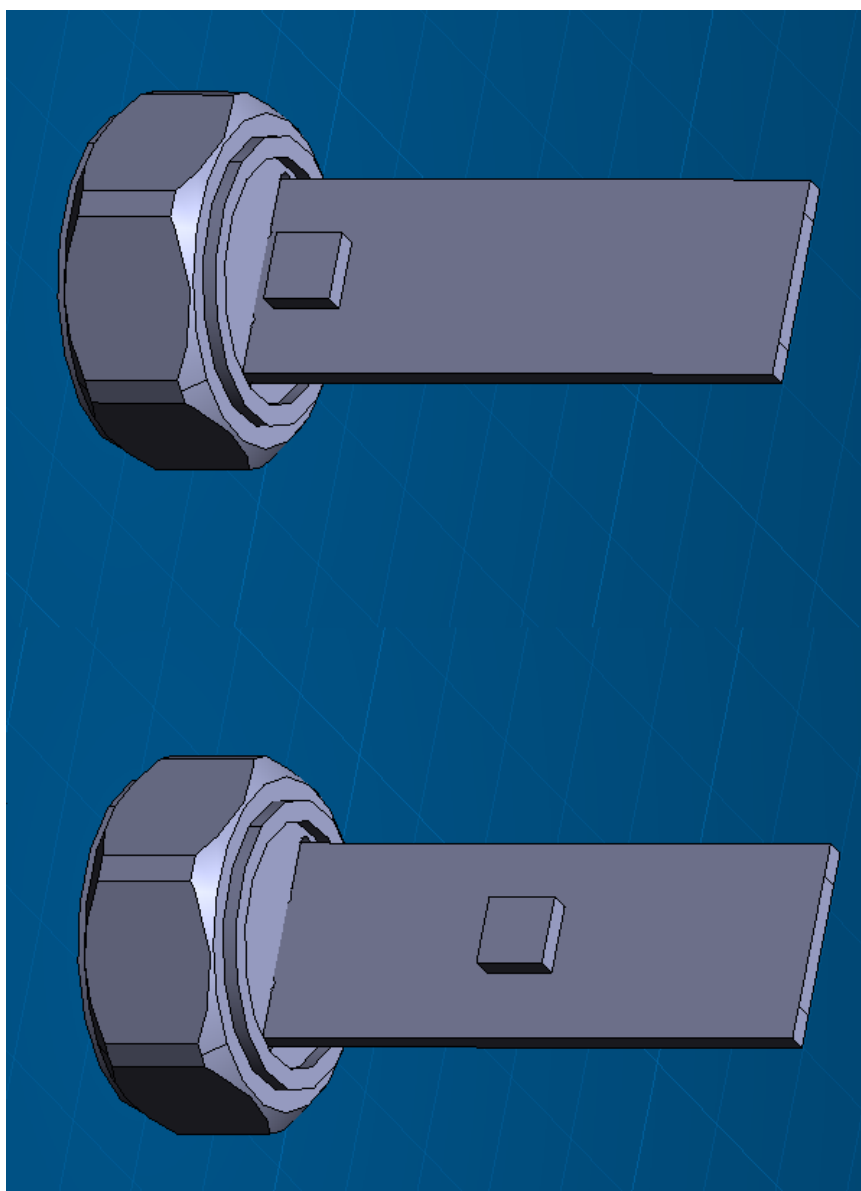
#### *5.2.1.1 Co to jest funkcja odpowiedzi częstotliwościowej?*

Funkcja odpowiedzi częstotliwościowej  $H(f)$  w dziedzinie częstotliwości oraz funkcja odpowiedzi impulsowej  $h(t)$  w dziedzinie czasu pełnią istotną rolę w opisie relacji wejście-wyście (siła-odpowiedź) dowolnego systemu, w którym sygnały  $a(t)$  i  $b(t)$  reprezentują wejście i wyjście tego układu fizycznego. W kontekście tych funkcji przyjmuje się, że analizowany system jest liniowy i niezmienny w czasie. Funkcja odpowiedzi częstotliwościowej oraz funkcja odpowiedzi impulsowej stanowią tzw. deskryptory systemu, które nie zależą od konkretnej postaci sygnałów zaangażowanych w analizę. Funkcja odpowiedzi częstotliwościowej  $H(f)$  jest szczególnie przydatna przy analizie systemów w dziedzinie częstotliwości. Przedstawia ona zależność pomiędzy amplitudą i fazą sygnału wyjściowego, a częstotliwością sygnału wejściowego. Innymi słowy, funkcja ta pozwala określić, jak układ reaguje na różne składowe częstotliwościowe sygnału wejściowego. Dzięki temu możliwe jest badanie charakterystyki systemu w różnych zakresach częstotliwości, co pozwala na identyfikację pasm częstotliwościowych, w których układ wykazuje specyficzne właściwości, takie jak wzmocnienie, tłumienie lub opóźnienie. Z kolei funkcja odpowiedzi impulsowej  $h(t)$  jest używana do analizy systemów w dziedzinie czasu. Przedstawia ona sposób, w jaki układ reaguje na impulsowy sygnał wejściowy w ciągu czasu. Odpowiedź impulsowa układu zawiera informacje o jego dynamicznym zachowaniu, takie jak czas narastania, czas opadania, oscylacje czy stabilność. Dzięki analizie funkcji odpowiedzi impulsowej możliwe jest zrozumienie reakcji systemu na różne rodzaje sygnałów wejściowych oraz identyfikacja charakterystycznych cech, które mogą wpływać na jego działanie. Warto zauważyć, że zarówno funkcja odpowiedzi częstotliwościowej, jak i funkcja odpowiedzi impulsowej są niezależne od konkretnych sygnałów wejściowych. Oznacza to, że opisują one zachowanie układu jako całości, niezależnie od konkretnej postaci sygnałów

zaangażowanych w analizę. Dlatego są one ważnym narzędziem w inżynierii systemów, umożliwiającym badanie właściwości systemów liniowych i niezmiennych w czasie.

### 5.3 Badanie wpływu lokalizacji układu scalonego

Lokalizacja czujnika i jego instalacja to dwie najważniejsze części zbierania danych o drganiach, dlatego punkt akwizycji musi być wybrany z uwzględnieniem pewnych kryteriów. To miejsce w maszynie nazywa się punktem czytania. Maszyny wirujące generują drgania za pomocą cyklicznych sił wewnętrznych, których nie można bezpośrednio zmierzyć. Procedury diagnostyczne opierają się na pomiarze przeniesienia tych sił przez konstrukcję maszyny (np. jej podwozie) i dlatego czujnik drgań mierzy odpowiedź tych elementów. Głównym powodem właściwego doboru punktu odczytu jest zminimalizowanie skutków tego mechanicznego wzbudzenia (zarówno jego części jak i całej konstrukcji), gdyż interesuje nas pomiar wewnętrznych sił cyklicznych. Dlatego ważne jest również, aby wziąć pod uwagę umiejscowienie akcelerometru MEMS wewnątrz czujnika przyspieszenia, aby zminimalizować niepożądane efekty. Na rysunku 5.1 przedstawiono proponowane uproszczone konstrukcje akcelerometru z czujnikiem MEMS umieszczonym w różnych częściach płytki PCB.



*Rysunek 5.1: Rozważane położenie czujnika MEMS na płytce PCB akcelerometru*

W kolejnym etapie przeprowadzono analizę metody elementów skończonych (MES) w celu symulacji wzbudzenia płytki czujnikowej pod obciążeniem o wartości 1 [g] w kierunku Z, czyli osi pomiaru akcelerometru MEMS. W celu uwzględnienia liniowego zakresu odpowiedzi częstotliwościowej czujnika do 11 kHz, analizę przeprowadzono w zakresie częstotliwości od 100 Hz do 10 kHz. Wyniki tej analizy zostały zestawione w tabeli 5.1, gdzie przedstawiono symulowane częstotliwości drgań własnych z analizy modalnej oraz częstotliwości rezonansowe z analizy harmonicznej. Analiza MES stanowi narzędzie numeryczne wykorzystywane do modelowania i symulacji zachowania struktur mechanicznych. Wykorzystuje ona technikę podziału analizowanej struktury na skończoną liczbę elementów, które są połączone ze sobą na podstawie warunków brzegowych i równań

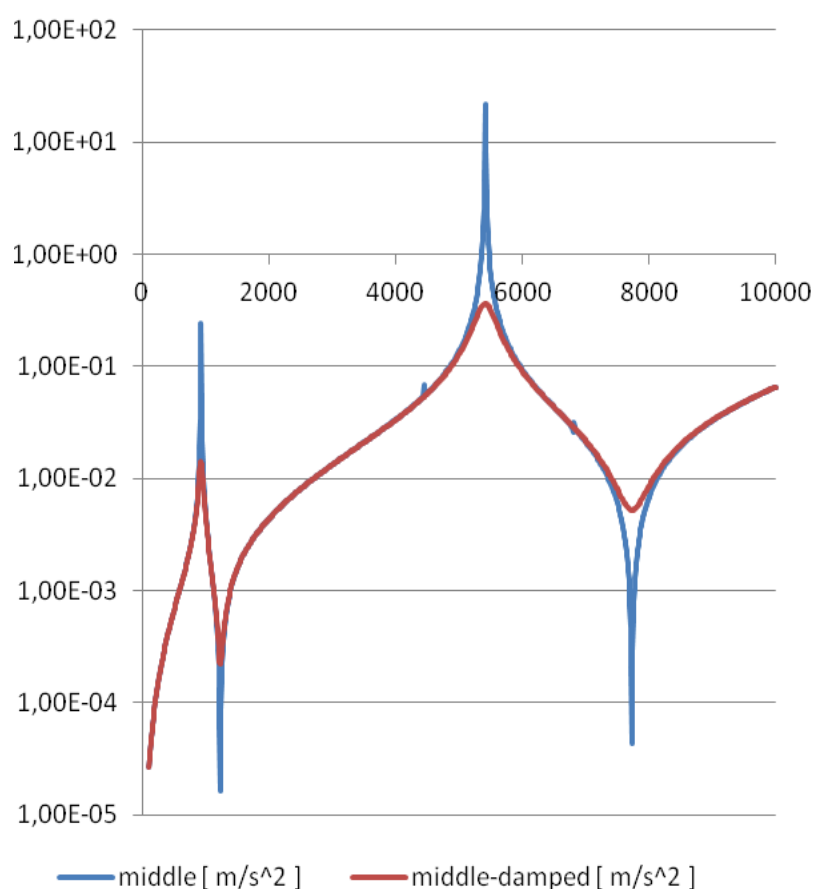
równowagi. Dzięki temu możliwe jest dokładne przewidywanie reakcji struktury na różne obciążenia, w tym w przypadku płytki czujnikowej pod działaniem siły o wartości 1 [g] w kierunku Z. Ważnym aspektem analizy MES jest uwzględnienie odpowiedniego zakresu częstotliwości w symulacji. W przypadku badanej płytki czujnikowej, istotne było uwzględnienie zakresu od 100 Hz do 10 kHz, ponieważ to właśnie w tym zakresie znajdował się liniowy zakres odpowiedzi częstotliwościowej czujnika. Dzięki temu możliwe było dokładne odwzorowanie jego charakterystyki i identyfikacja najważniejszych parametrów, takich jak częstotliwości drgań własnych. W tabeli 5.1 przedstawiono wyniki symulacji, gdzie zestawiono symulowane częstotliwości drgań własnych z analizy modalnej oraz częstotliwości rezonansowe z analizy harmonicznej. Analiza modalna umożliwia identyfikację i charakterystykę poszczególnych trybów drgań struktury, podczas gdy analiza harmoniczna skupia się na wykrywaniu częstotliwości rezonansowych, czyli tych, które wywołują największą odpowiedź układu. Przeprowadzenie symulacji MES w oparciu o określony zakres częstotliwości pozwoliło na dokładne poznanie zachowania płytki czujnikowej pod obciążeniem o wartości 1 [g] w kierunku Z.

*Tabela 5.1: Uzyskane wyniki analizy modalnej i harmonicznej dla dwóch rozważanych lokalizacji czujnika MEMS*

Mode	Frequency [Hz]	
	Location: Bottom	Location: Middle
1	0	0
2	1,33e-2	2,64e-6
3	44,16	1,24e-2
4	127,63	113,21
5	201,65	138,57
6	3 309,50	247,66
7	9 856,50	1 085,00
8	18 115,00	4 460,30
9	20 350,00	5 492,90
10	28 831,00	8 018,60
11	30 664,00	13 965,00
Resonance	Location: Bottom	Location: Middle
1	3 178,9	931,6
2	9 841,6	5 416,3

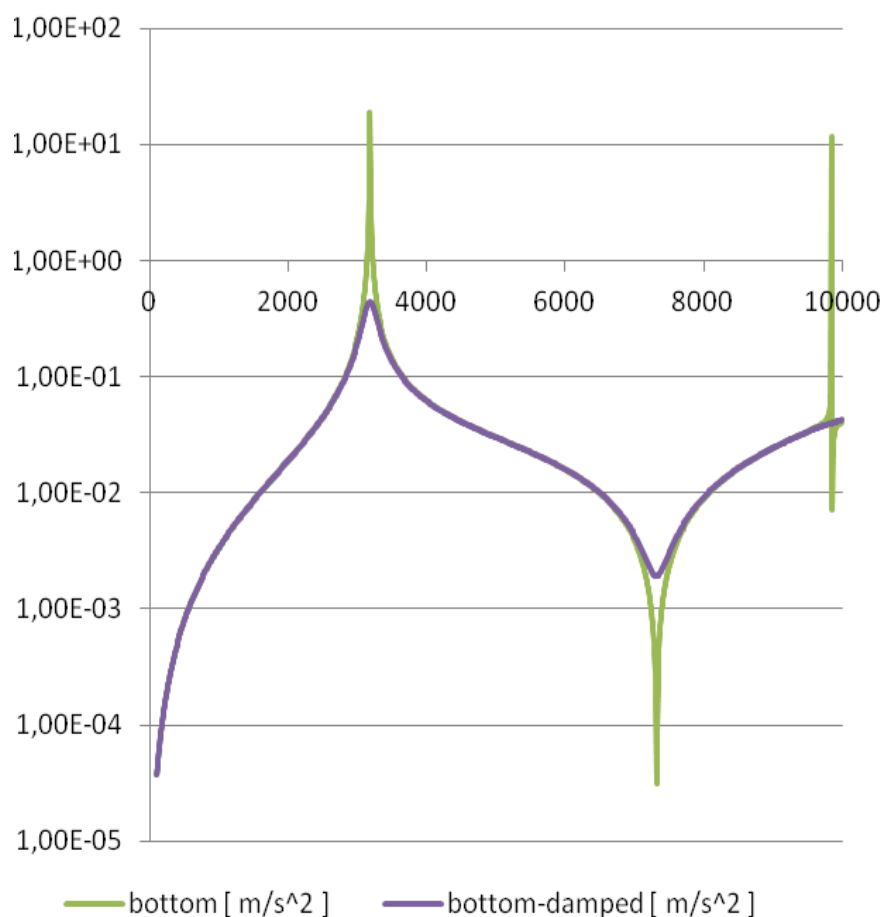
Dla uproszczenia wprowadzono współczynnik tłumienia na poziomie 3%, aby zbliżyć się do rzeczywistego tłumienia konstrukcji. Przyjęcie współczynnika tłumienia na poziomie 3%

może być uznane za odpowiednie w niektórych przypadkach, szczególnie gdy nie jest istotne uwzględnienie bardziej skomplikowanych aspektów tłumienia struktury. Wielkość współczynnika tłumienia zależy od wielu czynników, takich jak materiał struktury, jej geometria, warunki otoczenia i sposób ekscytacji. Dlatego wartość 3% może być przybliżeniem ogólnym, które daje satysfakcjonujące wyniki dla wielu typowych struktur. Warto jednak zaznaczyć, że istnieją sytuacje, w których dokładniejsze uwzględnienie tłumienia jest konieczne. W takich przypadkach, analiza MES może wymagać bardziej zaawansowanych modeli tłumienia, które uwzględniają właściwości materiałów i struktury w sposób bardziej precyzyjny.



*Rysunek 5.2: Charakterystyka częstotliwościowa czujnika z MEMS umieszczoną na środku płytki PCB*

Na rysunku 5.2 przedstawiono otrzymanie odpowiedzi częstotliwościowej dla czujnika z MEMS umieszczonym na środku płytki PCB.

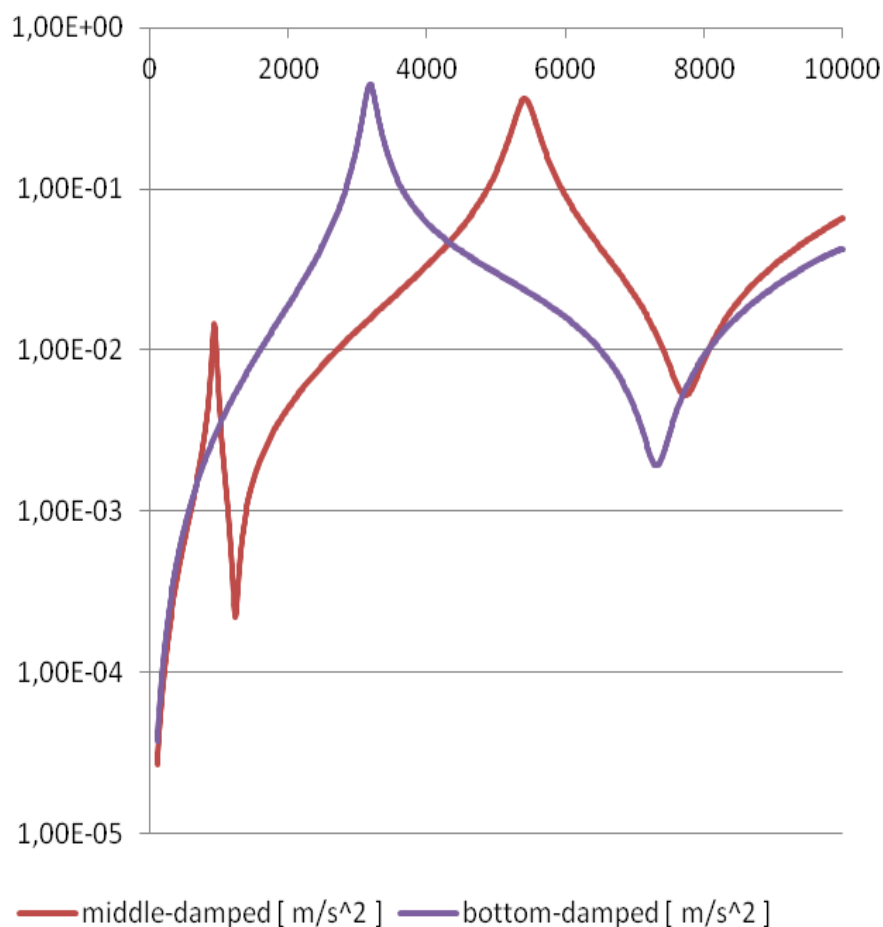


Rysunek 5.3: Charakterystyka częstotliwościowa czujnika z MEMS umieszczonym na spodzie płytki PCB.

Rysunek 5.3 przedstawia otrzymanie odpowiedzi częstotliwościowej dla czujnika z MEMS umieszczoną na spodzie płytki PCB.

Przesunięcie położenia akcelerometru może wpływać na rozkład naprężeń i deformacji w konstrukcji, co z kolei może zmieniać jej charakterystykę dynamiczną. Wybór optymalnego położenia akcelerometru jest istotny dla uzyskania dokładnych pomiarów przyspieszenia i zrozumienia zachowania struktury w odpowiedzi na obciążenie mechaniczne. Przesunięcie akcelerometru może również mieć wpływ na czułość i zakres dynamiczny pomiaru, a także na eliminację zakłóceń związanych z innymi czynnikami, takimi jak wibracje otoczenia. Wyniki przedstawione na rysunku 5.4 obrazują różnice między konstrukcją wytłumioną a niewytłumioną. Wytłumienie struktury może zmniejszyć amplitudę drgań i rezonanse w konkretnym paśmie częstotliwości, co przekłada się na poprawę wydajności i dokładności pomiaru. Analiza wpływu położenia akcelerometru na charakterystykę konstrukcji jest istotna przy projektowaniu czujników i systemów monitorowania. Pozwala to na optymalne

rozmieszczenie czujników w celu uzyskania jak najdokładniejszych i najbardziej reprezentatywnych danych. Przesunięcie akcelerometru może być wykorzystane jako narzędzie do dostosowania parametrów pomiarowych w zależności od konkretnych wymagań i warunków aplikacji. Wnioski wyciągnięte z analizy położenia akcelerometru mają istotne znaczenie dla doskonalenia projektów konstrukcyjnych, poprawy wydajności systemów monitorowania i zapewnienia wysokiej jakości danych pomiarowych.



Rysunek 5.4: Porównanie odpowiedzi częstotliwościowej struktury w zależności od lokalizacji MEMS na płycie PCB

Wstępna analiza harmoniczna modelu mechanicznego wykazała potencjalne rezonanse w zakresie szerokości pasma akcelerometru. Podczas symulacji okazało się, że w zakresie od 1 Hz do 10 kHz mogą wystąpić dwa potencjalne rezonanse około ~1 kHz, ~5,5 kHz dla konstrukcji „środkowej” i około ~3 kHz, ~10 kHz dla „dolnej części”. ' projekt. Wskazuje to, że zmiana położenia akcelerometru MEMS może podwoić częstotliwość rezonansową

konstrukcji (dla rozważanej konstrukcji mechanicznej). W celu sprawdzenia poprawności modelu zaplanowano i przeprowadzono eksperyment.

W tym rozdziale opisano eksperymenty, które przeprowadzono w celu walidacji użytkownika modelu do symulacji. W tym celu opracowano prototyp płytki PCB z akcelerometrem MEMS umieszczonym na spodzie płytki (obudowa na MEMS znajdującą się bliżej punktu mocowania czujnika). W tabeli 2 przedstawiono parametry techniczne wybranego akcelerometru.

*Tabela 5.2: Specyfikacja techniczna akcelerometru MEMS wybranego do prototypu czujnika*

Parameter	ADX1001	Unit
Measurement Range	±100	g
Sensitivity	20	mV/g
Noise Density	30	μg/√Hz
Sensor Resonant Frequency	21	kHz
Operating Temperature	-40, +125	°C

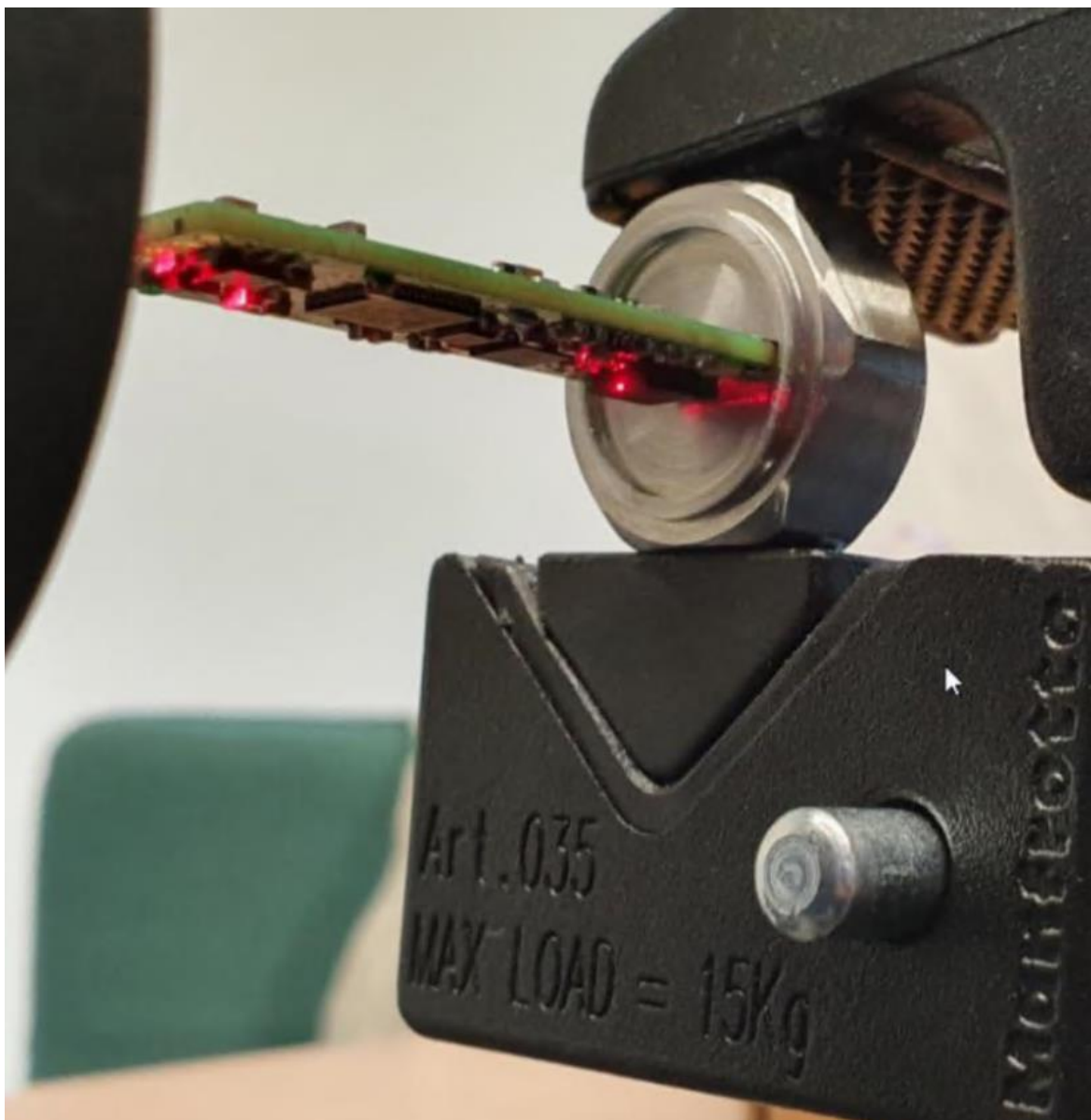
### 5.3.1 Pomiar za pomocą wibrometru laserowego PCV400

W celu potwierdzenia poprawności przeprowadzonych symulacji, przystąpiono do eksperymentu, w którym wykorzystano wibrometr laserowy do pomiaru drgań analizowanej konstrukcji. Wibrometr, który mierzy prędkość drgań, był używany do porównania wyników symulacji z danymi rzeczywistymi. W tym eksperymencie uwzględniono dane dotyczące prędkości, które zostały uzyskane z symulacji (symulacja obejmowała zarówno przemieszczenia, jak i prędkości oraz przyspieszenia drgań). Obrazowanie stanowiska badawczego użytego w tym eksperymencie przedstawiono na Rysunku 5.5 oraz Rysunku 5.6. Jako źródło pobudzenia wykorzystano głośnik audio. Eksperyment ten miał na celu porównanie wyników symulacji z rzeczywistymi danymi pomiarowymi, co pozwoliłoby na weryfikację poprawności modelu symulacyjnego. Przy użyciu wibrometru laserowego możliwe było bezkontaktowe i precyzyjne pomiar prędkości drgań, co umożliwiło porównanie z danymi symulacyjnymi. Wykorzystanie głośnika audio jako źródła wzbudzenia miało na celu generację kontrolowanego sygnału drgań, który był odpowiednio dostosowany do badanej konstrukcji. Dzięki temu możliwe było zbadanie reakcji konstrukcji na określone pobudzenie i porównanie jej z rezultatami symulacji. Przeprowadzenie eksperymentu z użyciem wibrometru laserowego oraz głośnika audio stanowiło niezależne potwierdzenie wyników symulacji, co przyczyniło się do wiarygodności i poprawności analizy. Porównanie

uzyskanych danych eksperymentalnych z danymi symulacyjnymi pozwoliło na ocenę dokładności i adekwatności modelu numerycznego oraz weryfikację jego użyteczności w przewidywaniu zachowania konstrukcji w odpowiedzi na różne pobudzenia.

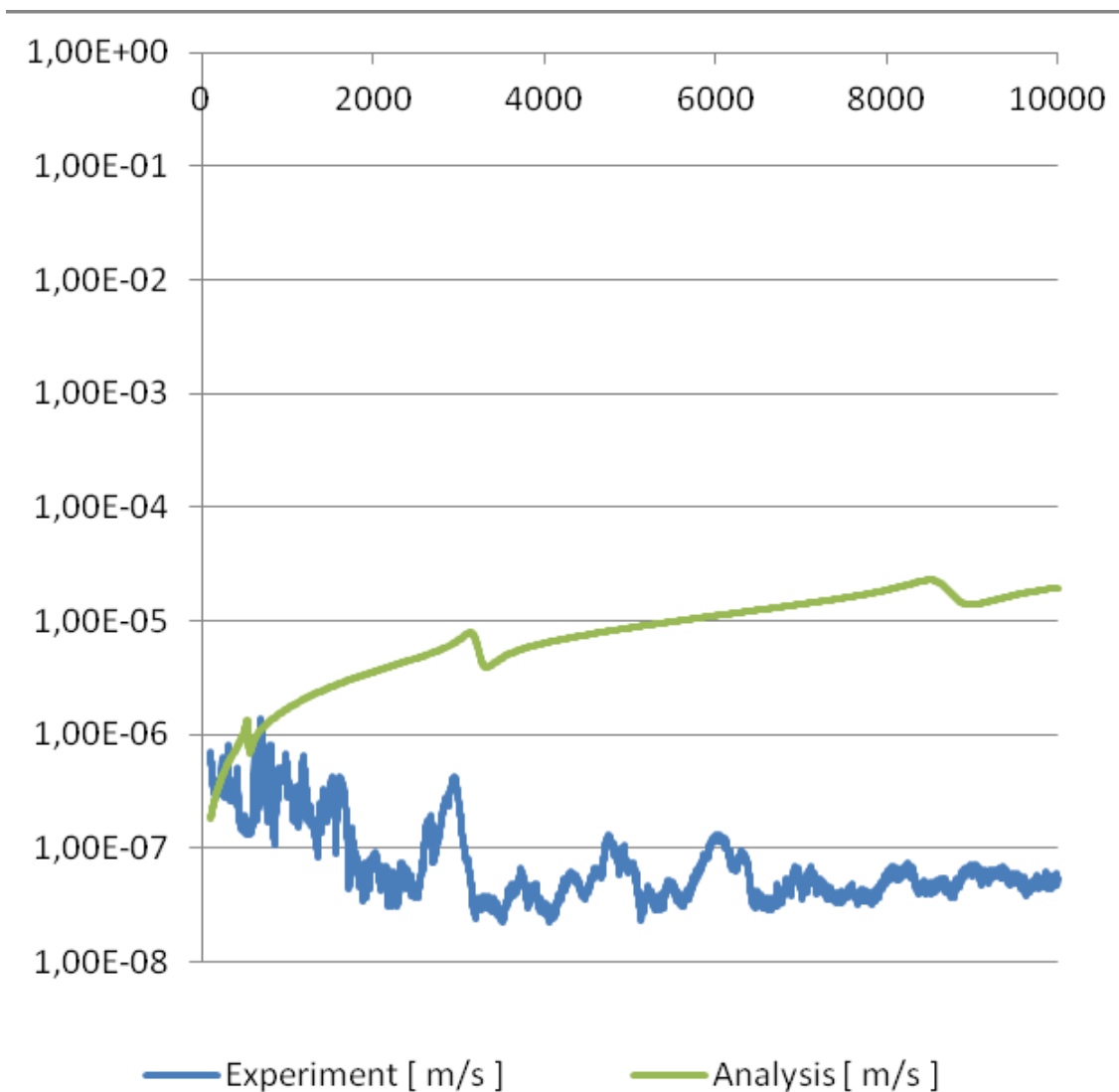


*Rysunek 5.5: Konfiguracja eksperymentu przy użyciu wibrometru laserowego Polytec PSV400*



*Rysunek 5.6: Konfiguracja eksperymentu z użyciem wibrometru laserowego Polytec PSV400 – wiązka lasera skierowana na tarczę akcelerometru w kierunku pomiaru*

Wyniki uzyskane z eksperymentu (przedstawione na rysunku 5.7) w porównaniu z analizą MES wskazują, że symulowany kształt odpowiedzi częstotliwościowej, a zwłaszcza położenie założonego rezonansu w okolicach 3-3,5 kHz, przejawia się zarówno w wynikach symulacji, jak i eksperymentu. Właściwie nie było możliwe prawidłowe wzbudzenie układu głośnikiem w zakresie od 100 Hz do 10 kHz, ale uzyskane wyniki sugerują, że symulacja może być trafna i stąd hipoteza, że im bliżej MEMS do punktu mocowania czujnik, im mniejszy jest na niego wpływ niepożądane efekty propagacji drgań, może być również ważny.



Rysunek 5.7: Porównanie otrzymanej odpowiedzi częstotliwościowej z analizy (obliczonej dla prędkości w [m/s]) i eksperymentu z użyciem wibrometru laserowego PSV400

Uzyskane wyniki eksperymentalne oraz symulacyjne wykazały, że zmiana położenia akcelerometru MEMS w stosunku do punktu mocowania czujnika ma istotny wpływ na charakterystykę rezonansową konstrukcji. W szczególności, umieszczenie akcelerometru bliżej punktu mocowania przyczynia się do przesunięcia dwóch charakterystycznych rezonansów w wyższe pasmo częstotliwości. Ten efekt można zaobserwować zarówno na podstawie uzyskanych wyników symulacji, jak i na podstawie danych eksperymentalnych. Warto zaznaczyć, że podczas eksperymentu użyto głośnika audio jako źródła wzbudzenia. Jednak ze względu na ograniczenia głośnika, nie było możliwe generowanie szerokiego zakresu częstotliwości, co stanowi pewne wyzwanie w dalszych badaniach. W związku z tym, konieczne jest zastosowanie innego, bardziej wszechstronnego źródła wzbudzenia, które

pozwole na badanie zachowania konstrukcji w szerszym zakresie częstotliwości. Dalsze eksperymenty będą wymagały zastosowania odpowiednich urządzeń wzbudzających, które umożliwią przyłożenie obciążenia w szerokim zakresie częstotliwości. Dzięki temu będzie można przeprowadzić bardziej kompleksowe badania, które pozwolą na szczegółową analizę charakterystyki rezonansowej konstrukcji w różnych częstotliwościach. Odpowiedni dobór źródła wzbudzenia będzie kluczowy dla uzyskania rzetelnych i wiarygodnych danych eksperymentalnych, które będą stanowić solidne podstawy do dalszej analizy i optymalizacji konstrukcji.

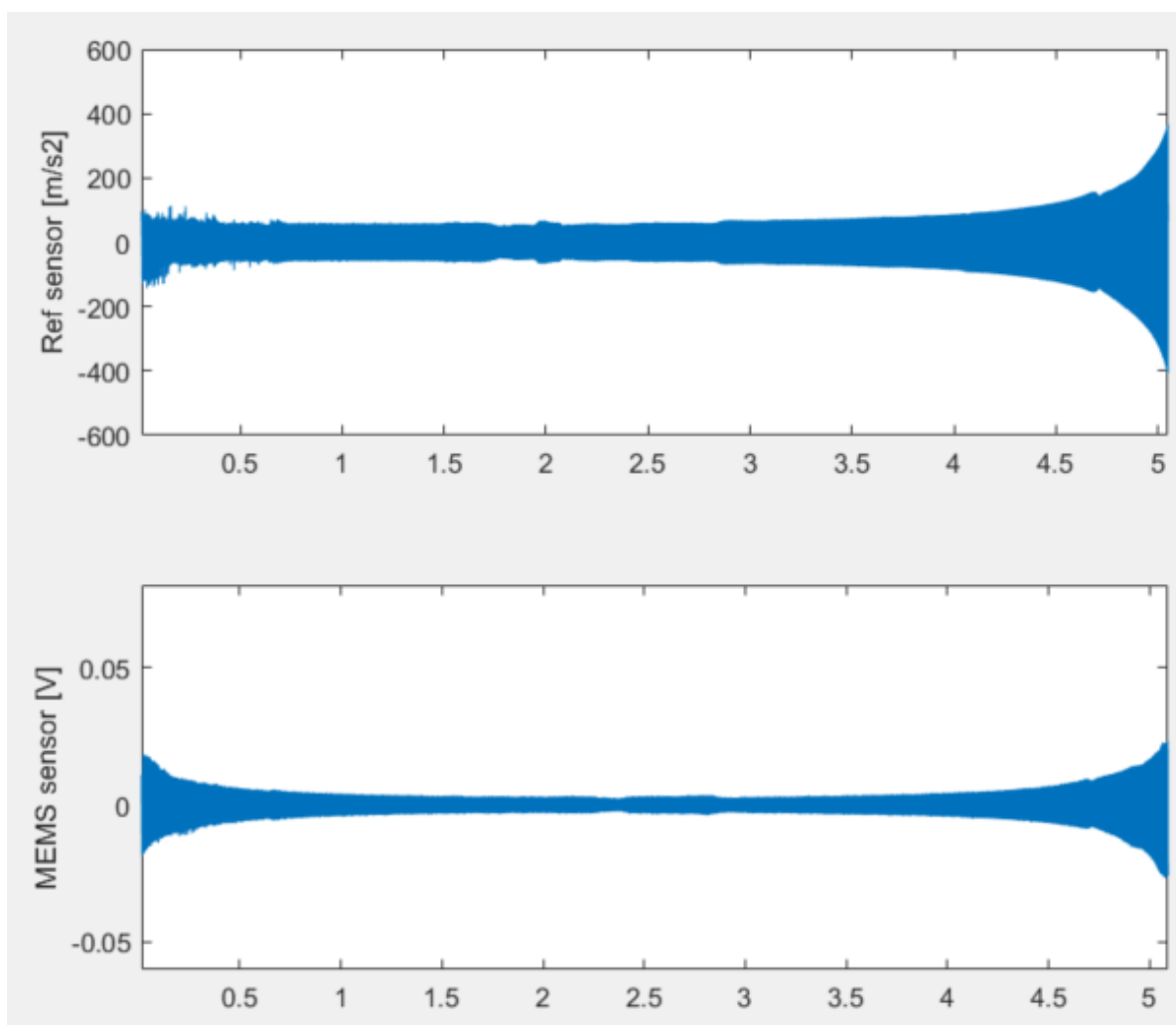
### 5.3.2 Pomiar za pomocą TIRAvib TV51144

Ponieważ wyniki poprzedniego eksperymentu wskazywały, że nasze założenie jest prawidłowe, przeprowadziliśmy kolejny eksperyment, podczas gdy byliśmy w stanie sterować sygnałem wejściowym w sposób bardziej bezpośredni. W tym celu wykorzystano shaker TIRAvib TV51144. Rysunek 5.8 przedstawia stanowisko badawcze użyte podczas tego eksperymentu. Jako źródło wzbudzenia zastosowano wytrząsarke, a jako odniesienie zastosowano dodatkowy czujnik piezoelektryczny. Rysunek 5.8 przedstawia stanowisko badawcze użyte podczas tego eksperymentu.



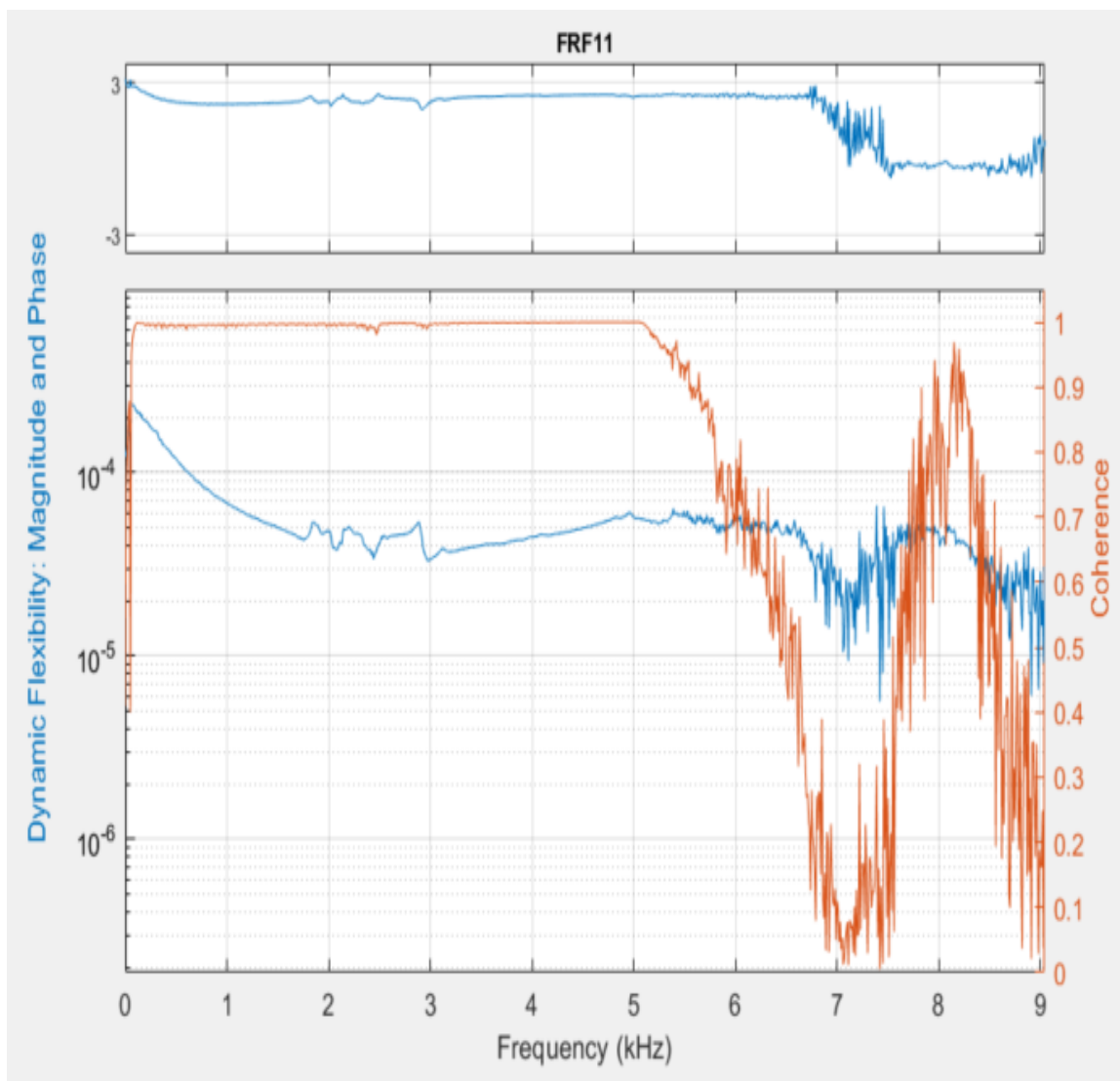
*Rysunek 5.8: Konfiguracja eksperymentu przy użyciu wzbudnika TIRAvib TV51144*

Wyniki uzyskane z eksperymentu przedstawiono na rysunku 5.9. Górny wykres przedstawia czujnik odniesienia, a dolny wykres akcelerometr MEMS. Wyniki pokazują, że wytrząsarka ma pewne mechaniczne niedoskonałości, które są wizualizowane przez „trzaski” na wykresie sygnału odniesienia. Częstotliwość rezonansowa wibratora ma (zgodnie ze specyfikacją techniczną) wynosić około 6,5 kHz, ale można zaobserwować, że rezonans faktycznie wynosi 5 kHz (zaczyna się około 4,5 kHz).



Rysunek 5.9: Surowy sygnał mierzony z czujników referencyjnych i MEMS na wzbudniku TIRAvib TV51144

Rysunek 5.10 przedstawia funkcję odpowiedzi częstotliwościowej (FRF) obliczoną w Matlab z sygnału zmierzonego przez akcelerometr MEMS (w obliczeniach zastosowano prostokątne okno 100ms).



Rysunek 5.10: Odpowiedź częstotliwościowa obliczona na podstawie zmasowanego sygnału z czujnika MEMS na wytryszarce TIRAvib TV51144

Otrzymany FRF jest stabilny do 5kHz, co pokazuje, że symulowany rezonans ( $\sim 3$ kHz) dla „dolnej” lokalizacji akcelerometru MEMS nie jest widoczny podczas eksperymentu, co oznacza, że symulacja może wskazać, gdzie może wystąpić rezonans, jednakże przeprowadzony eksperyment wskazuje, że rezonans ten jest skutecznie wytłumiony przez konstrukcję czujnika.

### 5.3.3 Analiza wyników

Otrzymana FRF (Funkcja Odpowiedzi Częstotliwościowej) konstrukcji, którą przeprowadzono zarówno w symulacji, jak i eksperymencie, wykazała stabilność do częstotliwości 5 kHz. Jest to istotne, ponieważ symulacja uwzględniała rezonans o częstotliwości około 3 kHz, który mógłby wystąpić dla "dolnej" lokalizacji akcelerometru

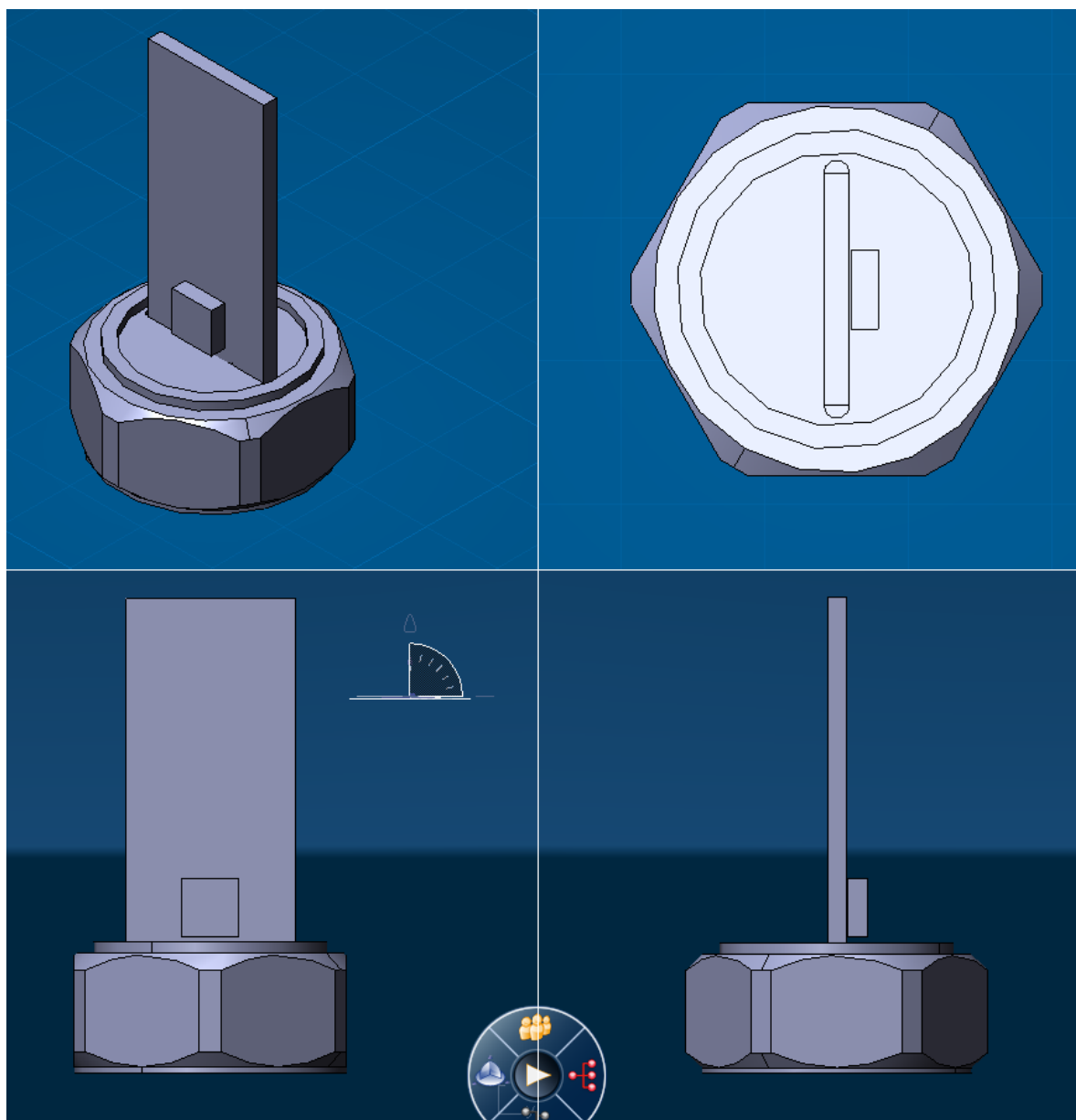
MEMS. Jednak wyniki eksperymentalne wskazują, że ten rezonans nie był widoczny, co sugeruje, że konstrukcja czujnika skutecznie wytłumiła ten rezonans. Porównanie wyników symulacji i eksperymentu jest istotne, ponieważ umożliwia ocenę skuteczności symulacji w przewidywaniu zachowania konstrukcji w rzeczywistości. W tym przypadku, choć symulacja wskazała na możliwość wystąpienia rezonansu dla "dolnej" lokalizacji akcelerometru, eksperyment potwierdził, że takiego rezonansu nie zaobserwowano. Oznacza to, że konstrukcja czujnika została zaprojektowana w taki sposób, aby skutecznie tłumić ten rezonans i zapewnić stabilność działania w badanym zakresie częstotliwości. Warto zauważyć, że porównanie symulacji i eksperymentu pozwala również na identyfikację ewentualnych rozbieżności między nimi. Jeśli symulacja wskazuje na obecność rezonansu, który nie został potwierdzony eksperymentalnie, może to sugerować pewne niedoskonałości w modelu symulacyjnym lub potrzebę dodatkowych poprawek w konstrukcji czujnika. Wnioskiem z tych obserwacji jest fakt, że symulacje mogą dostarczać cennych informacji na temat potencjalnych rezonansów i charakterystyk konstrukcji, ale ich wyniki powinny być zawsze weryfikowane poprzez eksperymenty, aby potwierdzić ich poprawność. Otrzymane rezultaty pokazują, że konstrukcja czujnika jest skuteczna w tłumieniu rezonansu i stabilizacji działania w badanym zakresie częstotliwości. Z przeprowadzonych badań wynika, że koncepcja stworzenia systemu monitorowania stanu, w tym akcelerometrów opartych na MEMS, ma potencjał i warty jest zbadania, aby wyniki rozwoju MEMS znalazły zastosowanie w przemyśle.

#### 5.4 Badanie wpływu geometrii czujnika

Konstrukcja mechaniczna czujnika i lokalizacja jego punktu instalacji to dwa najważniejsze czynniki akwizycji danych o drganiach, dlatego punkt odczytu musi być wybierany w sposób znaczący, ponieważ maszyny wirujące generują drgania za pomocą cyklicznych sił wewnętrznych i nie można ich bezpośrednio zmierzyć. Większość istniejących procedur diagnostycznych opiera się na ocenie przenoszenia tych sił przez podwozie maszyny. Oznacza to, że w rzeczywistości mierzona jest reakcja konstrukcji maszyny na obciążenie. W związku z tym należy przetestować konstrukcję czujnika akcelerometru i dobrać odpowiednią lokalizację punktu odczytu, aby zminimalizować niepożądane efekty wynikające z mechanicznego wzbudzenia konstrukcji.

Na rys. 5.11 przedstawiono proponowaną uproszczoną konstrukcję płytki czujnikowej z akcelerometrem MEMS. Następnie przeprowadzono analizę MES z wykorzystaniem stworzonego modelu w celu symulacji wzbudzenia płytki czujnikowej pod obciążeniem (1 [g]) w kierunku Z – osi pomiaru wybranego akcelerometru MEMS. Zgodnie z dokumentacją

techniczną czujnik ma liniowy zakres odpowiedzi częstotliwościowej do 11 kHz (Analog Devices 2018), a analizę przeprowadzono w zakresie od 10 Hz do 10 kHz.



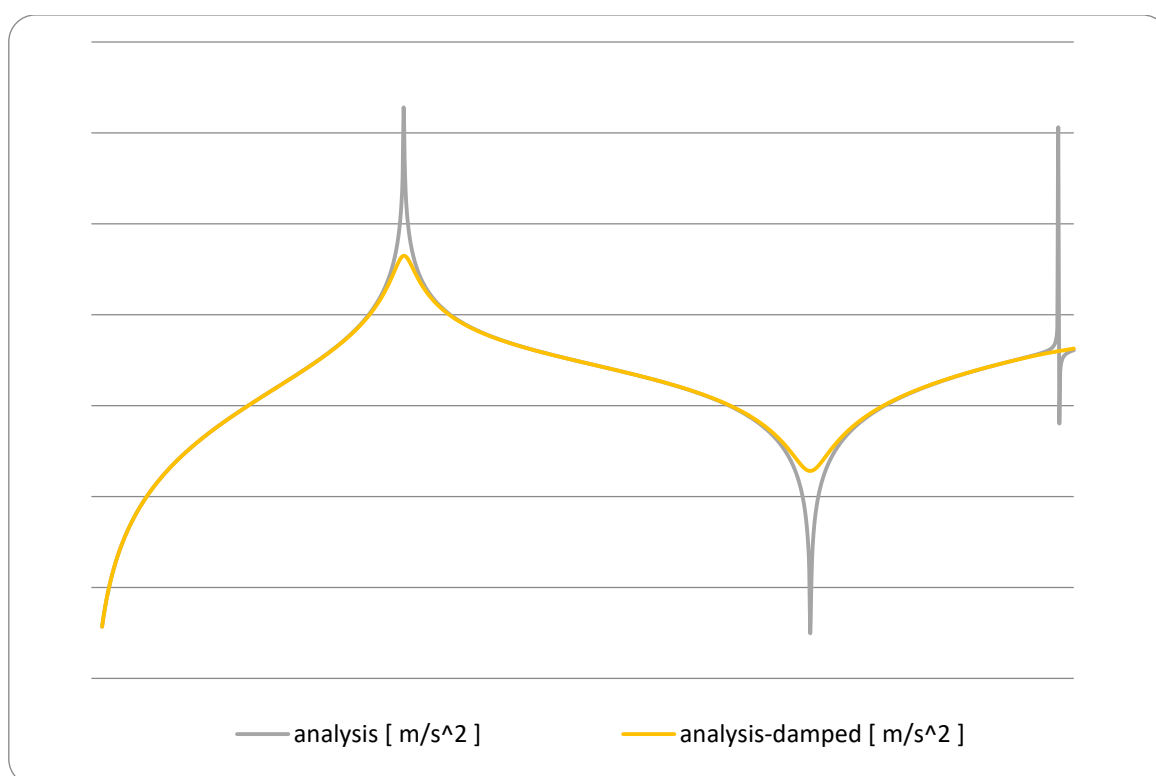
*Rysunek 5.11: Proponowana uproszczona konstrukcja płytki czujnikowej z akcelerometrem MEMS*

W tabeli 5.3 przedstawiono symulowane częstotliwości drgań własnych z analizy modalnej oraz częstotliwości rezonansowe z analizy harmonicznej.

Tabela 5.3: Własności mechaniczne zaproponowanego modelu

Mode	Częstotliwość [Hz]
1	0
2	1,33e-2
3	44,16
4	127,63
5	201,65
6	3 309,50
7	9 856,50
Rezonans	Częstotliwość [Hz]
1	3 178,9
2	9 841,6

Na rys. 5.12 przedstawiono uzyskaną charakterystykę częstotliwościową czujnika z MEMS umieszczonym na płytce PCB. Dla uproszczenia wprowadzono współczynnik tłumienia na poziomie 3%, aby zbliżyć się do rzeczywistego tłumienia konstrukcji.



Rysunek 5.12: Pasmo przenoszenia czujnika z akcelerometrem MEMS. Założone tłumienie 3% zostało uwzględnione podczas symulacji (wykres tłumienia analizy)

Uzyskane wyniki wskazują, że proponowana konstrukcja płytki czujnikowej może mieć rezonanse wewnętrzne w okolicach 3 – 3,5 kHz oraz 9,5-10 kHz, co mieści się w założonym zakresie częstotliwości pracy. Symulacja z założonym uproszczonym tłumieniem pokazuje, że drugi rezonans zostanie w pełni wytłumiony przez strukturę układu. Aby sprawdzić, czy ten wynik jest prawidłowy, zaplanowano serię pomiarów, aby sprawdzić, czy taka konstrukcja czujnika pozwoli na wykrycie awarii REB na stole probierczym. Pierwszy prototyp powstał według projektu przedstawionego na rys. 5.11.

Badania eksperymentalne przeprowadzono na stanowisku probierczym VibStand 2, produkt firmy AMC Vibro sp. z o.o. Jest to stanowisko doświadczalne przeznaczone do badań wibracyjnych. Umożliwia obserwację drgań generowanych przez silnik, przekładnię, drgań wywołanych niewyważeniem i pochodzących od łożysk, w tym łożysk uszkodzonych i pracujących pod obciążeniem.

Pierwszym krokiem eksperymentu było sprawdzenie poziomu szumu sygnału zarówno dla wzorca referencyjnego, jak i prototypu MEMS High Frequency, w celu sprawdzenia, czy otrzymany sygnał jest wystarczająco czysty, aby można było wykonać również kolejne kroki. W tabeli 5.3 przedstawiono uzyskane wyniki – czujnik MEMS wysokiej częstotliwości jest akcelerometrem jednoosiowym, a wybranym czujnikiem odniesienia jest akcelerometr trójosiowy.

*Tabela 5.4: Zmierzony poziom szumu sygnału*

	MEMS [g]	Czujnik Referencyjny (piezo) [g]
X: $\text{RMS}(x - \text{MEAN}(x))$	n/a	0.0042
Y: $\text{RMS}(y - \text{MEAN}(y))$	n/a	0.0114
Z: $\text{RMS}(z - \text{MEAN}(z))$	0.0033	0.0069

Tabela 5.4 przedstawia porównanie poziomu szumu dla dwóch rodzajów akcelerometrów: MEMS (Microelectromechanical Systems) oraz Czujnika Referencyjnego (piezoelektrycznego). Szum jest mierzony, jako RMS (Root Mean Square) różnicy między wartościami pomiarowymi a ich średnią wartością dla poszczególnych osi: X, Y i Z.

W przypadku osi X, nie został podany wynik dla akcelerometru MEMS, co oznacza, że nie były dostępne dane dotyczące poziomu szumu dla tej osi (jest to akcelerometr jednoosiowy). Natomiast dla Czujnika Referencyjnego (piezoelektrycznego) poziom szumu wynosi 0.0042 g. Dla osi Y, podobnie jak w przypadku osi X, brak danych dla akcelerometru

MEMS. Dla Czujnika Referencyjnego (piezoelektrycznego) otrzymano poziom szumu wynoszący 0.0114 g. Wreszcie, dla osi Z, poziom szumu dla akcelerometru MEMS wynosi 0.0033 g, podczas gdy dla Czujnika Referencyjnego (piezoelektrycznego) wynosi 0.0069 g.

Analizując te wyniki, można zauważyć, że dla osi Z, akcelerometr MEMS wykazuje niższy poziom szumu (0.0033 g) w porównaniu do Czujnika Referencyjnego (piezoelektrycznego) (0.0069 g). Uzyskane wyniki wskazują, że poziom szumu czujnika MEMS nie powinien zakłócać pomiarów.

## 6 Projekt wbudowanego systemu monitorowania stanu

W dzisiejszym dynamicznym środowisku przemysłowym, monitorowanie stanu maszyn jest nieodzownym elementem utrzymania wysokiej wydajności i niezawodności procesów produkcyjnych. Nowoczesne technologie umożliwiają rozwój zaawansowanych systemów monitorowania, które pozwalają na ciągłą ocenę i diagnozowanie stanu maszyn w czasie rzeczywistym. Takie rozwiązania umożliwiają wczesne wykrywanie potencjalnych awarii, minimalizując czas przestoju, koszty napraw i straty produkcyjne. Aby sprostać rosnącym wymaganiom i wyzwaniom, konieczne jest opracowanie nowych, nowoczesnych systemów monitorowania. W ramach niniejszego rozdziału przedstawiony zostanie projekt wbudowanego systemu monitorowania stanu, który ma na celu skonstruowanie efektywnego i precyzyjnego narzędzia diagnostycznego.

### 6.1 Analiza stanu techniki

Rozwiązanie opisane w międzynarodowym patencie CN204461486U dotyczy tworzenia bezprzewodowego systemu akwizycji danych opartego na technologii Internetu Rzeczy (IoT) [6.1]. W tym opatentowanym systemie wykorzystuje się czujnik prędkości lub przemieszczenia, bazę danych, serwis typu WEB oraz moduł komunikacji oparty na technologiach Wi-Fi lub GSM w celu udostępnienia danych pomiarowych. Głównym celem tego rozwiązania jest umożliwienie zdalnego monitorowania i gromadzenia danych pomiarowych za pomocą bezprzewodowego interfejsu komunikacyjnego. System składa się z układu pomiarowego, który odpowiada za pobieranie, przetwarzanie i przesyłanie danych pomiarowych. Układ pomiarowy zbudowany jest z czujnika analogowego, który rejestruje prędkość lub przemieszczenie, oraz przetwornika analogowo-cyfrowego, który przekształca sygnał analogowy na cyfrowy. Ważną cechą tego opatentowanego rozwiązania jest możliwość filtracji, amplifikacji i konwersji danych pomiarowych w układzie pomiarowym. Dzięki temu możliwe jest dokładne i precyzyjne zbieranie informacji o prędkości lub przemieszczeniu badanego obiektu. Dane pomiarowe są następnie przesyłane do bazy danych, gdzie mogą być przechowywane i analizowane w celu dalszej diagnostyki lub monitorowania stanu. System opisany w opatentowanym rozwiązaniu znacznie ułatwia zdalne monitorowanie i akwizycję danych pomiarowych. Dzięki technologii IoT i bezprzewodowej komunikacji, dane mogą być odczytywane i udostępniane z dowolnego miejsca za pomocą interfejsu internetowego. To otwiera szerokie możliwości zastosowań tego systemu w różnych dziedzinach, takich jak przemysł, budownictwo czy medycyna. Należy jednak zaznaczyć, że opisane w patencie rozwiązanie koncentruje się głównie na akwizycji danych i komunikacji,

nie uwzględniając aspektów miniaturyzacji, modularyzacji ani wykorzystania technologii bezprzewodowych, które są istotne w projektowaniu nowoczesnych systemów monitorowania stanu.

Opisana w amerykańskim wynalazku US 8614928 B2 metoda autonomicznej inicjalizacji bezprzewodowych systemów akwizycji danych opiera się na wykorzystaniu sygnału inicjalizującego [6.2]. Dzięki tej metodzie, sąsiednie urządzenia pomiarowe mają możliwość samodzielnego rozpoczęcia wykonywania pomiarów oraz ustalenia ścieżki wysyłania zebranych danych. Metoda ta ma istotne znaczenie dla skutecznego i efektywnego funkcjonowania bezprzewodowych systemów akwizycji danych. Dzięki inicjalizacji poprzez sygnał inicjalizujący, proces konfiguracji i uruchamiania systemu staje się automatyczny i nie wymaga bezpośredniej interakcji człowieka. To z kolei przyspiesza proces rozpoczęcia pomiarów i umożliwia szybkie zebranie danych pomiarowych z różnych urządzeń. Wykorzystanie tej metody pozwala na skoordynowane działanie wielu urządzeń pomiarowych, które mogą być rozmieszczone na większym obszarze. Dzięki temu możliwe jest monitorowanie i akwizycja danych z wielu punktów jednocześnie, co znacznie zwiększa efektywność procesu pomiarowego. Wynalazek US 8614928 B2 otwiera nowe perspektywy w dziedzinie bezprzewodowych systemów akwizycji danych, umożliwiając szybką inicjalizację i koordynację urządzeń pomiarowych. Ta metoda jest szczególnie przydatna w przypadku systemów, w których istnieje potrzeba zebrania danych z wielu źródeł jednocześnie, takich jak systemy monitorowania stanu, diagnostyka techniczna czy analiza wibracji.

Opisana w międzynarodowym patencie CN104931264A metoda akwizycji danych i monitorowania stanu maszyn opiera się na zastosowaniu specjalnego układu składającego się z czujnika, mikroprocesora oraz źródła zasilania [6.3]. Dzięki tej metodzie możliwe jest skuteczne zbieranie danych diagnostycznych oraz monitorowanie stanu maszyn w celu wykrywania ewentualnych awarii lub nieprawidłowości. W ramach tej metody, czujnik odpowiedzialny jest za pomiar różnych parametrów charakterystycznych dla monitorowanych maszyn, takich jak wibracje, temperatura czy poziom hałasu. Zebrane dane są następnie przekazywane do mikroprocesora, który dokonuje ich analizy i oceny stanu maszyny. W przypadku przekroczenia ustalonego progu alarmowego, metoda umożliwia wygenerowanie odpowiedniego sygnału alarmowego, informującego o potencjalnym zagrożeniu lub konieczności interwencji. Wprowadzenie tej metody akwizycji danych i monitorowania stanu maszyn przyczynia się do zwiększenia niezawodności i bezpieczeństwa procesów

produkcyjnych. Dzięki stałemu monitorowaniu i wykrywaniu nieprawidłowości w czasie rzeczywistym, możliwe jest zapobieganie awariom maszyn, minimalizacja strat wynikających z przestoju produkcyjnych oraz optymalizacja procesów serwisowych. Patent CN104931264A dostarcza istotnych wskazówek dotyczących efektywnego monitorowania stanu maszyn i akwizycji danych diagnostycznych. Metoda ta może być wykorzystana w różnorodnych branżach, gdzie istnieje potrzeba ciągłego nadzoru nad maszynami i urządzeniami w celu zapewnienia ich prawidłowego funkcjonowania i minimalizacji ryzyka awarii.

Z międzynarodowego patentu CN204516019U pochodzi innowacyjny sposób na efektywne obniżenie zużycia energii elektrycznej w bezprzewodowych systemach pomiarowych. Ten wyjątkowy system zapewnia optymalne zarządzanie energią, co przyczynia się do zwiększenia wydajności i trwałości urządzeń oraz przedłuża czas działania baterii lub innych źródeł zasilania [6.4]. Dzięki zastosowaniu opatentowanej technologii, systemy pomiarowe mogą działać dłużej, co jest szczególnie istotne w przypadku monitorowania stanu maszyn lub w innych zastosowaniach, które wymagają ciągłej akwizycji danych. Optymalizacja zużycia energii pozwala również na redukcję kosztów eksploatacji i minimalizację negatywnego wpływu na środowisko.

Amerykańska firma IMI Sensors, będąca częścią PCB Piezoelectronics, wprowadziła na rynek innowacyjny produkt o nazwie Echo® Wireless Vibration Monitoring System [6.5]. Ten zaawansowany system umożliwia bezprzewodowy pomiar, analizę i transmisję danych w celu monitorowania drgań. Jednak istnieje pewne ograniczenie, które może wykluczyć ten produkt z niektórych zastosowań. Chodzi o wagę czujnika drgań, która wynosi 454g. Ta masa czujnika może być problematyczna w niektórych środowiskach lub aplikacjach, gdzie wymagane są niewielkie rozmiary i lekkość urządzenia. W takich przypadkach konieczne jest poszukiwanie alternatywnych rozwiązań, które zapewnią odpowiednie funkcje monitorowania drgań przy mniejszej wadze.

Amerykańska firma SKF znana jest z innowacyjnych rozwiązań w dziedzinie monitorowania stanu maszyn. Jednym z ich produktów jest bezprzewodowy czujnik drgań, model CMWA 8800 [6.6]. Ten zaawansowany czujnik oferuje możliwość bezprzewodowego pomiaru i monitorowania drgań w maszynach. Niestety, wysokość tego konkretnego modelu wynosi 100mm, co może stanowić problem w niektórych aplikacjach, gdzie ograniczenie miejsca jest istotne. W takich sytuacjach konieczne jest poszukiwanie alternatywnych

rozwiązań, które zapewnią niezawodne monitorowanie drgań przy mniejszych wymiarach, aby móc zastosować je w tych specyficznych warunkach przestrzennych.

Amerykańska firma LORD Corporation jest cenionym dostawcą zaawansowanych rozwiązań w dziedzinie monitorowania drgań. Ich produkt, bezprzewodowy akcelerometr G-Link2™-LXRS, charakteryzuje się wysoką odpornością na warunki atmosferyczne (klasa IP 67), co pozwala na precyzyjny pomiar drgań nawet w trudnych środowiskach [6.7]. Niestety, ze względu na zastosowaną technologię modułu radiowego, wymiary podstawy tego czujnika wynoszą 68mm x 85mm, co może ograniczać jego zastosowanie w niektórych aplikacjach o ograniczonej przestrzeni montażowej. W przypadku takich specyficznych warunków przestrzennych, konieczne jest poszukiwanie alternatywnych rozwiązań, które zapewnią niezawodne i precyzyjne pomiary drgań przy mniejszych wymiarach, dostosowanych do konkretnych potrzeb i ograniczeń przestrzennych.

Amerykańska firma PCB Piezoelectronics jest powszechnie znana z produkcji wysokiej jakości czujników i systemów pomiarowych. Jednym z ich produktów jest czujnik 356A16, który oferuje precyzyjną akwizycję danych [6.8]. Niestety, jest to czujnik przewodowy, co oznacza konieczność prowadzenia okablowania, co może być problematyczne w niektórych zastosowaniach, zwłaszcza gdy występują ograniczenia przestrzenne lub trudne warunki środowiskowe. Ponadto, czujnik 356A16 jest przystosowany do wąskopasmowej akwizycji danych, co może być nieodpowiednie w przypadku potrzeby monitorowania szerokiego zakresu częstotliwości drgań. W takich przypadkach, poszukiwanie bezprzewodowych rozwiązań, które oferują większą elastyczność i łatwość w instalacji, może być bardziej odpowiednie dla konkretnych potrzeb i zastosowań.

## 6.2 Główne cechy prezentowanego systemu

Sposób według przedstawionego wynalazku otwiera nowe możliwości w dziedzinie akwizycji danych diagnostycznych przy pomocy miniaturowych, bezprzewodowych urządzeń. Kluczowym aspektem tego rozwiązania jest znacząca redukcja zapotrzebowania na energię elektryczną, co przekłada się na wydłużenie czasu pracy urządzenia na baterii. Oznacza to, że można dokonywać pomiarów przez dłuższy okres czasu bez konieczności częstej wymiany baterii, co zwiększa wygodę użytkownika i zmniejsza koszty eksploatacji. Dodatkowym atutem tego sposobu jest możliwość akwizycji sygnałów w szerokim paśmie, co jest istotne w przypadku monitorowania różnych rodzajów drgań i sygnałów diagnostycznych. Dzięki temu można zbierać dokładne i szczegółowe dane dotyczące badanych urządzeń. Ponadto, zapewniona jest odporność na zmienne warunki środowiskowe, w tym bryzgi wody oraz kurz, zgodnie z normą IP54. To oznacza, że urządzenie jest chronione przed uszkodzeniem spowodowanym działaniem wody czy zanieczyszczeniami, co jest ważne w wielu aplikacjach, zwłaszcza w środowiskach przemysłowych. Ważnym elementem prezentowanego rozwiązania jest również zamontowany na badanym urządzeniu odpowiedni czujnik. Dzięki temu możliwe jest rejestracja cyfrowego sygnału przyspieszenia drgań. Zastosowanie miniaturowych urządzeń pozwala na precyzyjne i skuteczne monitorowanie stanu technicznego maszyn oraz detekcję wszelkich nieprawidłowości. Małe rozmiary urządzeń umożliwiają ich łatwą instalację i montaż w różnych miejscach, co zwiększa elastyczność i dostępność systemu monitorowania. Prezentowane rozwiązanie zawiera następujące główne cechy:

1. Miniaturyzacja – zmniejszenie parametrów rozmiaru znamionowego urządzenia jest możliwe dzięki zastosowaniu nowoczesnej elektroniki, w tym:
  - a) Czujników przyspieszenia – rozwój nowoczesnych technologii przyczynił się do znaczącej poprawy parametrów akcelerometrów wykonanych w technologii MEMS, która to pozwala na uzyskanie czujników, które zachęcają inżynierów głównie ze względu na ich mały rozmiar, małą wagę, niskie zapotrzebowanie na energię elektryczną, wysoką funkcjonalnością oraz stosunkowo niski poziom szumu własnego,
  - b) Nowoczesnych procesorów z wbudowaną technologią Bluetooth Low Energy, pozwalających znacząco zredukować rozmiary urządzenia zachowując przy tym pożądane parametry operacyjne. Ze względu na zastosowanie technologii Bluetooth Low Energy, wymagany zasięg w trybie pracy bezprzewodowej wymaga zastosowania anteny o niewielkich rozmiarach,

2. Modularyzacja – wykorzystanie niskonapięciowego procesora – układu o zmniejszonym taktowaniu pozwalający na uzyskanie mniejszego zużycia energii elektrycznej przy jednoczesnym ograniczeniu wydzielanego podczas pracy ciepła - oraz implementacja różnych sposobów transmisji danych (wybór odpowiedniego interfejsu komunikacyjnego w celu optymalizacji stosunku czasu pracy do maksymalnych dostępnych parametrów pomiarowych, w tym czasu próbkowania), użycie wbudowanego lub zewnętrznego konwertera analogowo-cyfrowego jak również stosowanie wymiennych elementów, takich jak wymienne przetworniki drgań mechanicznych pozwalają na dostosowanie konfiguracji urządzenia do warunków, w których urządzenie ma wykonywać akwizycję danych, jak również parametry żądanych pomiarów,
3. Technologia bezprzewodowa – wykorzystanie technologii bezprzewodowej umożliwia stosowanie urządzenia w miejscach trudno dostępnych takich jak:
  - a) Miejsca, do których człowiek nie ma dostępu w trakcie pracy maszyny we względu na warunki, które mogą powodować zagrożenie życia lub zdrowia – między innymi klatki bezpieczeństwa, w obrębie których pracują roboty lub nowoczesne urządzenia sterowane numerycznie,
  - b) Warunki o zmiennej temperaturze i wilgotności – między innymi na urządzeniach pracujących przy wydobywaniu surowców lub urządzeniach pokonujących znaczące dystanse, jak np. samochód,
  - c) Miejsca narażone na działanie płynów, w tym płynów chłodniczych lub olejów smarujących – urządzenia posiadające otwarte systemy smarowania.

Jednym z priorytetów prezentowanego sposobu jest inteligentne zarządzanie zużyciem energii elektrycznej przez urządzenia pomiarowe korzystając z powszechnie znanych stanów, w których urządzenie może się znajdować w celu wydłużenia czasu pracy na baterii, które jest realizowane w dwojaki sposób:

- a) Zmniejszenie zużycia prądu w stanie aktywności tj. akwizycji, przetwarzania i przesyłania danych pomiarowych
- b) Zmniejszenie czasu aktywności – przełączanie w stan wstrzymania

Dzięki powszechnie znanym algorytmom przełączania urządzenia pomiędzy stanami uśpienia i aktywności możliwe jest ograniczenie zużycia energii. Sposób wynalazku zakłada dostępność następujących stanów pracy:

1. Stan aktywności: akwizycja i przetwarzanie sygnałów jest wykonywane z zadaną częstotliwością, system jest aktywny dopóki trwają zadania związane z trwającą sesją pomiarową,
2. Stan wstrzymania: aktywowany w przypadku zakończenia sesji pomiarowej, wszystkie aktywne operacje związane z akwizycją danych są wstrzymane, protokół komunikacyjny przechodzi w stan wstrzymania, wysyłając ograniczoną ilość pakietów mającą na celu zapewnienie możliwości przesyłu danych w przypadku wykrycia aktywności monitorowanego systemu,
3. Stan głębokiego snu: aktywowany, gdy urządzenie znajduje się w stanie wstrzymania i nie ma zaplanowanych kolejnych sesji pomiarowych, wszystkie aktywne operacje związane z akwizycją danych są wstrzymane, protokół komunikacyjny przechodzi w stan wstrzymania, wysyłając jedynie pakiety mające na celu podtrzymanie połączenia z urządzeniem host,
4. Stan hibernacji: w momencie wykrycia niskiego poziomu baterii urządzenie przechodzi w stan hibernacji mający na celu ograniczenie do minimum ilość energii zużywanej przez system, wszystkie operacje związane z akwizycją, przetwarzaniem danych i komunikacją zostają całkowicie wstrzymane.

### **6.3 Czym tak naprawdę jest przedstawiony system, czyli klasyfikacja układu**

Przedmiotem otrzymanego wynalazku (patent nr PL 238650 B1 „Sposób i urządzenie pomiarowo-transmisyjne do akwizycji danych pomiarowych układów mechatronicznych”) jest innowacyjny sposób akwizycji i bezprzewodowej transmisji danych za pomocą urządzenia o niewielkich rozmiarach. Ten nowatorski wynalazek pozwala na pobieranie danych diagnostycznych z urządzeń, dla których stosowanie tradycyjnych systemów akwizycji danych było do tej pory niemożliwe z różnych powodów, takich jak duży rozmiar systemów, brak możliwości poprowadzenia odpowiedniego okablowania czy trudne warunki środowiskowe, takie jak zmienna temperatura i wilgotność powietrza.

Wprowadzenie bezprzewodowego interfejsu komunikacyjnego stanowi przełomowy krok w dziedzinie monitorowania i akwizycji danych diagnostycznych. Dzięki temu rozwiązaniu możliwe jest znaczne zredukowanie ilości niezbędnego okablowania potrzebnego do

akwizycji danych diagnostycznych, co ma istotne znaczenie zwłaszcza w przypadku późniejszego serwisowania obiektu badań, gdzie konieczne jest demontowanie systemu pomiarowego. Dotychczasowe systemy akwizycji danych często charakteryzowały się dużymi rozmiarami, co utrudniało ich zastosowanie w przypadku urządzeń o ograniczonej przestrzeni montażowej lub mobilnych aplikacji. Wraz z postępem technologicznym i miniaturyzacją elektroniki, wynalazek skoncentrował się na stworzeniu kompaktowego urządzenia, które może być łatwo zamontowane w różnych typach urządzeń. Dzięki temu, nawet w przypadku ograniczonej przestrzeni montażowej, możliwe jest efektywne i precyzyjne pobieranie danych diagnostycznych. Kolejnym czynnikiem, który przyczynił się do rozwoju tego wynalazku, jest brak możliwości poprowadzenia tradycyjnego okablowania w niektórych sytuacjach. W niektórych przypadkach urządzenia są rozlokowane na dużym obszarze lub w trudno dostępnych miejscach, co utrudnia instalację kabli. Dzięki bezprzewodowej transmisji danych, wynalazek umożliwia elastyczną instalację urządzenia monitorującego bez konieczności prowadzenia kabli na duże odległości. Ponadto, warunki środowiskowe, takie jak zmienne temperatury i wilgotność powietrza, często stwarzają trudności w utrzymaniu stabilnej transmisji danych. Tradycyjne okablowanie może być podatne na uszkodzenia spowodowane zmianami warunków atmosferycznych. Dzięki bezprzewodowej transmisji, wynalazek eliminuje te problemy, umożliwiając niezawodną i ciągłą komunikację pomiarową niezależnie od warunków środowiskowych. Wprowadzenie tego wynalazku otwiera nowe możliwości w dziedzinie monitorowania stanu urządzeń. Dzięki bezprzewodowej akwizycji danych diagnostycznych, inżynierowie i operatorzy mogą łatwo monitorować i analizować stan urządzeń w czasie rzeczywistym. Jest to szczególnie istotne w przypadku urządzeń takich jak nowoczesne obrabiarki sterowane numerycznie, gdzie precyzyjne i niezawodne monitorowanie jest kluczowe dla efektywności i bezpieczeństwa procesów produkcyjnych.

Również w przypadku ewentualnych modyfikacji maszyny, czy to w wyniku modernizacji, rozbudowy lub innych zmian, system akwizycji danych w oparciu o opisywany wynalazek umożliwia wprowadzanie zmian w sposób szybszy i bardziej elastyczny niż w przypadku klasycznych systemów diagnostycznych. Dzięki modułowej budowie i elastycznym interfejsom, modyfikacje w systemie akwizycji danych mogą być łatwo wprowadzane, co przyspiesza proces adaptacji do nowych warunków pracy maszyny. Budowa uproszczonego systemu do diagnostyki wibracyjnej przyczynia się również do możliwości zwiększenia liczby monitorowanych obiektów. Dzięki mniejszym rozmiarom, łatwej instalacji i bezprzewodowej transmisji danych, system staje się bardziej dostępny dla

różnych typów urządzeń. Dotychczasowe systemy diagnostyczne często były kosztowne i wymagały skomplikowanych instalacji, co ograniczało ich zastosowanie do wybranych maszyn lub ograniczonej liczby monitorowanych parametrów. W przypadku obrabiarek sterowanych numerycznie, które są kluczowymi urządzeniami w przemyśle, dostępność i łatwość instalacji systemu pomiarowego miały duże znaczenie. Wprowadzenie uproszczonego systemu monitorowania stanu umożliwia zbieranie danych diagnostycznych z większej liczby maszyn, co przyczynia się do zwiększenia ilości dostępnych informacji o stanie urządzeń oraz możliwości wczesnego wykrywania i zapobiegania awariom.

Warto zaznaczyć, że monitorowanie obiektów takich jak obrabiarki sterowane numerycznie jest często trudne ze względu na różne czynniki. Przede wszystkim, trudny dostęp do wewnętrznych elementów maszyny i ograniczona przestrzeń utrudniają instalację tradycyjnego okablowania. Ponadto, obecność otwartego chłodzenia materiału obrabianego wprowadza dodatkowe wyzwania, takie jak obecność czynników zewnętrznych (np. opiłków) oraz zmienne warunki środowiskowe (np. temperatura, wilgotność). Wprowadzenie bezprzewodowego systemu monitorowania stanu pozwala na skuteczne przekraczanie tych przeszkód, eliminując konieczność prowadzenia kabli wewnątrz maszyny i umożliwiając monitorowanie w trudno dostępnych miejscach.

#### **6.4 Przykład zastosowania nowoczesnego systemu diagnostyki**

Głównym przedmiotem badań i analizy będą nowoczesne obrabiarki sterowane numerycznie, które pełnią kluczową rolę w przemysłowej obróbce materiałów. Obrabiarki tego typu są wykorzystywane do mechanicznej obróbki przedmiotów, nadając im określone kształty, wymiary oraz odpowiednią chropowatość powierzchni. Praca tych urządzeń opiera się na wykorzystaniu specjalistycznych narzędzi dostosowanych do konkretnego rodzaju obróbki. Jednak ze względu na specyficzne warunki pracy tych maszyn, konwencjonalne systemy akwizycji danych stają się niewystarczające.

Tradycyjne metody pomiarowe związane z przewodowym okablowaniem są niepraktyczne w przypadku obrabiarek sterowanych numerycznie. Istnieje wiele wyzwań, które utrudniają zastosowanie tradycyjnych urządzeń pomiarowych. Przede wszystkim brak możliwości poprowadzenia okablowania wewnątrz maszyny jest jednym z głównych ograniczeń. Ponadto, masywne urządzenia pomiarowe są nieodpowiednie do montażu na ruchomych elementach maszyny, a także nie są odporne na bryzgi wody, które są powszechnym elementem środowiska pracy obrabiarek. W tym kontekście przedstawiony wynalazek oferuje innowacyjne rozwiązanie, które pozwala na stworzenie miniaturowego

urządzenia akwizycji danych. Dzięki temu możliwe będzie umieszczenie czujnika drgań bezpośrednio na obudowie wrzeciona maszyny, które jest jednym z kluczowych elementów obrabiarki.

Zastosowanie przetwornika drgań wykonanego w technologii MEMS umożliwi osiągnięcie odpowiednio małych rozmiarów i niskiej wagi urządzenia. Ważnym aspektem prezentowanego sposobu jest również wykorzystanie bezprzewodowej transmisji danych, co znacząco ogranicza potrzebę stosowania skomplikowanego okablowania. Dzięki temu możliwe będzie zamontowanie urządzenia pomiarowego wewnątrz obrabiarki, bez konieczności obawy o przewody i ich ewentualne uszkodzenia. Bezprzewodowa transmisja danych zapewnia elastyczność i mobilność systemu, umożliwiając łatwą integrację z istniejącą infrastrukturą technologiczną.

W celu zapewnienia niezawodności i trwałości urządzenia, istotnym aspektem jest odpowiednie zaprojektowanie obudowy, która spełnia wymagania szczelności określone przez normę IP67. Taka konstrukcja zabezpiecza urządzenie przed wilgocią, pyłem i innymi czynnikami zewnętrznymi, które mogą występować w środowisku pracy obrabiarek.

Kolejną istotną cechą przedstawionego rozwiązania jest modularność, co pozwala na wymianę przetwornika drgań w razie potrzeby. Dzięki tej funkcji możliwe jest dostosowanie parametrów pomiarowych urządzenia do konkretnych wymagań i specyfikacji obrabiarki.

Wprowadzenie opisanego sposobu pozwoli na precyzyjne i niezawodne monitorowanie drgań w obrabiarkach sterowanych numerycznie. Otrzymane dane diagnostyczne będą stanowiły cenną informację dla operatorów, umożliwiając wczesne wykrywanie awarii, optymalizację procesów produkcyjnych i zapewnienie wysokiej jakości wyrobów. Dodatkowo, wydłużony czas pracy urządzenia na baterii oraz łatwość montażu wewnątrz obrabiarki przyczynią się do efektywnego funkcjonowania systemu pomiarowego.

## **6.5 Korzystne skutki uzyskiwane z rozwiązania**

Rozdział opisujący korzystne skutki uzyskane z prezentowanego rozwiązania poświęcony jest analizie wpływu zastosowania miniaturowego, bezprzewodowego systemu akwizycji danych w obrabiarkach sterowanych numerycznie. Przedstawione w poprzednich rozdziałach innowacyjne podejście do pomiarów drgań w maszynach tego typu przynosi liczne korzyści, które zostaną teraz szczegółowo omówione.

Jedną z kluczowych zalet jest zwiększenie liczby monitorowanych obiektów dzięki większej dostępności systemu. Tradycyjne metody diagnostyki wibroakustycznej często wiążą się z wysokim kosztem instalacji systemu pomiarowego, co może ograniczać możliwość monitorowania większej liczby urządzeń. Miniaturowe systemy, ze względu na swoje kompaktowe rozmiary i łatwość montażu, eliminują tę barierę. Dzięki temu można rozszerzyć zakres monitorowanych obiektów, co przekłada się na bardziej wszechstronne i kompleksowe zarządzanie stanem maszyn. Kolejnym istotnym aspektem jest zwiększenie ilości dostępnych danych diagnostycznych. Urządzenia, dla których dotychczas nie były zbierane dane diagnostyczne z powodu wysokich kosztów instalacji systemów pomiarowych, teraz mogą być objęte monitorowaniem za pomocą miniaturowego systemu. Przykładem takich urządzeń są obrabiarki sterowane numerycznie, które ze względu na swoją specyfikę i trudny dostęp, były często pomijane podczas monitorowania. Miniaturowe systemy umożliwiają zbieranie danych diagnostycznych z tych maszyn, co pozwala na wczesne wykrywanie ewentualnych usterek, optymalizację procesów oraz zwiększenie niezawodności i trwałości urządzeń. Dodatkowo, miniaturowe systemy do diagnostyki wibroakustycznej oferują możliwość ciągłego monitorowania, co umożliwia szybką reakcję na pojawiające się problemy i zapobieganie awariom. Dzięki stałemu dostępowi do danych diagnostycznych, można śledzić zmiany w charakterystykach drgań i akustycznych, identyfikować nieprawidłowości oraz podejmować odpowiednie działania naprawcze w odpowiednim czasie. To przyczynia się do minimalizacji przestojów i kosztów związanych z naprawami oraz zwiększa efektywność i wydajność procesów produkcyjnych.

Wykorzystanie nowoczesnej elektroniki w miniaturowych systemach diagnostyki wibroakustycznej przynosi liczne korzyści, zarówno pod względem ekonomicznym, jak i diagnostycznym. Dzięki postępowi technologicznemu, nowoczesna elektronika jest bardziej dostępna i efektywna, co przekłada się na redukcję kosztów produkcji i implementacji miniaturowych systemów diagnostycznych. Wykorzystanie zaawansowanych komponentów elektronicznych, takich jak miniaturowe przetworniki drgań wykonane w technologii MEMS, pozwala na osiągnięcie niskich kosztów produkcji przy jednoczesnym zachowaniu wysokiej jakości pomiarów. Dodatkowo, miniaturyzacja systemów diagnostycznych umożliwia łatwą instalację na maszynach bez konieczności kosztownej modyfikacji ich konstrukcji. Niski poziom szumów własnych to kolejna korzyść wynikająca z zastosowania nowoczesnej elektroniki. Szumy własne to niepożądane zakłócenia generowane przez urządzenie pomiarowe, które mogą wpływać na dokładność i wiarygodność wyników diagnostycznych. Miniaturowe systemy wykorzystujące zaawansowane technologie elektroniczne i cyfrowe

przetwarzanie sygnałów mogą osiągnąć bardzo niski poziom szumów własnych. To ma kluczowe znaczenie podczas określania stanu technicznego maszyny, ponieważ pozwala na dokładne wykrywanie subtelnych zmian w charakterystykach drgań i akustycznych, które mogą wskazywać na potencjalne problemy. Niski poziom szumów własnych przyczynia się również do poprawy jakości danych diagnostycznych. Wiarygodne i precyzyjne pomiary są kluczowe dla skutecznego monitorowania i analizy stanu maszyn.

Dzięki modularyzacji, poszczególne elementy systemu mogą być projektowane, modyfikowane i rozwijane niezależnie od siebie. Możliwość rozbudowy poszczególnych modułów oraz wymiany ich pomiędzy różnymi systemami diagnozującymi przekłada się na większą elastyczność w dostosowaniu systemu do konkretnych potrzeb i architektur maszyn. Oznacza to, że moduły mogą być zoptymalizowane dla specyficznych zastosowań, co zwiększa efektywność diagnostyki i optymalizuje proces monitorowania. Modularyzacja systemu przyczynia się również do uproszczenia konstrukcji. Dzięki podziałowi systemu na moduły, każdy z nich może być projektowany w sposób optymalny, niezależnie od reszty systemu. To umożliwia redukcję skomplikowanych struktur i zwiększenie efektywności produkcji. Dodatkowo, standaryzacja podzespołów w ramach modułów przyczynia się do prostszego montażu, mniejszych kosztów produkcji i mniejszego zużycia energii. Szybszy montaż jest kolejnym korzyścią modularyzacji systemu. Moduły mogą być przygotowane z wyprzedzeniem i z łatwością montowane w odpowiednich miejscach na maszynach. Proces instalacji staje się bardziej efektywny i mniej czasochłonny, co ma pozytywny wpływ na wydajność całego procesu diagnostycznego.

Dzięki miniaturyzacji, miniaturowe systemy diagnostyczne stają się bardziej kompaktowe i poręczne, co znacznie zwiększa ich dostępność i elastyczność zastosowania. Dotychczasowe urządzenia diagnostyczne były często duże, skomplikowane i wymagały specjalistycznych warunków instalacji. W rezultacie, wiele maszyn, szczególnie te o ograniczonej przestrzeni montażowej, nie były objęte regularnym monitorowaniem stanu technicznego. Jednak dzięki miniaturyzacji, miniaturowe systemy diagnostyczne można łatwo zamontować nawet w trudno dostępnych miejscach, co otwiera nowe możliwości diagnostyki dla szerokiego spektrum maszyn. Zastosowanie miniaturyzowanego systemu diagnostycznego ma szczególne znaczenie w przypadku maszyn o skomplikowanej konstrukcji, ograniczonej przestrzeni montażowej lub maszyn mobilnych. Na przykład, w branży motoryzacyjnej, gdzie miejsce montażu jest często ograniczone, tradycyjne urządzenia diagnostyczne mogą być trudne do zainstalowania lub mogą powodować znaczne zmiany w strukturze pojazdu. Dzięki

miniaturyzacji, miniaturowe systemy diagnostyczne można łatwo zintegrować w istniejących komponentach pojazdu, co umożliwia monitorowanie stanu technicznego na bieżąco, nawet podczas normalnej eksploatacji. Ponadto, miniaturyzacja systemu diagnostycznego umożliwia wykonywanie pomiarów w miejscach, które wcześniej były niedostępne lub trudno dostępne dla tradycyjnych urządzeń diagnostycznych. Na przykład, w przypadku maszyn złożonych, gdzie istnieje wiele skomplikowanych układów i części, tradycyjne urządzenia diagnostyczne mogą mieć trudności z dotarciem do kluczowych punktów pomiarowych.

Obecność przewodów wewnątrz urządzeń często jest problematyczna, zwłaszcza w przypadku maszyn o skomplikowanej strukturze lub ograniczonej przestrzeni montażowej. Wiele urządzeń nie jest zaprojektowanych z myślą o instalacji przewodowej, a prowadzenie kabli wewnątrz takich urządzeń jest trudne, kosztowne i czasochłonne. W takich sytuacjach wykorzystanie komunikacji bezprzewodowej staje się kluczowe. Miniaturowe systemy diagnostyczne, wyposażone w moduły komunikacji bezprzewodowej, mogą być łatwo zamontowane i zintegrowane z różnymi urządzeniami, eliminując potrzebę prowadzenia skomplikowanych instalacji przewodowych. Dodatkowo, zmienne warunki środowiskowe mogą utrudniać wykorzystanie tradycyjnych systemów diagnostycznych. Na przykład, w przypadku maszyn pracujących w ekstremalnych warunkach, takich jak wysoka temperatura, wilgotność lub obecność substancji chemicznych, przewody i przewodniki mogą ulegać uszkodzeniom, co wpływa na niezawodność i dokładność pomiarów. Wykorzystanie komunikacji bezprzewodowej pozwala uniknąć tych problemów, umożliwiając bezpieczne i niezawodne monitorowanie stanu technicznego urządzeń nawet w trudnych warunkach środowiskowych.

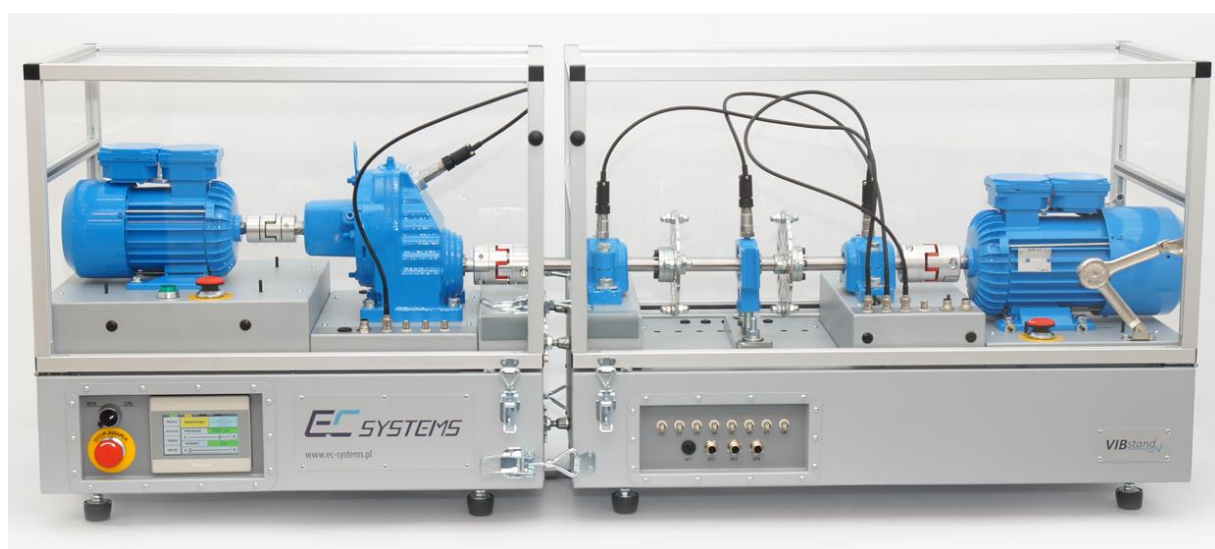
Tradycyjne systemy diagnostyczne często były skomplikowane i drogie, co ograniczało liczbę punktów pomiarowych, które można było monitorować. Jednak dzięki prostocie i redukcji kosztów opracowywanego systemu, można zainstalować większą liczbę sensorów w różnych częściach maszyny. To z kolei pozwoli na zbieranie bardziej szczegółowych danych dotyczących stanu technicznego, a tym samym umożliwi opracowanie bardziej precyzyjnych i zaawansowanych metod wczesnego ostrzegania. Zwiększenie ilości potencjalnych punktów pomiarowych w systemie diagnostycznym otwiera nowe możliwości w dziedzinie monitorowania. Dzięki większej liczbie sensorów, które mogą być rozmieszczone w różnych obszarach maszyny, możliwe jest dokładniejsze i wszechstronne śledzenie parametrów pracy oraz identyfikacja nawet subtelnych zmian, które mogą wskazywać na potencjalne problemy. To z kolei pozwoli na rozwinięcie zaawansowanych algorytmów analizy danych i

opracowanie nowoczesnych metod diagnostycznych, które będą bardziej precyzyjne, skuteczne i reaktywne na zmiany stanu technicznego maszyny. Opracowanie nowoczesnych metod diagnostycznych ma ogromne znaczenie dla zapewnienia wysokiej niezawodności i efektywności maszyn w różnych branżach. Będzie można wykryć nieprawidłowości i potencjalne awarie na etapie wczesnego ich rozwoju, co pozwoli na podjęcie odpowiednich działań naprawczych zanim dojdzie do poważniejszych uszkodzeń lub zatrzymania maszyny.

## 7 Pomiary i walidacja prototypów

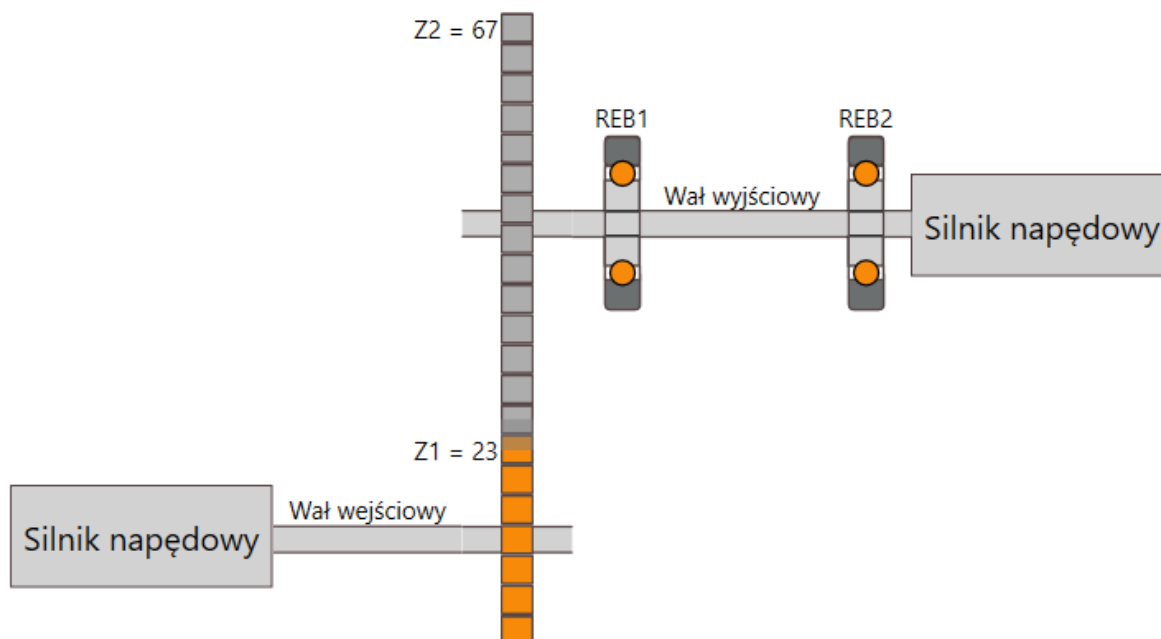
### 7.1 Opis stanowiska badawczego

Stanowisko badawcze użyte do przeprowadzenia eksperymentu przedstawiono na Rys. 7.1. Vibstand 2 składa się z silnika elektrycznego o mocy 1.1 kW, wału wejściowego, 1-stopniowej przekładni równoległej, trzech łożysk tocznych oraz silnika hamującego. Przekładnia napędowa jest reduktorem o kołami zębatymi – odpowiednio 23 zęby oraz 67 zębów, co daje redukcję prędkości o współczynnik  $23/67 = 0,343$ . Łożyska zamontowane na stanowisko to dwurzędowe łożyska baryłkowe 22202EK (łożysko środkowe stosowane do zadawania obciążenia poprzecznego widoczne na zdjęciu nie było użyte). Dodatkowo, wał wyjściowy ma możliwość zamontowania dodatkowych tarcz wprowadzających niewyrównoważenie wału wyjściowego.



Rysunek 7.1: Stanowisko badawcze Vibstand 2 (laboratorium AGH, WIMiR)

Stanowisko badawcze pozwala na wprowadzanie i badanie różnych uszkodzeń maszyny wirnikowej, w tym niewyrównoważenia, rozosiowania, uszkodzenia przekładni zębatej oraz uszkodzenia łożysk tocznych, zarówno lokalne, jak i rozproszone. Stanowisko badawcze Vibstand może pracować w przedziale prędkości obrotowej 1000 RPM – 3000 RPM, z obciążeniem w skali od 0% do 100%, co odpowiada momentowi obrotowemu równemu około 1.6 Nm. Układ kinetostatyczny maszyny przedstawiono na Rys. 7.2.



Rysunek 7.2: Schemat kinetostatyczny urządzenia Vibstand 2

Vibstand 2 jest sprawdzonym, doświadczalnym stanowiskiem badawczym, przeznaczonym do badania drgań. Stanowisko umożliwia obserwację drgań. Stanowisko umożliwia pomiar drgań w wybranych punktach, m.in. na obudowach łożysk oraz na przekładni. Obudowy łożysk wyposażone są w odpowiednie podstawki przystosowane do montażu czujników drgań. Instalacja elektryczna oprócz zasilania silników pełni funkcję regulacyjną. Pracą stanowiska steruje dotykowy panel HMI, na którym zaimplementowano opisowo-graficzny interfejs użytkownika. Komendy wydawane przez użytkownika kierowane są do falowników, których zadaniem jest odpowiednie sterowanie silnikiem napędzającym i obciążającym. Zastosowane w urządzeniu zaawansowane falowniki firmy ABB pozwalają na częściowy odzysk energii z silnika obciążającego, dzięki czemu zminimalizowano pobór energii z sieci zasilającej przy zachowaniu pełnej funkcjonalności w szerokim zakresie prędkości obrotowych i momentu obciążającego. Stanowisko może być sterowane lokalnie przy użyciu panelu operatorskiego lub zdalnie, ze współpracującego komputera.

## 7.2 Uszkodzenie 1 – niewyrównoważenie wału wyjściowego

W ramach prowadzonych badań w stanie niewyrównoważenia zarejestrowano sygnały drgań z obudowy łożyska oznaczonego na Rys. 7.2 symbolem REB1, umieszczonym na wale wyjściowym. Do tarczy maszyny wkręcono śruby M8, które dla zadanej prędkości

generowały dodatkową siłę powodującą zwiększenie drgań maszyny do poziomu awaryjnego VRMS zgodnie z normą ISO 10816-3.



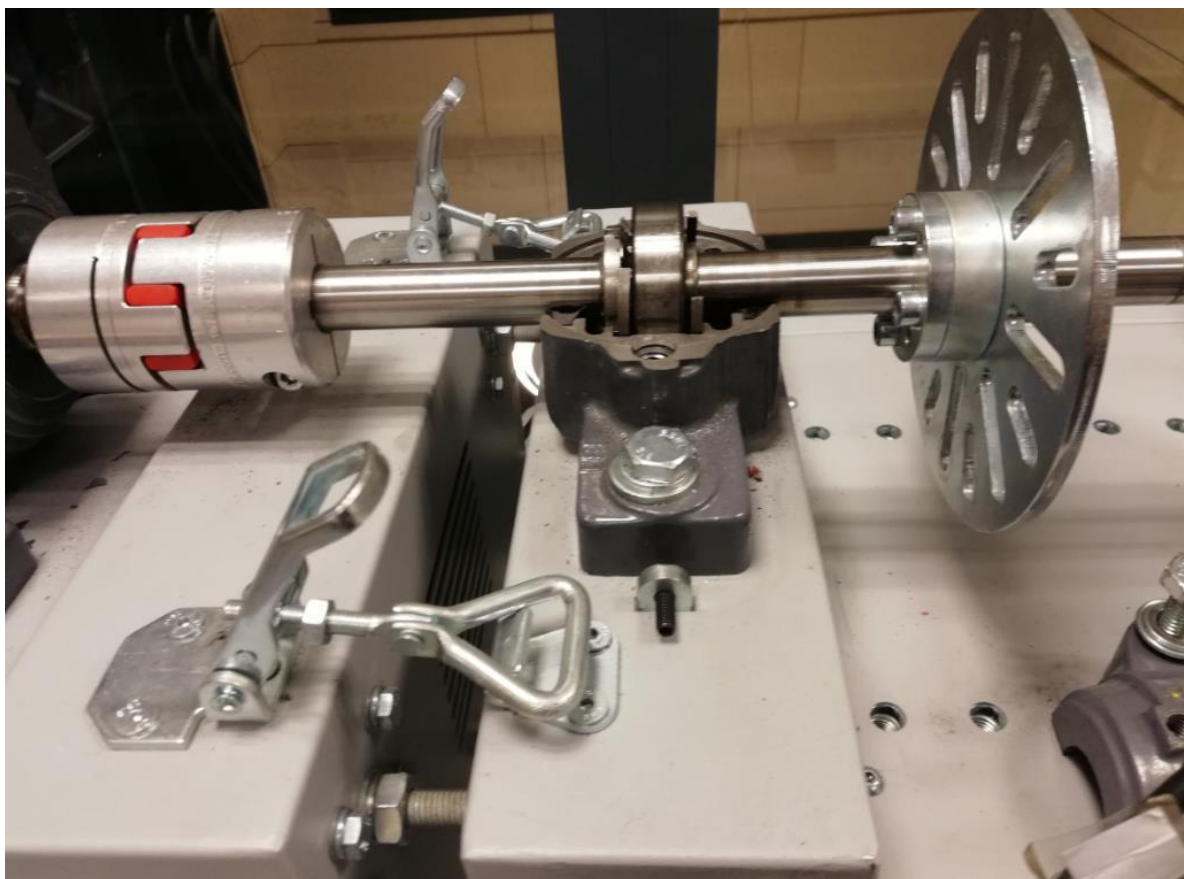
*Rysunek 7.3: Tarcza ze śrubami wprowadzającymi niewyrównowazenie na wale wyjściowym stanowiska*

Uszkodzenie w postaci niewyrównowazenia zilustrowane na Rys. 7.3 wykonano autonomicznie, tj. przeprowadzono je dla obiektu z łożyskami w poprawnym stanie

technicznym, a następnie uszkodzenie usunięto w celu przeprowadzenia dalszych badań na uszkodzonym łożysku.

### 7.3 Uszkodzenie 2 – łożysko toczne

W ramach prowadzonych badań rejestrowano sygnały drgań z obudowy łożyska oznaczonego na Rys. 7.2 symbolem REB1, umieszczonym na wale wyjściowym. Demontaż łożyska konieczny do wprowadzenia uszkodzenia przedstawiono na Rys 7.4, a uszkodzenie bieżni wewnętrznej łożyska przedstawiono na Rys. 7.5.



*Rysunek 7.4: Demontaż łożyska tocznego na stanowisku Vibstand 2*

W celu zapewnienia poprawności ponownego montażu, wykonano testy kontrolne przed i po demontażu porównując rozkład widmowy sygnałów drgań. Testy wykazały zmianę energii komponentów charakterystycznych dla bieżni zewnętrznej łożyska BPFO (z ang. *Ball Passing Frequency of the Outer Race*). Zmiana szerokopasmowych wartości RMS dla porównywanych przebiegów była pomijana (co jest typowe dla uszkodzeń łożyska tocznych we wczesnej fazie rozwoju [2.1]).



*Rysunek 7.5: Uszkodzenie lokalne bieżni zewnętrznej łożyska tocznego*

Na Rys. 7.5 przedstawiono uszkodzenie lokalne bieżni zewnętrznej dwurzędowego łożyska tocznego 22202EK. Uszkodzenie ma średnicę około 2 mm i głębokość około 0,5 mm.

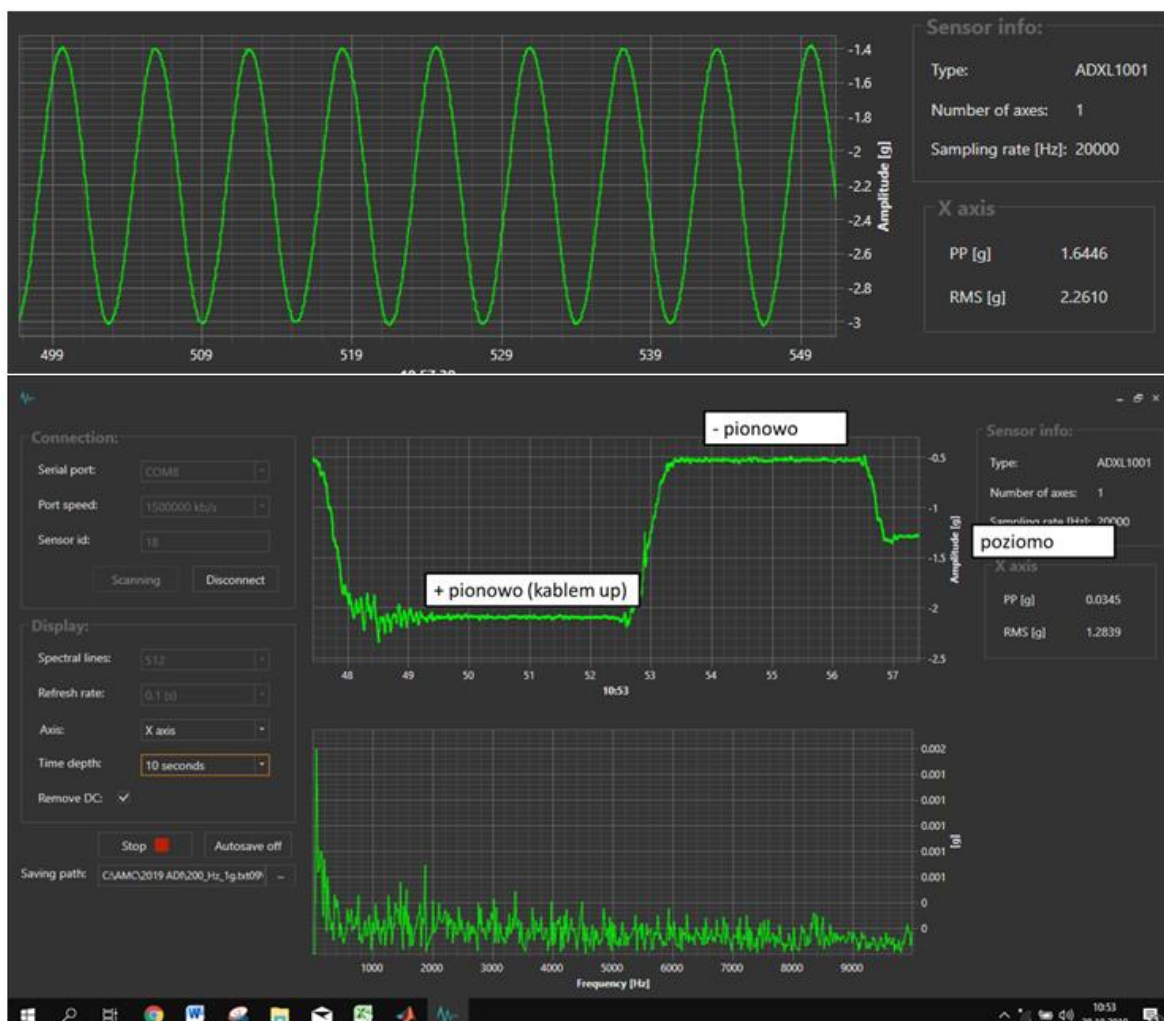
#### **7.4 Aparatura pomiarowa**

Pomiary sygnałów drgań wykonano z wykorzystaniem dwóch różnych aparatów badawczych. Sygnał referencyjny zarejestrowano z wykorzystaniem czujnika PCB 356A16 oraz standardowej, wielokanałowej jednostki akwizycji SCADAS. Sygnał badany rejestrowano za pomocą prototypowego czujnika MEMS oraz dedykowanej, prototypowej jednostki akwizycji wraz z prototypową wersją oprogramowania, co schematycznie przedstawiono na Rys. 7.6.



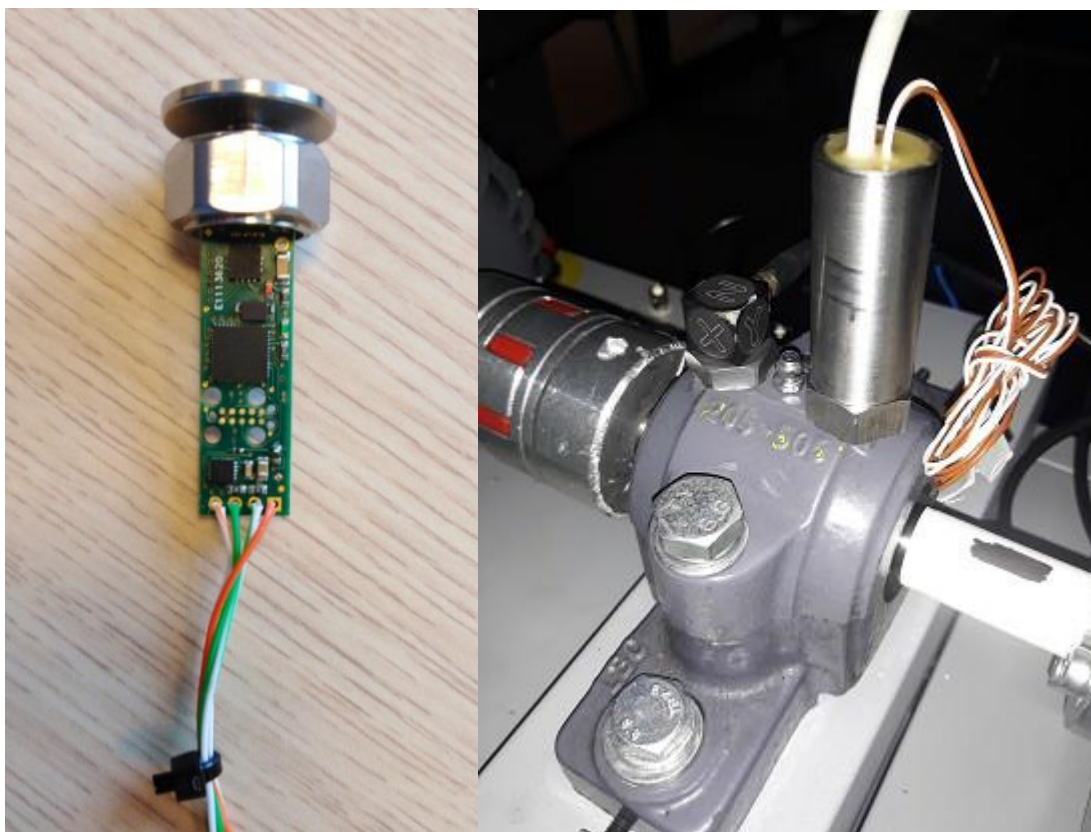
*Rysunek 7.6: Schemat prototypowego układu akwizycji danych*

Jak przedstawiono na Rys. 7.6, prototypowy układ jest prosty w użyciu, gdyż w przeciwieństwie do układów klasycznych nie posiada jednostki akwizycji, która nie jest konieczna w przypadku przesyłania danych w formacie cyfrowym. Co istotne, zaproponowane rozwiązanie pozwala na wybór przesyłania danych surowych oraz danych skalarnych. Przykładowy widok środowiska do rejestracji oraz wizualizacji sygnału drgań przedstawiono na Rys. 7.7. Zilustrowany system używa protokołu RS 485 do komunikacji sensora z komputerem osobistym.



Rysunek 7.7: Przykładowy widok aplikacji do wizualizacji drgań z prototypowego czujnika MEMS

Prototypowy czujnik drgań bez obudowy oraz montaż obu czujników drgań na obudowie łożyska przedstawiono na Rys. 7.8. Czujnik prototypowy został przymocowany śrubowo połączeniem M6, natomiast czujnik referencyjny został zamocowany używając wosku.



*Rysunek 7.8: Montaż prototypu i czujnika referencyjnego na obudowie łożyska. Lewa: Prototypowy czujnik bez obudowy, prawa: montaż czujników na obudowie łożyska*

Jak przedstawiono na Rys. 7.8, bliskie umiejscowienie obu czujników zapewnia porównywalną funkcję przejścia dla obu sygnałów rejestrowanych równolegle (tj. dokładność ręczna dla sygnałów 10-sekundowych).

### **7.5 Plan pomiarów**

W ramach pomiarów rejestrowane były sygnały o długości 10s z częstotliwością próbkowania charakterystyczną dla indywidualnych jednostek akwizycji. W przypadku układu prototypowego było to odpowiednio 40 kHz dla czujnika HF oraz 4 kHz dla czujnika LF, natomiast referencyjny układ pomiarowy rejestrował sygnał drgań z częstotliwością próbkowania 20 kHz. Sygnały zostały zarejestrowane dla maszyny rozgrzanej (po około 10 min) pracy w warunkach maksymalnej prędkości (3000 RPM) oraz przy połowie obciążenia nominalnego. W ramach dokonywanych analiz, porównywane były albo pojedyncze przebiegi albo uśrednienia 10-krotne.

## 7.6 Analiza wyników

### 7.6.1 Poziom szumów

W Tabeli 7.1 przedstawiono porównanie poziomu wartości skutecznej RMS (z ang. *Root Mean Square*) przyspieszenia drgań szumów dla prototypowego czujnika w wersji HF (z ang. *High Frequency* – o paśmie przenoszenia do 20 kHz), czujnika w wersji LF (z ang. *Low Frequency* – o paśmie przenoszenia do 2 kHz). Wynik uśredniono 10-krotnie.

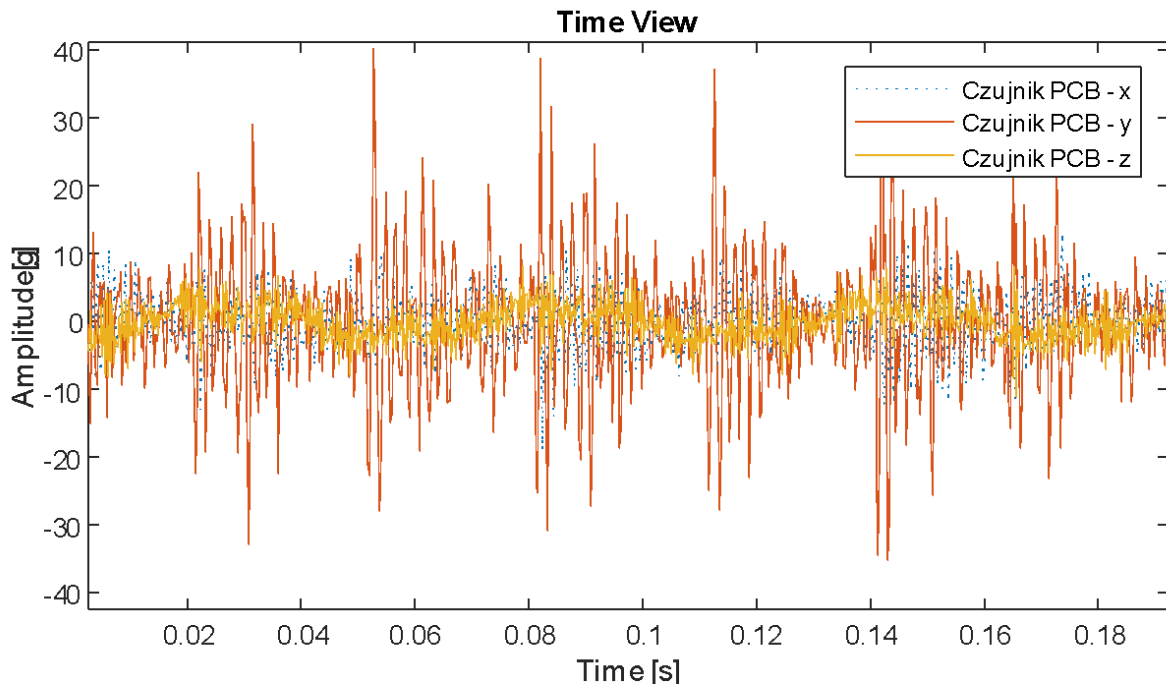
Tabela 7.1: Porównanie wartości RMS sygnału przyspieszenia szumu

	HF		LF		PCB
<b>RMS(y-mean(y))</b>	0.0033	RMS(x-mean(x))	0.0017	RMS(x-mean(x))	0.0042
		RMS(y-mean(y))	0.0021	RMS(y-mean(y))	0.0114
		RMS(z-mean(z))	0.0023	RMS(z-mean(z))	0.0069

Jak wynika z Tabeli 7.1, poziomy szumów we wszystkich osiach prototypowych czujników MEMS są niższe niż dla laboratoryjnych czujników referencyjnych, co jest wynikiem bardzo korzystnym, gdyż może umożliwiać pomiary jednym czujnikiem w różnych zakresie drgań – w przeciwieństwie do akcelerometrów, których typowy zakres czułości stosowany w przemyśle jest w przedziale 10 mVg-500mV/g.

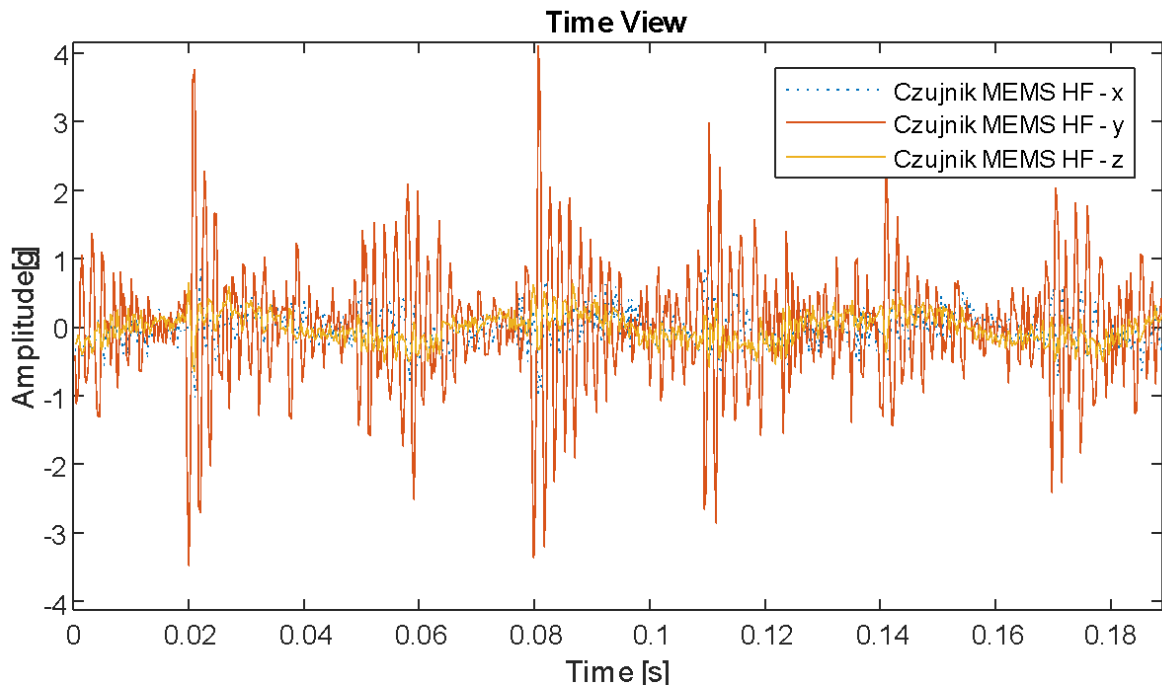
### 7.6.2 Niewyrównowazenie

Rys. 7.9 przedstawia referencyjny sygnał drgań zawierający komponent niewyrównowazenia dla czujnika PCB. Dla badanego sygnału, największe wartości obserwuje się dla osi Y (płaszczyzna poprzeczna do wału wyjściowego).



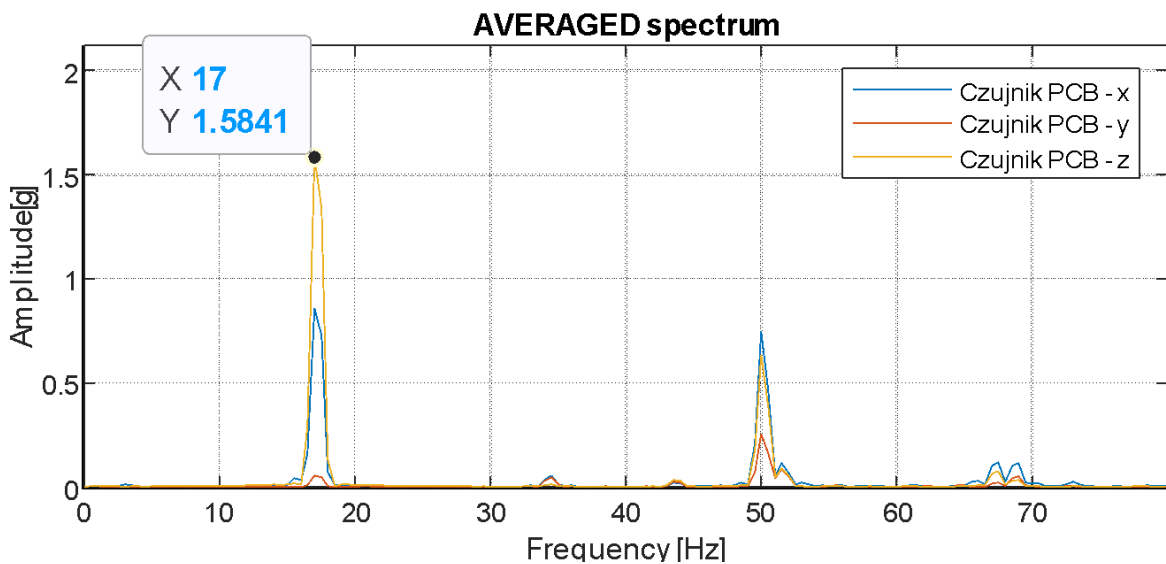
*Rysunek 7.9: Przebiegi czasowe dla czujnika referencyjnego PCB*

W analizowanym oknie czasowym – około 0.2 s obserwuje się około 6 cykli częstotliwościowych modulacji AM, co wskazuje na obecność komponentów w sygnale. Rys. 7.10 przedstawia analogiczne pomiary dla prototypowego czujnika MEMS, dla którego obserwuje się podobną modulację AM.



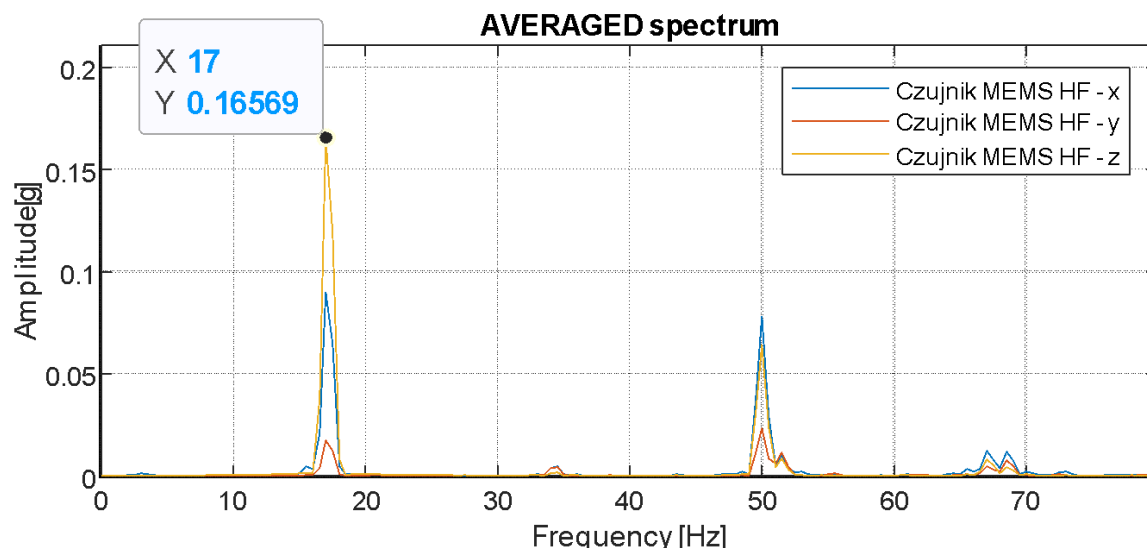
Rysunek 7.10: Przebiegi czasowe dla czujnika referencyjnego PCB

Na Rys. 7.11 przedstawiono uśrednione widmo częstotliwościowe dla sygnału referencyjnego z trzech osi XYZ. Widmo pokazuje komponent niewyrównoważenia wału wyjściowego (dla rozdzielczości  $1 \text{ Hz } 3000\text{RPM}/60 \cdot 23/67 = 17 \text{ Hz}$ ) oraz dodatkowo komponent wału wejściowego 50 Hz.



Rysunek 7.11: Widmo uśrednione z sygnałów z czujnika referencyjnego PCB

Na Rys. 12 przedstawiono uśrednione widmo częstotliwościowe dla sygnału drgań z prototypowego czujnika MEMS z trzech osi XYZ. Widmo pokazuje komponent niewyrównoważenia wału wyjściowego (dla rozdzielczości  $1 \text{ Hz } 3000\text{RPM}/60*23/67 = 17 \text{ Hz}$ ) oraz dodatkowo komponent wału wejściowego 50 Hz.



Rysunek 7.12: Widmo uśrednione z sygnałów z czujnika prototypowego MEMS

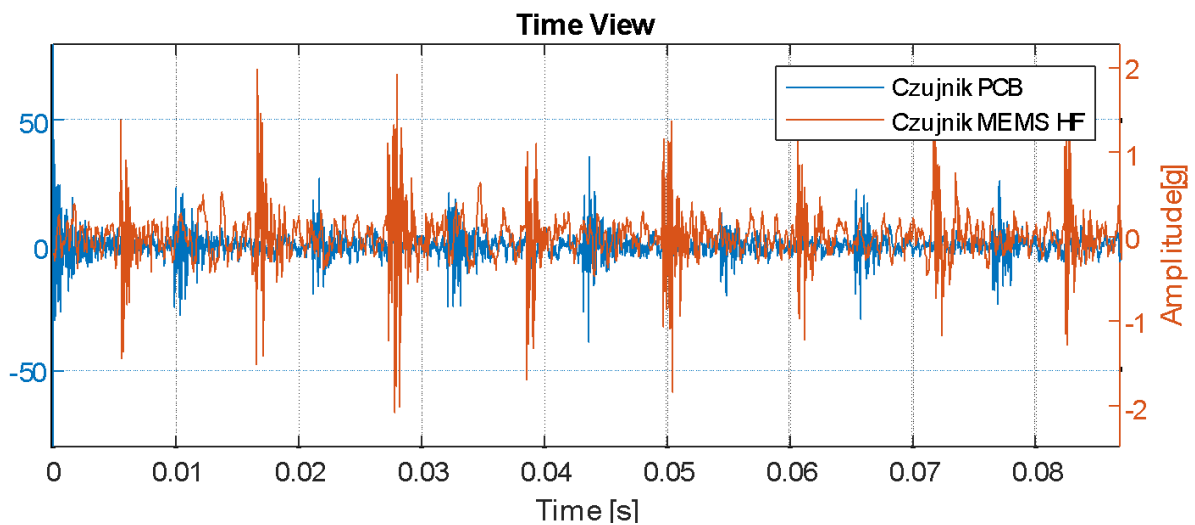
Komponent 1x (niewyrównoważenie wału wolnego) na uśrednionym widmie (okno 2s z 10s przebiegu, 50% overlap) jest praktycznie taki sam dla obu czujników. Ważne jest to, że w porównaniu z „otwartym” prototypem, kolejne osie x,y,z czujnika MEMS są znacznie lepiej rozróżnialne. Wartości VRMS (10Hz -1000 Hz) dla kolejnych kanałów z kolejnych czujników przedstawiono poniżej:

Tabela 7.2: Poziomy RMS dla czujnika referencyjnego i prototypowego w osiach XYZ

Kanał	Wartość VRMS ISO 10-1000 [mm/s]
Referencja PCB – oś x	76
Referencja PCB – oś y	19
Referencja PCB – oś z	135
MEMS – oś x	77
MEMS – oś y	20
MEMS – oś z	139

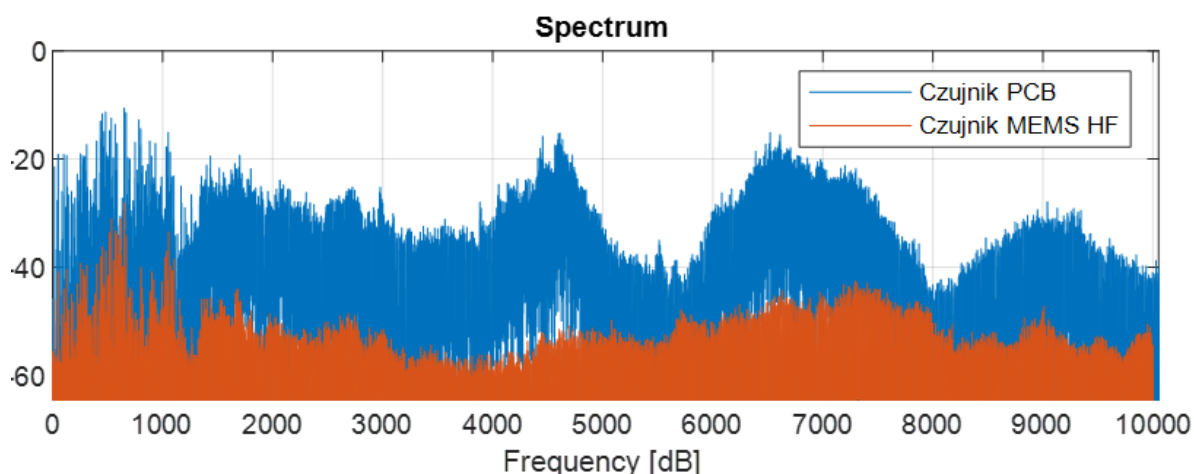
### 7.6.3 Uszkodzenie bieżni zewnętrznej REB

W ramach weryfikacji możliwości detekcji i identyfikacji uszkodzenia łożyska tocznego z wykorzystaniem prototypowego czujnika MEMS, zarejestrowano sygnał z obudowy łożyska. Jak ilustruje Rys. 7.13, przebiegi czasowe dla czujnika MEMS oraz dla czujnika referencyjnego zawierają wyraźne komponenty cyklostacjonarne 2-go rzędu pochodzące od uszkodzenia lokalnego bieżni zewnętrznej łożyska. Dla przedziału czasowego równego około 0.09 s widać 8 pulsów, a więc częstotliwość charakterystyczna BPFO wynosi poniżej 100 Hz, co dla prędkości 3000 RPM jest zgodne z oczekiwaniem ( $3000/60 \cdot 23/67 \cdot \text{BPFO}$ ).



Rysunek 7.13: Przebiegi czasowe sygnału referencyjnego i badanego

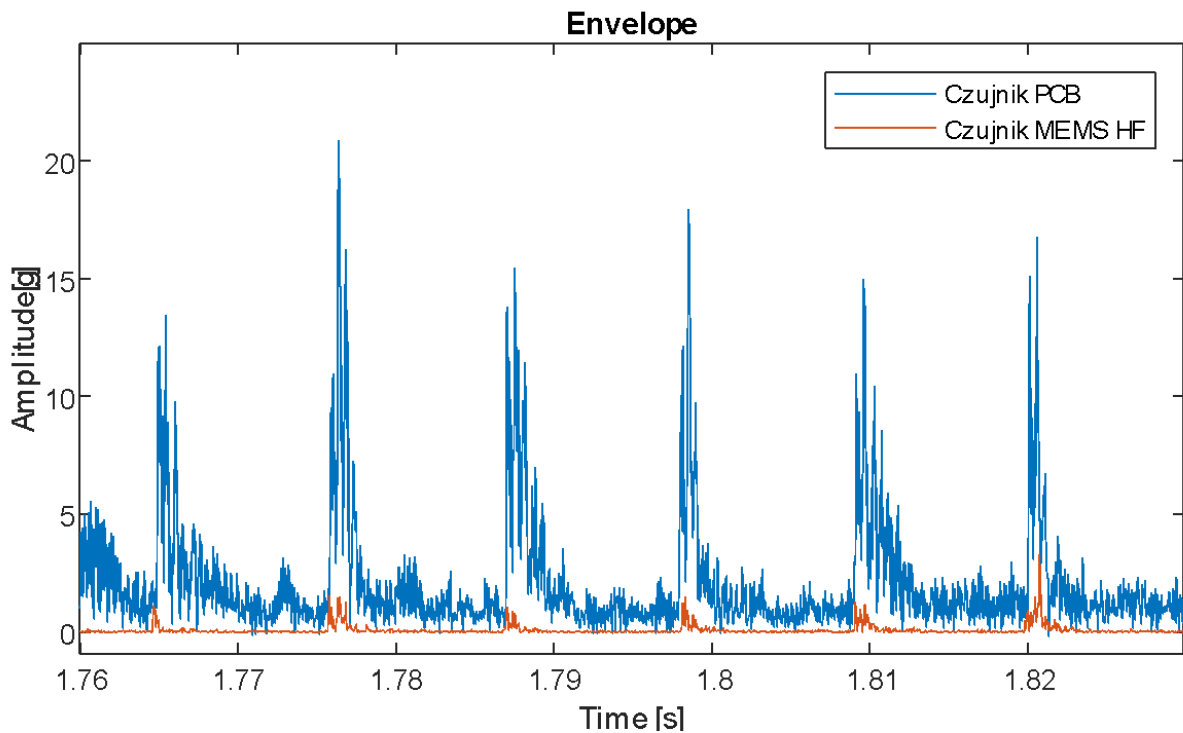
Na Rys. 7.14 pokazano widma sygnałów w skali dB dla czujnika referencyjnego oraz prototypowego MEMS.



Rysunek 7.14: Widma w pełnej rozdzielczości sygnału referencyjnego i sygnału badanego

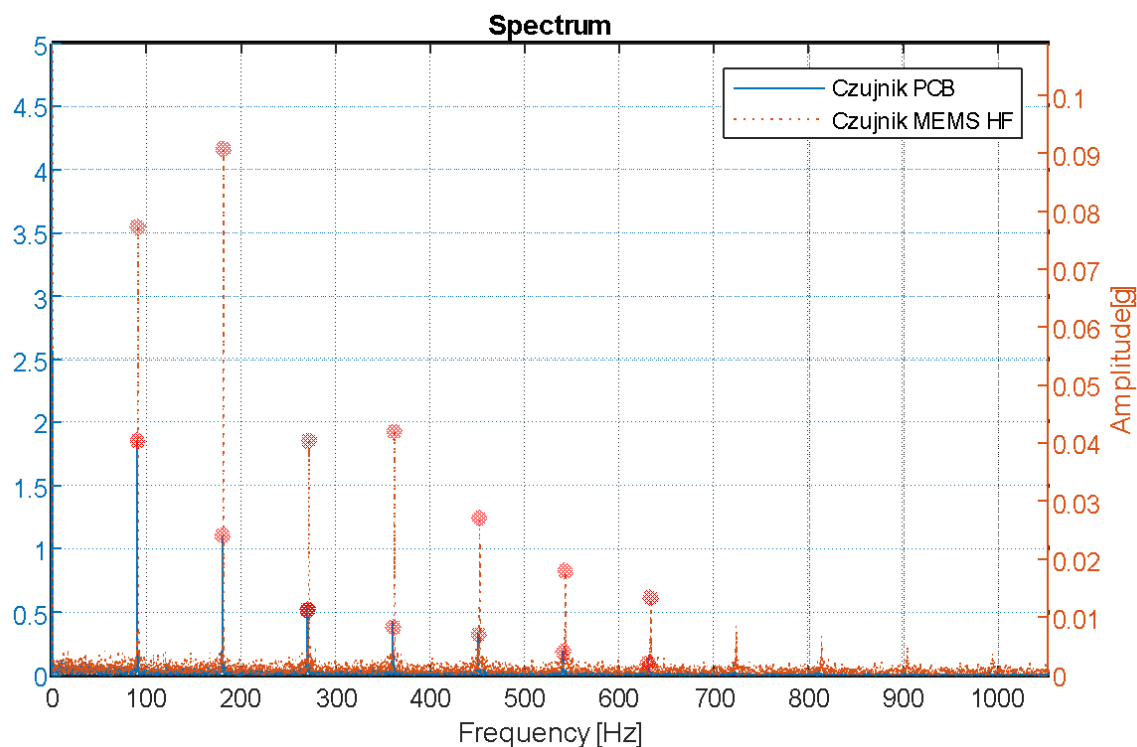
Widmo z czujnika MEMS jest mniej dynamiczne niż dla widma sygnału referencyjnego, ale co ważne, nie posiada żadnego krytycznego wzbudzenia, które mogłoby wskazywać na istotną wadę konstrukcyjną. Porównując wyniki z poprzednią konstrukcją, odpowiedź czujnika spadła średnio w mocy o 20dB. W ramach przeprowadzonych badań udało się zredukować dwa obszary częstotliwości strukturalnych (3-3.5 kHz oraz 9.5-10kHz), które były wyraźne dla pierwszych modeli MEMS prototypów.

Rysunek 7.15 ilustruje porównanie widma obwiedni dla sygnału referencyjnego oraz dla sygnału z czujnika MEMS. Sygnały zostały przesunięte w celu porównania częstotliwości impulsów. Sygnał obwiedni uzyskano wykorzystując dwa filtry w dziedzinie czasu (filtr HF: częstotliwość zaporowa 2000 Hz, filtr LF: częstotliwość zaporowa 500 Hz) oraz operację prostowania sygnału (*abs*).



Rysunek 7.15: Przebiegi czasowe sygnałów obwiedni dla czujnika referencyjnego i czujnika MEMS

Po transformacji FFT sygnału czasowego obwiedni oraz odpowiednim wyskalowaniu, uzyskano widma obwiedni przedstawione na Rys. 7.16. Widma przedstawiają kolejnych 10 harmoniczných częstotliwości charakterystycznej BPFO uszkodzenia łożyska tocznego wynoszącej 88.8 Hz dla prędkości nominalnej 3000 RPM dla wału wyjściowego (przełożenie 0,343).



Rysunek 7.16: Widma sygnału obwiedni dla czujnika referencyjnego oraz czujnika MEMS

W obu przypadkach, komponenty charakterystyczne są wyraźne, co dowodzi, że prototypowy czujnik MEMS nadaje się w porównywalnym stopniu do detekcji oraz identyfikacji wysokoczęstotliwościowych uszkodzeń łożysk tocznych co klasyczny akcelerometr laboratoryjny. Dodatkowo, Rys. 7.16 wskazuje, że stosunek sygnału do szumu (z ang. SNR – *Signal to Noise Ratio*) dla badanego czujnika MEMS wynosi około 20, natomiast dla referencyjnego czujnika PCB – około 15,5. Jednocześnie należy podkreślić, że poziom szumu dla czujnika MEMS ma charakterystykę bardziej płaską niż klasyczny akcelerometr, co korzystnie wpływa na reprezentację fizycznych drgań mechanicznych.

Przeprowadzone prace badawcze pozwalają stwierdzić, że zaprojektowany prototyp czujnika drgań oparty na technologii MEMS nadaje się do pomiaru drgań maszyn wirnikowych w częstotliwościach do 20 kHz, a tym samym wydaje się być odpowiedni do diagnostyki i identyfikacji uszkodzeń łożysk tocznych w typowych maszynach wirnikowych.

## 8 Przetwarzanie sygnałów dla cyfrowych czujników drgań wykorzystujących technologię MEMS

### 8.1 Analiza wymagań przemysłowych

Nowoczesne systemy monitorowania i diagnostyki maszyn pomagają wielu gałęziom przemysłu zwiększyć niezawodność maszyn i linii produkcyjnych. Głównym praktycznym oczekiwaniem od nowych systemów jest wykrycie wzrostów drgań oraz wskazanie, jaki komponent widmowy jest odpowiedzialny za wzrost. Pozornie, takie podejście brzmi podobnie do podejścia stosowanego przez ostatnie dekady, jednak rozbudowa systemów w ramach Industry 4.0 wyraźnie zmieniła stosunek wnioskowania automatycznego (nienadzorowanego) do wnioskowania nadzorowanego. W nowoczesnym podejściu, kluczowa jest możliwość wstępnej analizy bardzo dużej ilości danych na poziomie podstawowym, która kierkuje dalsze działania eksperckie. W opisanym schemacie, sprzężenie wiedzy o elementach maszyny z modelem rozwoju uszkodzenia (z ang. *Failure Mode*) jest wykonywane przez osobny moduł analizy danych, którego zadaniem jest wskazanie zakresu koniecznych działań dla służb utrzymania ruchu, korzystnie zawierający również narzędzia predykcyjne. Duża ilość danych wymaga możliwości instalowania dużej ilości czujników, zarówno przewodowych, jak i bezprzewodowych, które mogą komunikować się w ramach jednego systemu. Jak opisano w pracy, z uwagi na korzystne charakterystyki mechaniczne, w ostatnich latach przemysł systemów diagnostyki technicznej zwrócił szczególną uwagę na technologię MEMS (z ang. *Micro-Electro-Mechanical-System*), która zapewnia porównywalne możliwości odczytu drgań mechanicznych przy znacznie niższym poborze mocy w porównaniu do układów starszej generacji. W pracy doktorskiej przedstawiono szereg prac badawczych, które umożliwiły opracowanie konkurencyjnego rozwiązania w tym zakresie. W dalszej perspektywie, z uwagi na cyfrowe wyjścia nowych czujników, wdrażanie czujników drań opartych na technologii MEMS może przyczynić się do unifikacji danych przesyłanych w obrębie całego systemu, co znacznie ułatwi jego skalowalność. Co istotne, skalowalność infrastruktury czujników może być wykonana niezależnie od modułów analizy i predykcji, takich jak np. nowoczesny algorytm redukcyjno-redundantny dla rozproszonych systemów skalowalnych, umożliwiający szybką analizę względnie dużej ilości danych drganiowych [8.1].

Jedną z kluczowych korzyści zastosowania cyfrowych czujników drgań jest możliwość auto-walidacji zbieranych danych przed przetworzeniem, zwłaszcza przed obliczeniem wskaźników skalarnych (z ang. *HealthIndicators* HI lub tzw. „cech”). Walidacja sygnałów

umożliwia znaczną redukcją fałszywych alarmów FAR (z ang. *False Alarm Rate*), co jest bardzo pożądaną cechą w rozproszonych systemach monitorowania i diagnostyki maszyn. Zasady walidacji dostosowane do nowych czujników wykonanych w technologii MEMS przedstawiono w Dodatku 6.

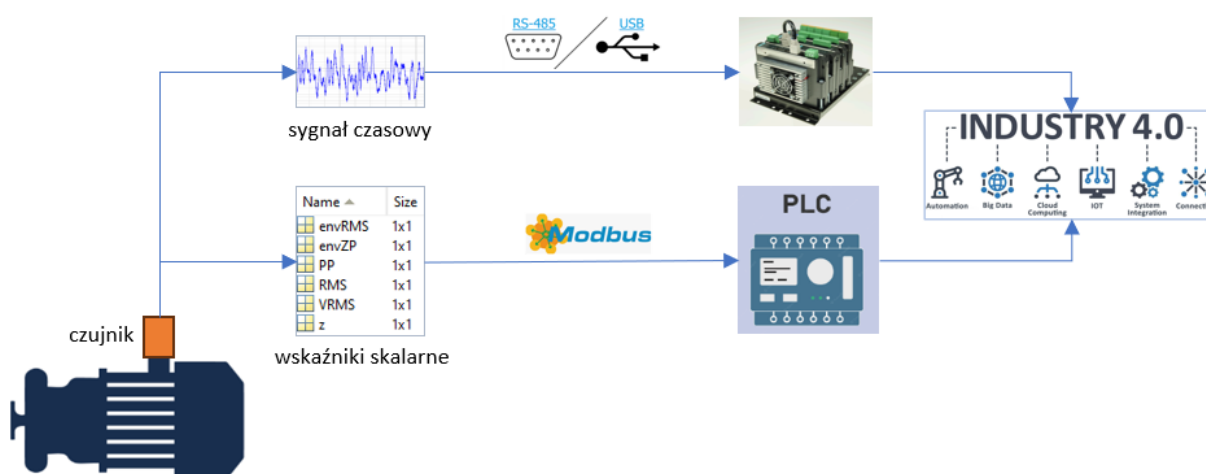
Inną cechą cyfrowych czujników drgań badanych w pracy doktorskiej jest możliwość bezpośredniego korzystania z bibliotek przetwarzania sygnałów dostępnych w trybie *open-source* dla procesorów o programowalnych modelach architektury ARM (z ang. *Advanced RISC Machine*). W prowadzonych badaniach wykorzystywano platformę z 32-bitowym procesorem ARM Cortex-M, dla którego korzystano z zestawu bibliotek podstawowych funkcji przetwarzania sygnałów *CMSIS DSP Software Library*, wersja 1.10.0. Wykorzystany zestaw CMSIS (z ang. *Cortex Microcontroller Software Interface Standard*) to opracowany przez ARM, zunifikowany programowy interfejs do programowania mikrokontrolerów Cortex-M, niezależny od ich producentów. W praktyce, wykorzystanie bibliotek CMSIS (tu modułu CMSIS.DSP) tworzy środowisko projektowe niezależne od sprzętu (HAL – z ang. *Hardware Abstractions Layer*), co ułatwia przenoszenie kodów pomiędzy mikrokontrolerami z rdzeniem Cortex-M różnych producentów. W realizowanym projekcie wykorzystywano głównie platformy rodziny STM32F4x z procesorem ARM Cortex-M4. Na etapie realizacji projektu, przyjęcie architektury Cortex-M umożliwiło praktycznie bezproblemowe przenoszenie kodu źródłowego, dzięki integracji większej ilości bloków zwiększających ich możliwości funkcjonalne poprzez zachowanie takich samych elementów obejmujących głównie:

- moduł CPU,
- wektorowy kontroler przerw z zagnieżdżeniami (NVIC – z ang. *NestedVectoredInterrupt Controller*),
- rdzeń timera systemowego (SysTick),
- moduł logiczny do wykrywania błędów w programie (debugowania).

W ramach realizacji pracy doktorskiej zaobserwowano brak różnic w uruchamianiu oprogramowania na platformach z tym samym rdzeniem Cortex-M. Metodologię wykorzystania modułu cyfrowego przetwarzania sygnałów DSP (z ang. *Digital Signal Processing*), którego dokumentacja opisana jest w [8.2] oraz [8.3] przedstawiono w Dodatku 1. W rezultacie, możliwe było opracowanie ścieżek przetwarzania sygnałów

zoptymalizowanych zarówno dla wspomnianego kryterium FAR, jak i dla kryterium ograniczenia cykli obliczeniowych.

Rysunek 8.1 przedstawia dwie główne ścieżki możliwego zastosowania w przemyśle nowego czujnika wykonanego w technologii MEMS.



Rysunek 8.1: Zastosowanie nowego cyfrowego czujnika drgań wykonanego w technologii MEMS

**Pierwsza ścieżka** obejmuje przesyłanie surowych sygnałów drganiowych do centralnej jednostki akwizycji DAQ (z ang. *Data Acquisition Unit*). Pod względem funkcjonalnym, czujnik cyfrowy spełnia tę samą funkcję, co klasyczny piezoelektryczny analogowy czujnik drgań. Dane przesyłane do systemu są wykorzystywane do permanentnej rejestracji sygnałów drgań (wg. tzw. logiki rejestracji) oraz do obliczeń różnego rodzaju analiz diagnostycznych wg. konfiguracji systemu obejmujących definicję ścieżek przetwarzania, parametry analiz i opcjonalnie rzędy analiz na podstawie modelu kinetostatycznego maszyny, co opisano szczegółowo w części teoretycznej pracy doktorskiej. Pod względem komunikacji, proponowanym interfejsem komunikacyjnym jest RS-485, który spełnia wymagania typowej instalacji przemysłowej pod względem szybkości transmisji danych, odległości transmisji oraz rozmiaru pętli. Ta ścieżka przeznaczona jest do monitorowania i diagnostyki maszyn krytycznych, wymagającej specjalistycznej analizy danych (typowo analizy widmowej).

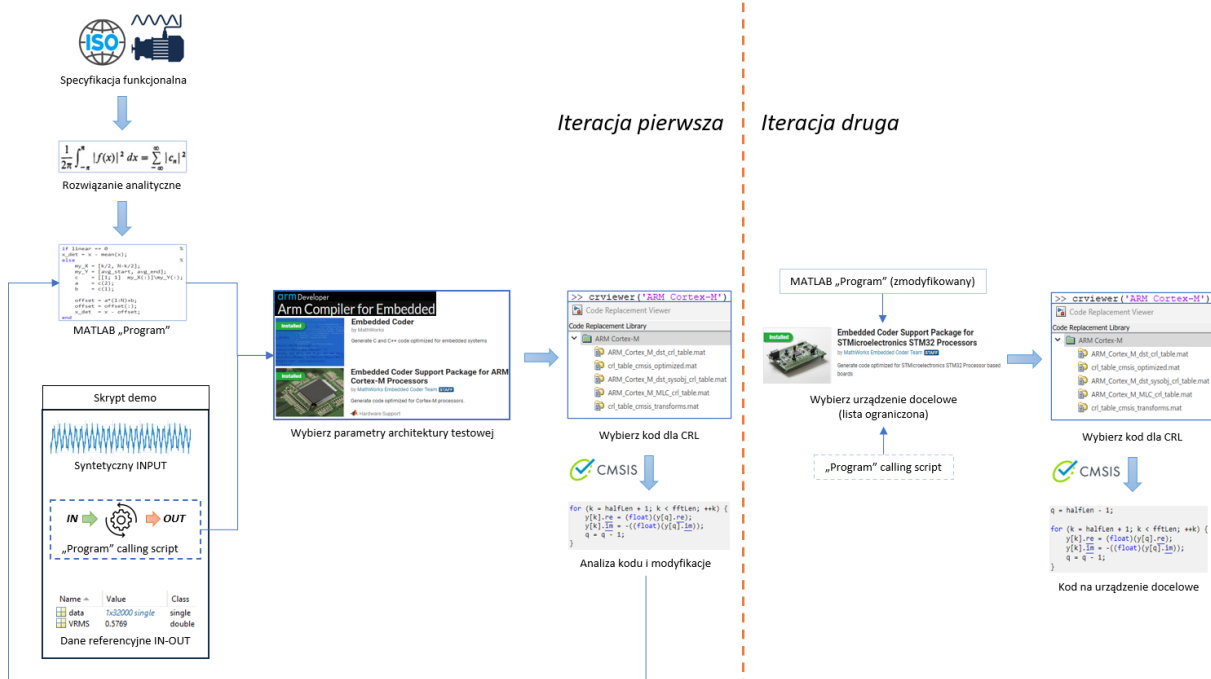
**Druga ścieżka** obejmuje obliczanie wartości zestawu wskaźników diagnostycznych w czasie rzeczywistym i przesyłanie ich do programowalnych układów logicznych, np. PLC. W

koncepcji modelu prototypu czujnika opracowano kody źródłowe do obliczenia zarówno prostych analiz pasmowo przepustowych (wartość międzyszczytowa PP sygnału przyspieszenia drgań, z ang. *Peak-to-Peak* i wartość skuteczna sygnału przyspieszenia drgań RMS, z ang. *Root-Mean-Square*) oraz szerokopasmowych analiz wymagających bardziej zaawansowanych operacji matematycznych (wartość skuteczna prędkości drgań VRMS, z ang. *Velocity Root-Mean-Square*, zgodnej z ISO 20816, dalej wartość ZP, z ang. *Zero-Peak* sygnału obwiedni oraz wartość RMS sygnału obwiedni). Pod względem komunikacji, proponowanym protokołem jest Modbus ze względu na popularność i łatwość użycia. Ta ścieżka przeznaczona jest do podstawowej oceny stanu technicznego maszyn wirnikowych pracujących w stabilnych warunkach eksploatacyjnych, ze względu na brak istniejących rozwiązań umożliwiających równoległą rejestrację sygnału znacznika fazy bez rozbudowy infrastruktury systemu pomiarowego.

## 8.2 Metodologia obliczania wartości wskaźników skalarnych

W ramach prowadzonych prac badawczych okazało się, że druga ścieżka zilustrowana na Rysunku 8.1, w której obliczenia skalarnych wskaźników diagnostycznych prowadzone są w czasie rzeczywistym na systemie wbudowanym (tj. przez zintegrowany układ czujnika), stanowi wyzwanie z uwagi na stosunkowo duży poziom zaawansowania wymaganych metod cyfrowego przetwarzania sygnałów względem mocy obliczeniowej platformy STM32 czujnika w świetle stawianych wymagań funkcjonalnych dla systemów CMS (z ang. *Condition Monitoring Systems*).

Rysunek x przedstawia wynik przeprowadzonych prac R&D w postaci schematu opracowanej metodologii rozwoju oprogramowania metod przetwarzania sygnałów dla cyfrowych czujników drgań wyposażonych w procesor ARM Cortex-M.



Rysunek 8.2: Schemat rozwoju oprogramowania na wbudowany układ nowego czujnika drgań z wykorzystaniem narzędzia MATLAB Embedded Coder i modułu CodeReplacement Library wraz z bibliotekami CMSIS dla procesorów ARM Cortex-M

Rysunek 8.2 jest rozwinięciem ścieżki „skalarniej” zilustrowanej na Rysunku 8.1 i przedstawia dwuetapowy pomysł na jej realizację. Etapy oznaczono jako „Iteracja pierwsza” oraz „Iteracja druga” ze względu na wykonanie pewnej ilości tych samych kroków ze zmienionymi parametrami. Iteracja pierwsza na Rysunku 8.2 rozpoczyna się analizą funkcjonalną systemu (tj. nowego czujnika drgań) pod kątem wymagań. Wymagania dotyczą zdefiniowania zestawu skalarnych wskaźników diagnostycznych, które będą obliczane przez czujnik. Wskaźniki diagnostyczne dobrano na podstawie konsultacji przemysłowych oraz analizy [2.1-2.10]. Dla każdego zdefiniowanego wskaźnika diagnostycznego opracowano rozwiązanie analityczne, jak pokazano na Rysunku 8.2, a następnie opracowano kod wzorcowy w środowisku MATLAB.

### 8.3 Wybrane zagadnienia algorytmów przetwarzania sygnałów

Do największych problemów naukowych na etapie opracowania zintegrowanego algorytmu obliczeniowego dla nowego czujnika zalicza się:

- opracowanie algorytmu do szybkiego usunięcia składowej niskoczęstotliwościowej,
- opracowanie algorytmu do zwiększenia wiarygodności wskaźnika VRMS w warunkach przemysłowych,

- rozwiązanie problemu względnie małego dopuszczalnego rozmiaru bufora FFT (4096 dla funkcji *arm\_rfft\_fast\_f16*) względem wymaganej ilości próbek sygnału wejściowego (32000) w procesie obliczania sygnału obwiedni.

Poniżej przedstawiono szczegółowo sposób rozwiązania kolejnych przytoczonych problemów naukowych.

### **Rozwiązanie cząstkowego problemu naukowego 1: szybka operacja detrendu**

W ramach prowadzonych badań okazało się, że czujniki drgań wykonane w technologii MEMS charakteryzują się dość dużymi wartościami składowych niskoczęstotliwościowych (głównie na skutek dryftu składowej stałej) w stanach nieustalonych, zwłaszcza po podłączeniu zasilania. Z uwagi na różny charakter składowych należało opracować sposób usunięcia wartości funkcji wielomianu pierwszego stopnia, jednakże moc obliczeniowa układu nie pozwalała na użycie klasycznych metod wyznaczania współczynników wielomianu (np. metoda najmniejszych kwadratów). Problem rozwiązano poprzez opracowanie dwuetapowego usunięcia składowej. W pierwszym kroku wykonuje się test sprawdzający, czy wartość średnia początkowej części sygnału jest istotnie różna od końcowej części sygnału, tj. sprawdza się, czy operacja jest konieczna. Jeśli tak, wartości tych lokalnych fragmentów (początkowego i końcowego) traktuje się jako wartości rzędne pary punktów u kartezjańskim układzie współrzędnych, dla których wartości odciętych przyjmuje się odpowiedni jako  $k/2$  i  $N-k/2$ , gdzie  $k$  oznacza wielkość fragmentów lokalnych, a  $N$  oznacza wielkość danego segmentu danych. W ten sposób osiągnięto szybkie usunięcie wartości wielomianu pierwszego stopnia, którego współczynniki są wyliczone z pewnym przybliżeniem. Algorytm można w łatwy sposób zapisać w środowisku MATLAB:

```
my_X = [k/2, N-k/2];
my_Y = [avg_start, avg_end];
c = [[1; 1] my_X(:)]\my_Y(:); % Calculate Parameter Vector
a = c(2); % y = a*x+b
b = c(1); % y = a*x+b
offset = a*(1:N)+b;
offset = offset(:);
x_det = x - offset;
```

Jak wskazuje przedstawiony kod, opracowany algorytm bierze zawsze dwa punkty do obliczeń zamiast wszystkich punktów, jak dokładne metody klasyczne.

### **Rozwiązanie cząstkowego problemu naukowego 2: wiarygodny VRMS**

Zgodnie z normą ISO 20816, systemy monitorowania i diagnostyki maszyn wyliczają skalarny wskaźnik diagnostyczny VRMS, czyli wartość skuteczną sygnału prędkości drgań.

Zgodnie z normą, VRMS obliczany jest z pasma 2Hz-1000Hz albo 10Hz-1000Hz, w zależności od nominalnej prędkości obrotowej maszyny. Obliczanie wskaźnika VRMS w układach numerycznych jest obarczone trudnościami z uwagi na to, że:

- sygnałem wejściowym jest sygnał przyspieszenia, a nie prędkości, co wymaga wykonanie operacji całkowania wraz z koniecznym przetwarzaniem wstępnym,
- wszelkie niefizyczne wartości drgań, związane np. ze stałą czasową elementów elektronicznych mogą prowadzić do szybkiego narastania kumulacji błędów [8.4],
- wzmocnienie wartości komponentów niskoczęstotliwościowych może powodować dość duże zmiany wartości wskaźnika VRMS przy względnie krótkich buforach czasowych sygnałów wejściowych.

Problem zwiększenia wiarygodności wskaźnika VRMS w warunkach przemysłowych rozwiązano zastosowanie szeregu operacji matematycznych, których implementację przedstawiono w Kroku 3 w Dodatku 1, jako kod tekstowy funkcji *FD\_VRMS10\_fs32000\_nFFF4096\_T1s* w środowisku MATLAB. Statyczne parametry algorytmu obejmują kolejno: długość okna *w\_len*, zgodną z maksymalną wartością dla funkcji *arm\_rfft\_fast\_f16* interfejsu CMSIS.DSP, zakres częstotliwościowy zgodny z ISO20816 (10 Hz – 1000 Hz), ilość fragmentów równą 4, współczynnik decymacji równy 4 oraz empiryczne parametry detrendu. Statyczne tablice opracowanego kodu obejmują wektor okna Tukeya [8.5] (dla minimalizacji przecieków widma) oraz wektor skalujący do przekształcenia wartości widmowych sygnału przyspieszenia do wartości widmowych sygnału prędkości wg. Omega Arythmetics [8.6]. Należy zwrócić uwagę, że w procesie decymacji zmianie ulega częstotliwość próbkowania, zgodnie z równaniem  $fs_{new} = \text{ceil}(fs/dec\_factor)$ , a więc wektor skalujący przyjmuje postać  $scaling\_int = 2 * \pi * (df:df:fs\_new/2)$ , gdzie rozdzielczość widmowa  $df$  wynosi  $fs\_new/w\_len$ . Do decymacji wykorzystano operator wektorowy środowiska MATLAB bezpośrednio jako *data(1:dec\_factor:fs)*.

Kluczowym etapem algorytmu jest wyznaczenie wielkości zakładki, która umożliwia zdefiniowanie czterech buforów po 4096 próbek (zob. *arm\_rfft\_fast\_f16* interfejsu CMSIS.DSP) dla zdecymowanego sygnału wejściowego *data\_down* (32000 próbek → 8000 próbek). Opracowane przekształcenia matematyczne można wygodnie przedstawić w środowisku MATLAB używając przyjęte zmienne jako:

$$N\_overlap = \text{floor}((w\_Len - (fs\_new - w\_Len) / (No\_frag - 1)) + 1),$$

gdzie  $N\_overlap$  oznacza ilość próbek w zakładce (tu 2795). W trakcie rozwoju oprogramowania okazało się, że iteracyjne wypełnianie pojedynczego bufora z odpowiednim wykorzystaniem operatora *switch* jako warunku pętli *for* jest rozwiązaniem lepszym, niż użycie funkcji *coder.const* proponowanej do tego typu operacji. W świetle zdefiniowanym wymagań projektowych, uzyskano następującą tablicę indeksów:

```
[ 1    4096
 1302  5397
 2603  6698
 3904  7999],
```

która ma wymiar uniwersalny w kontekście programowania cyfrowych czujników drgań wyposażonych w procesor ARM Cortex-M. Po operacji detrendu (zob. Krok 3, Dodatek 1), każdy fragment jest wymnażany przez okno Tukeya. Następnie, wyznaczane są wyskalowane, jednostronne wartości bezwzględne zespolonych współczynników transformaty Fouriera (czyli tzw. „amplitudy widmowe”). Po wyskalowaniu widmowych amplitud przyspieszenia do amplitud prędkości, wyznaczany jest wektor odpowiadający granicom pasma VRMS, zgodnie z ISO 20816. Następnie, zgodnie z Twierdzeniem Parsewała, wartość skuteczna w dziedzinie częstotliwości wyznaczana jest jako:

$$VRMS = \sqrt{\frac{1}{2} \sum Vel[i_{start}:i_{stop}]^2} \quad (4)$$

Opisany algorytm obliczania wskaźnika VRMS umożliwia obliczenie aż czterech wyników cząstkowych, każdy dla fragmentu o długości 512 ms i 4096 próbek (przy sygnale wejściowym o długości 1000 ms i 32000 próbek), co oznacza, że w procesie obliczeń widm uwzględniono więcej niż 5 obrotów wału (z uwagi na próg 10 Hz). W ostatnim kroku algorytmu zaproponowano posortowanie wyników cząstkowych i wybór wartości najmniejszej, zgodnie z obserwacją, że wskaźnik VRMS poprawnie działającego układu pomiarowego może być albo właściwy albo za duży (z uwagi na powody opisane wcześniej), lecz nigdy mniejszy niż rzeczywista wartość skuteczna prędkości drgań maszyny.

### **Rozwiązanie cząstkowego problemu naukowego 3: szybka obwiednia**

Chociaż przedstawione wcześniej rozwiązanie problemu naukowego 2 umożliwiło uzyskanie wymaganej ilości krótkich buforów poprzez operację decymacji, kolejno detrendu oraz zastosowanie zakładki, to użycie tego samego podejścia w ramach wyznaczenia sygnałów obwiedni nie jest właściwe, gdyż sygnały obwiedni z definicji muszą analizować wysokie pasma częstotliwościowe, co wyklucza zastosowanie decymacji. Pierwsze próby obliczania sygnału obwiedni obejmowały użycie filtracji w dziedzinie czasu z wykorzystaniem funkcji *filter* środowiska MATLAB skonwertowanej do języka C, jednak obliczenia były zbyt wymagające. Następnie, skorzystano z doświadczeń z interfejsem CMSIS.DSP (zob. Dodatek 1) i opracowano prototyp przetwarzania oparty o metodę *arm\_biquad\_cascade\_df2T\_f32* na podstawie źródła [8.7]. W literaturze można znaleźć sposoby na wykorzystanie środowiska MATLAB do wyznaczenia współczynników filtrów dla metod interfejsu CMSIS.DSP [8.8], jednak uznano za korzystne wypracowanie metodologii umożliwiającej automatyczne generowanie kodu dla procesorów ARM Cortex-M nap podstawie referencyjnych kodów w postaci m-plików, analogicznie do metody *fft* (zob. Dodatek 1). W tym celu zbadano dostępne możliwości i w ramach tworzenia funkcji wzorcowej wykorzystano kolejno metody środowiska MATLAB *butter*, *tf2sos*, *dsp.BiquadFilter* oraz *step* wraz z konieczną definicją zmiennych za pomocą funkcji *coder.const*, ale i tym razem czas przetwarzania przekraczał dopuszczalne wartości. Niemniej jednak, z uwagi na brak odpowiedniej dokumentacji w tym zakresie oraz na zdiagnozowane niedoskonałości w środowisku MATLAB - co wymagało dodatkowej pracy o charakterze R&D - w pracy doktorskiej, w Dodatku 4 zamieszczono opracowane instrukcje umożliwiające wygenerowanie kodu w języku C przez MATLAB *Embedded CoderToolbox* z wykorzystaniem metody *arm\_biquad\_cascade\_df2T\_f32* poprzez nakładkę *CodeReplacement Library* wraz z komentarzem. Niemniej jednak, z uwagi na długi czas operacji filtracji w dziedzinie czasu, zdecydowano się na obliczenia w dziedzinie częstotliwości z wykorzystaniem poznanej metody *arm\_rfft\_fast\_f16*, a więc na obliczenie sygnału obwiedni z wykorzystaniem transformaty Hilberta [8.9], co opisano poniżej.

Wymagania przemysłowe oraz parametry akwizycji układów elektronicznych nowego czujnika drgań wykorzystującego technologię MEMS definiują następujące wartości wstępne:

```
% T      = 1;          % dlugosc sygnalu [s]
% N      = 32000;     % rozmiar buforu wejsciowego (ilosc probek w sygnale)
% fs     = 32000;     % czestotliwosc probkowania
% f1     = 4000;      % dolna granica czestotliwosci dla demodulacji AM [Hz]
% f2     = 8000;      % gorna granica czestotliwosci dla demodulacji AM [Hz]
% w_len  = 4096;      % maksymalna dlugosc okna (FFT z interfejsu CMSIS.DSP)
```

Analiza wartości początkowych wykazała, że źródłem problemu jest blisko **aż 8-krotna różnica** pomiędzy wielkością **sygnału wejściowego**, a maksymalną wielkością **bufora wejściowego** do wybranej funkcji dla FFT, tj. *arm\_biquad\_cascade\_df2T\_f32*. W ramach rozwiązania tego problemu zaproponowano jednoczesne skrócenie czasu buforu oraz zastosowanie transformaty Hilberta w dziedzinie częstotliwości (czyli z użyciem FFT) do obliczenia sygnału obwiedni (tu jako demodulację AM). Aby umożliwić zastosowanie funkcji FFT z biblioteki CMSIS.DSP (maksymalnie 4096 próbek), na wyniku prac analitycznych zaproponowano, aby kolejno:

- podzielić cały sygnał na 7 fragmentów po 4096 próbek każdy i pominąć ostatnich  $32000 - (7 \cdot 4096) = 3342$  próbek, stanowiących 10.4 % sygnału, uzyskując rozdzielczość widmową  $32000/4096 = 7.8125$  Hz (znaczenie tych argumentów podane dalej),
- dla każdego fragmentu obliczyć FFT,
- wyznaczyć indeksy pasma widmowego odpowiadające zakresowi 4kHz-8kHz (na podstawie ISO 10816-21) jako odpowiednio  $f_{low} = f_1/df = 4000 \text{ Hz}/7.8125\text{Hz} = 512$  oraz  $f_{high} = f_2/df = 8000 \text{ Hz}/7.8125\text{Hz} = 1024$ ,
- utworzyć sygnał analityczny, generując wektor zer o długości 1024 i wypełnić jego prawą połowę zespolonymi współczynnikami FFT w paśmie 4kHz-8kHz wg. indeksów obliczonych powyżej i następnie podwoić jego wartości i wykonać operację IFFT (tu docelowo *arm\_cmsis\_fft\_do\_complex* z flagą „1U” jako trzeci argument, zgodnie z dokumentacją CMSIS.DSP)
- obliczyć wartość bezwzględną otrzymanego sygnału w celu uzyskania sygnału obwiedni.

Warto odnotować, że podobnie jak w przypadku opisu rozwiązania poprzedniego problemu naukowego, w świetle zdefiniowanym wymagań projektowych, uzyskano następującą tablicę indeksów:

```
[ 1      4096
 4097   5397
 8193   6698
12289   7999
16385  20480
20481  24576
24577  28672],
```

która również ma wymiar uniwersalny w kontekście programowania cyfrowych czujników drgań wyposażonych w procesor ARM Cortex-M, gdyż jest propozycją rozwiązania problemu

możliwości zastosowania funkcji FFT z buforem 4096 próbek dla sygnału wielkości 32000 próbek o paśmie przetwarzania ca. 4kHz-8kHz z zastosowaniem transformaty Hilberta, przy nieznaczonej utracie danych. Wymienione kroki zapisano w środowisku MATLAB w poniższej postaci:

```
No_frag = 7;           % No_frag * w_len = 7 * 4096 = 28658, 3342 probki trimmed (10.4%)
from     = 512;        % starting demodulating band index: f1/df = 4000/7.8125
till     = 1024;       % ending demodulating band index: f2/df = 8000/7.8125
envRMS_frag = zeros(1,No_frag);
envZP_frag = zeros(1,No_frag);
shifted_spectrum = complex(zeros(1,1024)); % konieczne dla ECoder

for i = 1:No_frag
    switch i
        case 1
            x = data(1:4096);
        case 2
            x = data(4097:8192);
        case 3
            x = data(8193:12288);
        case 4
            x = data(12289:16384);
        case 5
            x = data(16385:20480);
        case 6
            x = data(20481:24576);
        otherwise
            x = data(24577:28672);
    end
    % Calculate envelope signal of the current fragment
    X = fft(x);
    shifted_spectrum(1:till-from) = X(from:till-1); % "-1" due to the "fence-post" rule
    shifted_spectrum = shifted_spectrum * 2;
    analytic_signal = ifft(shifted_spectrum);
    x_env = abs(analytic_signal);
    x_env_scaled = x_env./FD_SCALING_FACTOR;
    % Store scalar values from individual fragments
    envRMS_frag(i) = rms(x_env_scaled);
    envZP_frag(i) = max(x_env_scaled);
    % clear x_env x_env_scaled analytic_signal X x
end
```

Nieoczekiwanie zauważono, że tak skonstruowany algorytm można wykorzystać do zmniejszenia wariancji wartości statystycznych opartych na sygnale obwiedni. Na podstawie testów przeprowadzonych na stanowisku badawczym AVM TestBench zaproponowano, aby wybrać wartość średnią z pierwszej połowy wartości poszczególnych wskaźników wg. sortowania rosnącego. W ten sposób można zmniejszyć wariancję wskaźników przy jednoczesnym zachowaniu odwzorowania wpływu chwilowych wartości parametrów eksploatacyjnych (badane były obciążenie i prędkość obrotowa) oraz nierówności w rozkładzie sił podczas pracy łożyska (głównie poślizg elementów tocznych wpływający na zróżnicowanie amplitud kolejnych odpowiedzi impulsowych [2.2]). Zaproponowany algorytm zapisano w środowisku MATLAB w poniższej postaci:

```
envRMS_sorted = sort(envRMS_frag);  
envZP_sorted = sort(envZP_frag);  
  
envRMS = mean(envRMS_sorted(1:4));  
envZP = mean(envZP_sorted(1:4));
```

Finalna wartość *envRMS* jest uśrednioną wartością skuteczną sygnału obwiedni przyspieszenia drgań, natomiast wartość *envZP* jest wartością szczytową ZP (z ang. *Zero-Peak*) tego samego sygnału obwiedni przyspieszenia drgań. Oba wskaźniki są powszechnie stosowane w systemach monitorowania i diagnostyki maszyn wirnikowych do wykrywania uszkodzeń łożysk tocznych oraz innych uszkodzeń generujących okresowe wzbudzenia struktury (np. niektóre uszkodzenia przekładni zębatych przy określonych parametrach eksploatacyjnych oraz przytarcia wałów).

## 9 Podsumowanie

### 9.1 Uwagi końcowe

Zakres rozprawy doktorskiej dotyczy szerokiej i dynamicznie rozwijającej się problematyki zastosowania nowoczesnej technologii MEMS do rozwoju cyfrowych czujników drgań. W szczególności w pracy doktorskiej skoncentrowano się na zagadnieniach dotyczących aspektów mechatronicznych projektu czujnika i ich wzajemnym oddziaływaniu. Praca ma charakter interdyscyplinarny, a ściśle określenie wymaganej funkcjonalności poprzez zestawienie analizy trzech głównych punktów odniesienia (Polski Komitet Normalizacyjny, reżim Industry 4.0 oraz funkcjonalność komercyjnych systemów monitorowania i diagnostyki maszyn) nadaje pracy formułę aplikacyjną.

Autor niniejszej pracy samodzielnie zaplanował i przeprowadził badania naukowe, których celem było zbadanie możliwości zastosowania niskomocowych urządzeń monitorowania opartych na nowoczesnej technologii MEMS (Micro-Electro-Mechanical Systems) do wykrywania uszkodzeń w maszynach wirnikowych. W świetle dynamicznego rozwoju technologii MEMS, która umożliwia miniaturyzację i integrację sensorów, pojawiły się nowe możliwości w zakresie monitorowania i diagnostyki maszyn. W szczególności, niskomocowe urządzenia monitorowania wykorzystujące technologię MEMS stały się obiecującym rozwiązaniem, które może przyczynić się do skutecznego wykrywania uszkodzeń w maszynach wirnikowych. Autor pracy przystąpił do realizacji badań, mając na uwadze potencjał technologii MEMS oraz potrzebę skutecznego monitorowania maszyn wirnikowych. Samodzielnie zaplanował badania naukowe, które miały na celu zbadanie wydajności i możliwości zastosowania niskomocowych urządzeń monitorowania opartych na technologii MEMS w kontekście wykrywania uszkodzeń w maszynach wirnikowych.

Wykorzystano tablicę morfologiczną jako narzędzie analityczne do systematycznego zbierania, porządkowania i analizy informacji dotyczących różnych aspektów mechatronicznych zaprojektowanych układów. Tablica morfologiczna pozwoliła na przedstawienie kompleksowej listy cech i parametrów, które są istotne w kontekście systemów monitorowania i diagnostyki maszyn (CMS). Literatura naukowa dostarczyła autorowi informacji o zasadach działania akcelerometrów MEMS, ich charakterystykach, wydajności i możliwościach pomiarowych. Ponadto, literatura techniczna dostarczyła praktycznych przykładów zastosowania układów niskomocowych w systemach diagnostyki maszyn, co stanowiło punkt odniesienia dla badań autora. Następujące po analizie literatury

badania własne przeprowadzone przez autora opierały się na wykorzystaniu układów developerskich dla wybranych platform wbudowanych oraz inżynierskich próbek akcelerometrów MEMS. Układy developerskie stanowiły praktyczne narzędzia do implementacji i testowania różnych funkcjonalności oraz wydajności układów obliczeniowych i pamięci. Autor przeprowadził testy, które uwzględniały zarówno wydajność układów obliczeniowych, jak i zużycie energii elektrycznej, co pozwoliło ocenić potencjał niskomocowych układów w kontekście systemów diagnostyki. Badania autora obejmowały również badanie właściwości protokołów komunikacyjnych pod kątem zużycia energii oraz możliwości transferu danych. Autor analizował różne protokoły komunikacyjne, oceniając ich wydajność energetyczną oraz przepustowość danych. Badania te miały na celu zidentyfikowanie optymalnych rozwiązań komunikacyjnych, które mogłyby być wykorzystane w budowie nowoczesnych systemów diagnostyki. Wszystkie te badania, zarówno oparte na analizie literatury, jak i przeprowadzone własne eksperymenty, miały na celu sprawdzenie możliwości układów niskomocowych w budowie nowoczesnych systemów diagnostyki. Autor skupiał się na identyfikacji potencjalnych korzyści, takich jak oszczędność energii, kompaktowość, wydajność i niezawodność, jakie mogą wynikać z zastosowania tych układów w systemach diagnostyki maszyn. Badania autora stanowią cenny wkład w rozwój tej dziedziny i mogą przyczynić się do dalszych prac nad budową nowoczesnych systemów diagnostyki, które będą wykorzystywały układy niskomocowe.

Autor wykorzystał analizę modalną oraz metodę elementów skończonych do zaprojektowania układu pomiarowego składającego się z akcelerometru MEMS oraz wbudowanej jednostki akwizycji danych z kilku powodów. Po pierwsze, analiza modalna umożliwia identyfikację charakterystycznych częstotliwości drgań własnych badanego obiektu. Dzięki temu można określić zakres częstotliwości, w którym układ pomiarowy powinien być odpowiednio czuły. Metoda elementów skończonych pozwala na modelowanie i symulację zachowania mechanicznego układu, co umożliwia przewidywanie jego reakcji na różne warunki obciążenia. Przeprowadzenie badań laboratoryjnych wykorzystujących czujnik referencyjny (piezoelektryczny) miało na celu potwierdzenie poprawności wyników uzyskanych w analizie metodą elementów skończonych. Porównanie wyników uzyskanych z obu rodzajów czujników pozwoliło na weryfikację dokładności i wiarygodności symulacji MES oraz potwierdzenie zgodności z rzeczywistymi danymi pomiarowymi. Badania te stanowiły istotną część procesu weryfikacji modelu. Na podstawie przeprowadzonych badań Autor zaproponował istotne zmiany w projekcie prototypów mające na celu poprawę parametrów mechanicznych układu pomiarowego. Na podstawie wyników analizy modalnej

oraz badań laboratoryjnych, autor zidentyfikował obszary, w których istniała potrzeba optymalizacji. Następnie Autor zaplanował i przeprowadził eksperymenty na specjalnie przygotowanym stanowisku badawczym w celu pokazania możliwości nowych, poprawionych prototypów czujników w badaniu popularnych uszkodzeń maszyn wirnikowych. Eksperymenty miały na celu sprawdzenie skuteczności układu pomiarowego w wykrywaniu niewyrównoważenia oraz uszkodzenia lokalnego bieżni zewnętrznej łożyska tocznego. Przebieg eksperymentów obejmował symulację różnych warunków obciążenia i monitorowanie odpowiedzi układu pomiarowego. Wyniki tych eksperymentów pozwoliły na ocenę wydajności i skuteczności nowych prototypów czujników oraz potwierdziły ich zdolność do wykrywania i diagnozowania uszkodzeń maszyn wirnikowych.

Podsumowując, wszystkie powyższe działania były istotne dla autora, ponieważ umożliwiły przeprowadzenie kompleksowej analizy i oceny możliwości zastosowania niskomocowych urządzeń monitorowania opartych na nowoczesnej technologii MEMS w wykrywaniu uszkodzeń maszyn wirnikowych. W ramach przeprowadzonych badań, autor skupił się na ocenie działania i skuteczności niskomocowych urządzeń monitorowania, analizując różne parametry i wskaźniki diagnostyczne. Przeprowadzono szereg eksperymentów, które obejmowały symulację uszkodzeń oraz analizę danych pomiarowych. Wyniki przeprowadzonych badań naukowych stanowią istotny wkład w dziedzinie monitorowania maszyn wirnikowych. Autor dokonał analizy zebranych danych, wyciągając wnioski dotyczące skuteczności i możliwości zastosowania niskomocowych urządzeń monitorowania opartych na technologii MEMS w wykrywaniu uszkodzeń. Kluczowe elementy rozwiązań opracowanych w ramach realizacji prac badawczych zostały zawarte w opisie wynalazku autora, patent nr PL 238650 B1 „Sposób i urządzenie pomiarowo-transmisyjne do akwizycji danych pomiarowych układów mechatronicznych”.

## 9.2 Wkład kandydata

Prace własne kandydata w ramach przygotowania rozprawy doktorskiej kolejno obejmują:

- a. Teoria diagnostyki technicznej i obsługa systemów CMS
  - przegląd rozwiązań systemów istniejących na rynku,
  - analiza zadań systemów CMS w oparciu o literaturę przedmiotu,
  - zapoznanie się z architekturą i konfiguracją wielokanałowych, rozproszonych, przemysłowych systemów monitorowania i diagnostyki maszyn wirnikowych na przykładzie systemu AVM4000,

- zapoznanie się z klasyfikacją skalarnych wskaźników diagnostycznych,
- zapoznanie się ze ścieżkami obliczeniowymi wskaźników diagnostycznych w dziedzinie czasu i dziedzinie częstotliwości.

#### b. Projektowanie mechatroniczne

- analiza rozwiązań chronionych dla czujników drgań oraz rozwiązań przemysłowych dostępnych na rynku,
- analiza zastrzeżeń patentowych,
- opracowanie koncepcji modelu komunikacji bezprzewodowej dla czujnika drgań wykorzystującego technologię MEMS
- wykonanie tablic morfologicznych dla koncepcji modelu czujnika,
- wykonanie modeli MES dla kolejnych wersji projektu czujnika,
- wykonanie testów analizy modalnej w warunkach laboratoryjnych,
- przyswojenie obsługi stanowiska do badań AVM Test Bench i opracowanie nadzorowanego wprowadzenia dwóch stanów uszkodzenia (niewyrównoważenie oraz lokalne uszkodzenie bieżni REB),
- opracowanie planu pomiarów,
- wykonanie pomiarów testowych na maszynie wirnikowej,
- opracowanie wyników pomiarów badanego czujnika względem czujnika referencyjnego.

#### c. Systemy wbudowane

- analiza protokołów i interfejsów komunikacji
- przegląd platform wbudowanych dostępnych na rynku
- analiza możliwości zastosowania poszczególnych protokołów i interfejsów komunikacyjnych dla sensora, jednostki akwizycji i przetwarzania danych, miejsca zapisu danych oraz modułu komunikacji i opracowanie wniosków z analizy
- zdobycie wiedzy i umiejętności w zakresie projektowania systemów wbudowanych oraz programowania systemów wbudowanych w modelu Host-Target

#### d. Metody cyfrowego przetwarzania sygnałów

- opracowanie metody szybkiego detrendu do zastosowań w systemach wbudowanych
- opracowanie metody na zwiększenie wiarygodności wskaźników szerokopasmowych, szczególnie VRMS ISO 20816

- opracowanie nowej metody obliczania obwiedni sygnału drgań dla względnie trudnych uwarunkowań systemu wbudowanego
- opracowanie metodologii na zautomatyzowane generowanie kodu źródłowego dla wybranych platform STM32 wyposażonych w procesor Cortex-M z użyciem narzędzia MATLAB Embedded Coder oraz nakładki CRL (z ang. Code Replacement Library) umożliwiającej wykorzystanie interfejsu programistycznego CMSIS.DSP (z ang. Cortex Microcontroller Software Interface Standard Digital Signal Processing).

Dodatkowe prace własne wykonane przez kandydata obejmowały przygotowanie wniosku patentowego, publikacji naukowych oraz wystąpień konferencyjnych. W ramach wykonania pracy doktorskiej, kandydat zdobył również dość szeroką wiedzę z zakresu projektowania systemów IT, a niebagatelnym elementem wkładu własnego jest długoletnie doświadczenie w programowaniu. Kandydat wykazywał ogromne zainteresowanie nowymi technologiami z zakresu optymalizacji kodu źródłowego pod różne platformy, w tym platformy wbudowane, a także procesami rozpraszania komponentów oprogramowania w celu zapewnienia wysokiej dostępności. Ostateczny wynik pracy doktorskiej kandydata wskazuje, że posiada on umiejętność samodzielnego przeprowadzenia badań naukowych, od postawienia tezy, poprzez opracowanie metodologii, wybór narzędzi, wybór kryteriów i przeprowadzenie weryfikacji symulacyjnej oraz na obiekcie rzeczywistym, jak również wyciągnięcie wniosków.

### 9.3 Dalsze prace

W ramach dalszych prac badawczych, autor planuje skoncentrować się na dwóch istotnych obszarach, które poszerzą zastosowanie monitorowania i diagnostyki maszyn.

Pierwszym z tych obszarów jest opracowanie algorytmów automatycznej synchronizacji danych w systemach rozproszonych. Współczesne systemy monitorowania i diagnostyki często składają się z wielu czujników rozmieszczonych na różnych elementach maszyn lub urządzeń. Każdy z tych czujników gromadzi dane, które są niezbędne do analizy i diagnostyki. Jednakże, ze względu na różnice w czasie i opóźnienia komunikacyjne, dane te mogą być niesynchronizowane, co utrudnia skuteczną analizę i wnioskowanie diagnostyczne. Dlatego ważne jest opracowanie zaawansowanych algorytmów, które pozwolą na automatyczną synchronizację danych z różnych źródeł w systemach rozproszonych. Takie algorytmy umożliwią dokładne porównanie i analizę danych, co z kolei przyczyni się do poprawy precyzji i skuteczności procesu diagnostycznego.

Drugim obszarem, na którym autor skupi się w dalszych badaniach, jest opracowanie metod automatycznego wnioskowania diagnostycznego z dużych zbiorów danych. Wraz z rosnącą ilością danych generowanych przez systemy monitorowania, staje się coraz trudniejsze i czasochłonne manualne analizowanie i wnioskowanie na podstawie tych danych. W związku z tym istnieje potrzeba opracowania zaawansowanych metod i technik, które umożliwią automatyczne wnioskowanie diagnostyczne z dużych zbiorów danych. Autor planuje wykorzystać techniki z zakresu uczenia maszynowego, sztucznej inteligencji oraz analizy danych do opracowania algorytmów i modeli, które automatycznie będą wnioskować i generować diagnozy na podstawie zebranych danych. Wykorzystanie takich metod pozwoli na efektywną analizę i interpretację danych diagnostycznych, identyfikację potencjalnych uszkodzeń oraz predykcję przyszłych awarii.

Wszystkie wskazane dalsze prace badawcze mają na celu rozszerzenie możliwości i skuteczności systemów monitorowania i diagnostyki maszyn. Opracowanie algorytmów automatycznej synchronizacji danych oraz metod automatycznego wnioskowania diagnostycznego z dużych zbiorów danych przyczyni się do usprawnienia procesu diagnozowania uszkodzeń maszyn, umożliwiając szybsze i bardziej precyzyjne identyfikowanie problemów oraz podejmowanie odpowiednich działań naprawczych.

## 10 Bibliografia

Bibliografię podzielono na rozdziały w celu ułatwienia wprowadzania bieżących aktualizacji, z uwagi na rozwojowy charakter niektórych elementów pracy, zwłaszcza tam, gdzie kandydat opracowywał sposoby rozwoju oprogramowania, które dopiero pojawiają się jako hasła na stronie mathworks.com (jako wersję obecną środowiska MATLAB przyjęto R2023a).

### Rozdział 1

- [1.1] A. Osan, K. Somers, Optimizing production in the age of the machine, McKinsey 2017, (<https://www.mckinsey.com/business-functions/operations/our-insights/optimizing-production-in-the-age-of-the-machine>, dostęp 20.05.2023)
- [1.2] NAM MANUFACTURERS' OUTLOOK SURVEY FOURTH QUARTER 2016, December 7, 2016, National Association of Manufacturers 2016
- [1.3] US Department of Defense. Report AD-A066-579 "Reliability-Centered Maintenance" Nowlan, F.S., Heap, H.F. December 1978
- [1.4] Waddington, C.H., Operational research against the U-boat in World War 2, London 1973
- [1.5] US Bureau of Labor Statistics (<https://www.bls.gov/> dostęp 20.05.2023)

### Rozdział 2<sup>1</sup>

- [2.1] A. Jablonski, Condition Monitoring Algorithms in MATLAB, Springer, 2021
- [2.2] R.B. Randall, Vibration-based condition monitoring: industrial, aerospace and automotive applications, John Wiley & Sons, 2011<sup>2</sup>
- [2.3] R.B. Randal, J. Antoni Rolling element bearing diagnostics: a tutorial, Mechanical Systems and Signal Processing, 25(2), 2011
- [2.4] Howard I, A review of rolling element bearing vibration detection, diagnosis and prognosis. Department of Defense, DSTO Aeronautical and Maritime Research Laboratory, Australia, DSTO-RR-0013, 1994 (dostęp <https://apps.dtic.mil/dtic/tr/fulltext/u2/a291123.pdf>, 01.07.2023 r.)
- [2.5] W. Heylen, S. Lammens, P. Sas, Modal Analysis Theory and Testing, Katholieke Universiteit Leuven, Faculty of Engineering, Department of Mechanical

---

<sup>1</sup> Referencje [2.1-2.10] zostały użyte również w rozdziale 8.2. Dodatkowo, referencje [2.1-2.2] zostały użyte wielokrotnie w innych rozdziałach pracy.

- Engineering, Division of Production Engineering, Machine Design and Automation, 1998
- [2.6] T. Zielinski, Cyfrowe przetwarzanie sygnałów. Od teorii do zastosowań), WKŁ, 2013
- [2.7] A. Muszynska, Vibrational diagnostics of rotating machinery malfunctions. *Int J Rotating Mach* 1(3-4), 1995 (dostęp <https://downloads.hindawi.com/journals/ijrm/1995/102137.pdf>, 01.07.2023 r.)
- [2.8] C. Cempel, Podstawy wibroakustycznej diagnostyki maszyn, WNT, 1982
- [2.9] Z. Engel, J. Plechowicz, L. Stryczniewicz, Podstawy wibroakustyki przemysłowej, Wydawnictwo AGH, Kraków 2003
- [2.10] W. Moczulski, Diagnostyka Techniczna. Metody pozyskiwania wiedzy, Wydawnictwo Politechniki Śląskiej, Gliwice 2002
- [2.11] W. Batko, Z. Dąbrowski, Z. Engel, J. Kiciński, S. Weyna, Nowoczesne metody badania procesów wibroakustycznych, Wydawnictwo ITE, 200A
- [2.12] W. Heylen, S. Lammens, P. Sas, Modal Analysis Theory and Testing, Katholieke Universiteit Leuven, Faculty of Engineering, Department of Mechanical Engineering, Division of Production Engineering, Machine Design and Automation, 1998
- [2.13] T. Zielinski, Od teorii do cyfrowego przetwarzania sygnałów, Wydział EaiE AGH, Krakow 2002, ISBN 83-88309-55-2
- [2.14] K. Mendrok et al., Application of modal filtration for damage detection of rotating shaft, *Key Engineering Materials*, Vol. 413-414, 2009
- [2.15] N. Sawalhi and Randall, Vibration response of spalled rolling element bearings: observations, simulations and signal processing techniques to track the spall size, *Mechanical Systems and Processing*, 25 (3) pp. 846-870, 2011
- [2.16] T.R. Pattabiraman, K. Srinivasan, and K. Malarmohan, Assessment of sideband energy ratio technique in detection of wind turbine gear defects, *Case Studies in Mechanical Systems and Signal Processing*, Volume 2, pp. 1-11, 2015
- [2.17] J.P. Hofmeister, Prognostic health monitoring overview and theory of operation (White Paper), Ridgetop Group, Inc., 2019 (available at: [www.researchgate.net](http://www.researchgate.net))
- [2.18] S. Sheng, M. McDade, and R. Errichello, Wind Turbine Gearbox Failure Modes - A Brief, in: ASME/STLE International Joint Tribology Conference, Los Angeles, California, 2011 (available at: <http://www.nrel.gov/docs/fy12osti/53084.pdf>)

- [2.19] M. Kans, D. Galar, The Impact of Maintenance 4.0 and Big Data Analytics within Strategic Asset Management, Proceedings of MPMM 2016, 6th International Conference on Maintenance Performance Measurement and Management, 28 November 2016, Lulea, Sweden
- [2.20] N. F. O. Evbuomwan, Generation of design solutions using morphological analysis, International Conference on Engineering Design, ICED 97 Tampere, August 19-21, 1997
- [2.21] R. B. Randall, Vibration-based Condition Monitoring: Industrial, Aerospace and Automotive Applications, John Wiley & Sons, 2011

#### **Rozdział 4**

- [4.1] T. Ritchey, General morphological analysis: A general method for non-quantified modelling. Adapted from the paper Morphologie and Policy Analysis presented at the 16th EURO Conference on Operational Analysis, Brussels, 1998. Swedish Morphological Society, 1-10. [dostęp: <http://swemorph.com/pdf/gma.pdf>]
- [4.2] T. Ritchey, General morphological analysis as a basic scientific modelling method. Technological Forecasting & Social Change, 126(1), 81-91, 2017.
- [4.3] N. Cross, Engineering Design Methods: Strategies for Product Design (3rd ed.). Chichester, UK: John Wiley & Sons Ltd, 2000.
- [4.4] E.B. Magrab, S.K. Gupta, F.P. McCluskey and P.A. Sandborn, Integrated Product and Process Design and Development: The Product Realization Process (2nd ed.). Boca Raton, FL, USA: CRC Press, 2010.
- [4.5] G. Pahl, W. Beitz, In K. Wallace (Ed) Engineering Design: A Systematic Approach. (K. Wallace, L. Blessing and F. Bauert, Trans. from German). Great Britain: SpringerVerlag London Ltd, 1996.
- [4.6] N.F.M. Roozenburg, J. Eekels, Product Design: Fundamentals and Methods. Chichester: John Wiley & Sons Ltd, 1995.
- [4.7] I. Wright, Design Methods in Engineering and Product Design. London: McGraw-Hill Publishing Company, 1998.
- [4.8] J.L. Richardson, J.D. Summers, G.M. Mocko, Function representations in morphological charts: An experimental study on variety and novelty of means generated. In: (M.K. Thompson, ed.) Proceedings of the 21st CIRP Design Conference, Korea 2011: Interdisciplinary Design, 76-84. KAIST, 2011.

- [4.9] N. Guangshan, L. Cong, Z. Jianwei, L. Xuetao, L. Xiangdong, (2021). Research progress of time-interleaved analog-to-digital converters. *Integration*. 81. 10.1016/j.vlsi.2021.08.007.
- [4.10] B. LaMeres, *Analog-to-Digital Converters*. 10.1007/978-3-031-20888-1\_15, 2023.
- [4.11] S. Bashir, N. Hakim, A. Rather, (2023). Analog-to-digital converters: a review of existing architectures and a new proposal for high resolution sensors. 10.1108/SR-09-2022-0369.
- [4.12] M. Zibayiwa, (2021). A Review on The Inter-Processor Communication: I2C, UART, and SPI interfacing techniques.
- [4.13] B. LaMeres, (2023). *Serial Communication in C*. 10.1007/978-3-031-20888-1\_14.
- [4.14] J. Chen, S. Huang, (2023). Analysis and Comparison of UART, SPI and I2C. 272-276. 10.1109/EEBDA56825.2023.10090677.
- [4.15] K. Murali, H. Priyadarshini, (2021). Improved design debugging architecture using low power serial communication protocols for signal processing applications. *International Journal of Speech Technology*. 24. 10.1007/s10772-020-09784-x.
- [4.16] R. Toulson, T. Wilmshurst, (2017). *Starting with Serial Communication*. 10.1016/B978-0-08-100880-5.00007-4.
- [4.17] S. Barrett, D. Pack, (2019). *Communication Systems*. 10.1007/978-3-031-79899-3\_10.
- [4.18] A. Maier, A. Sharp, Y. Vagapov, (2017). Comparative Analysis and Practical Implementation of the ESP32 Microcontroller Module for the Internet of Things. 10.1109/ITECHA.2017.8101926.
- [4.19] M. Babiuch, P. Foltynek, P. Smutný, (2019). Using the ESP32 Microcontroller for Data Processing. 1-6. 10.1109/CarpathianCC.2019.8765944.
- [4.20] N. Cameron, (2023). *ESP32 Microcontroller*. 10.1007/978-1-4842-9376-8\_1.
- [4.21] A. Perdomo-Campos, I. Vega-Gonzalez, J. Ramírez-Beltrán, (2023). ESP32 Based Low-Power and Low-Cost Wireless Sensor Network. 10.1007/978-3-031-26361-3\_24.
- [4.22] B. Deng, Z. Bo, Y. Jia, Z. Gao and Z. Liu, "Research on STM32 Development Board Based on ARM Cortex-M3," 2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT, Weihai, China, 2020, pp. 266-272, doi: 10.1109/ICCASIT50869.2020.9368860.
- [4.23] E. Alkim, P. Jakubeit, P. Schwabe, (2016). A new hope on ARM Cortex-M. 10.13140/RG.2.1.1377.5608.

- [4.24] J. Ďud'ák, G. Gaspar, S. Sedivy, R. Budjac, (2022). Utilization of RTOS Solutions in IoT Modules Based on RISC Microcontrollers. 10.1007/978-3-031-09073-8\_8.
- [4.25] J. Hu, S. Zeng, Z. Zhang, (2012). The Design of Wireless Data Acquisition System Based on STM32 and Virtual Instrument. 1-4. 10.1109/WiCOM.2012.6478710.
- [4.26] A. Nayyar, V. Puri, (2015). Raspberry Pi-A Small, Powerful, Cost Effective and Efficient Form Factor Computer: A Review. International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE). 5. 720-737.
- [4.27] H. Ghael, (2020). A Review Paper on Raspberry Pi and its Applications. 10.35629/5252-0212225227.
- [4.28] A. Kurniawan, (2021). Introduction to NVIDIA Jetson Nano. 10.1007/978-1-4842-6452-2\_1.
- [4.29] N. Nenadic, C. Valant, S. McConky, M. Thurston, P. Horney, A. Jones, (2022). Evaluation of NVIDIA Jetson System for Vibration HUMS. Annual Conference of the PHM Society. 14. 10.36001/phmconf.2022.v14i1.3232.
- [4.30] H. Halawa, H. Abdelhafez, A. Boktor, M. Ripeanu, (2017). NVIDIA Jetson Platform Characterization. 92-105. 10.1007/978-3-319-64203-1\_7.

## **Rozdział 5**

- [5.1] P. Żur, A. Kołodziej, A. Baier, "Finite Elements Analysis of PLA 3D-printed Elements and Shape Optimization.", European Journal of Engineering Science and Technology, 2019,1 (2) p. 59-64
- [5.2] S. Zargari, S. Falaki and H. Veladi, "Design and finite element analysis of a MEMS based capacitive pressure sensor using CNT/PDMS nanocomposite electrodes," 24th Iranian Conference on Electrical Engineering (ICEE), Shiraz, 2016, pp. 1619-1623.doi:10.1109/IranianCEE.2016.7585780
- [5.3] R. Isermann. Fault–diagnosis Applications: Model–based conditio Monitoring: Actuators, Drives, Machinery, Plants, Sensors, and Fault–tolerant Systems, Berlin Heidelberg: Germany: Springer, 2011.DOI: 10.1007/978–3–642–12767–0.
- [5.4] M. A. Siddhpura, A. M. Siddhpura, S.K. Bhave, "Vibration as a parameter for monitoring the health of precision machine tools",ICDM-08, Coimbatore, India, February 01-02, 2008.
- [5.5] A. Corigliano, R. Ardito, C. Comi, A. Frangi, A. Ghisi, S. Mariani, "Mechanics of Microsystems", Wiley, 2018,doi:10.1002/9781119053828.ch1.

- [5.6] J. Korvink, O. Paul, MEMS: A Practical Guide of Design Analysis and Applications, Berlin Heidelberg:Springer, 2006..
- [5.7] A. R. Jha, "MEMS and Nanotechnology-Based Sensors and Devices for Communications, Medical and Aerospace Applications", CRC Press, Boca Raton, 2007.
- [5.8] Y. J. Chen, "Advantages of MEMS and its Distinct New Applications", Advanced Materials Research, vol. 813, pp. 205-209,2013.

## Rozdział 6

- [6.1] R. Zhang, P. Wang, X. He, Wireless data acquisition system based on Internet-of-things, Application CN201520073705U·2015-01-30, Publication CN204461486U·2015-07-08, [dostęp:<https://worldwide.espacenet.com/patent/search/family/053668357/publication/CN204461486U?q=CN204461486U>, 12.06.2017]
- [6.2] S. Kooper, K. Elder, Wireless data acquisition system and method using self-initializing wireless modules, Application US65112509A·2009-12-31, Publication US8614928B2·2013-12-24, [dostęp:<https://worldwide.espacenet.com/patent/search/family/044187428/publication/US8614928B2?q=US%208614928%20B2>, 12.06.2017]
- [6.3] J. Xia, Y. Wang, Y. Ruan, Novel engine data acquisition and monitoring system, Application CN201510318057A·2015-06-11, Publication CN104931264A·2015-09-23, [dostęp:<https://worldwide.espacenet.com/patent/search/family/054118537/publication/CN104931264A?q=CN104931264A>, 12.06.2017]
- [6.4] X. Zhang, Y. Tian, Y. Wan, P. Wang, Ultra-low power consumption industry wireless data collection system, Application CN201520146595U·2015-03-16, Publication CN204516019U·2015-07-29, [dostęp:<https://worldwide.espacenet.com/patent/search/family/053713968/publication/CN204516019U?q=CN204516019U>, 12.06.2017]
- [6.5] PCB Piezotronics,ECHO® WIRELESS VIBRATION SENSOR, Model 670A01, [dostęp: <http://www.pcb.com/products.aspx?m=670A01>, 12.06.2017]
- [6.6] SKF, Wireless Machine Condition Sensor, Model CMWA 8800, [dostęp: <http://www.skf.com/group/products/condition-monitoring/surveillance-systems/on->

line-systems/wireless-systems/wireless-machine-condition-sensor/index.html,  
12.06.2017]

[6.7] Lord, Wireless Accelerometer Node, Model G-Link2™-LXRS,[dostęp:  
[http://microstrain.com/sites/default/files/g-link2\\_datasheet\\_8400-0072.pdf](http://microstrain.com/sites/default/files/g-link2_datasheet_8400-0072.pdf),  
12.06.2017]

[6.8] PCB Piezotronics, Triaxial Accelerometer, Model 356A16, [dostęp:  
<http://www.pcb.com/Products.aspx?m=356A16>, 12.06.2017]

## Rozdział 8

[8.1] A. Jabłoński, Sposób do automatycznej detekcji uszkodzeń maszyn wirnikowych poprzez automatyczną detekcję zmian w sygnałach drgań maszyny, CTT 622-29/23 AGH

[8.2] [https://arm-software.github.io/CMSIS\\_5/latest/DSP/html/modules.html](https://arm-software.github.io/CMSIS_5/latest/DSP/html/modules.html) (dostęp 01.06.2023 r.)

[8.3] ST life.augmented Application note AN4841 Rev 2, Digital signal processing for STM32 microcontrollers using CMSIS, February 2018, (dostęp na stronie: [https://www.st.com/resource/en/application\\_note/an4841-digital-signal-processing-for-stm32-microcontrollers-using-cmsis-stmicroelectronics.pdf](https://www.st.com/resource/en/application_note/an4841-digital-signal-processing-for-stm32-microcontrollers-using-cmsis-stmicroelectronics.pdf), 08.07.2023 r.)

[8.4] H. William Press et al., Numerical Recipes in C. The Art of Scientific Computing, 2<sup>nd</sup> Edition, Cambridge University Press, 2002

[8.5] P. Bloomfield, Fourier Analysis of Time Series: An Introduction. New York: Wiley-Interscience, 2000

[8.6] C. Mercer, Acceleration, Velocity and Displacement Spectra - Omega Arithmetic, Prosig White Papers (dostęp 01.06.2023 <https://www.scribd.com/document/85038391/Omega-Arithmetic-Algorithm#>)

[8.7] <https://www.mathworks.com/help/supportpkg/armcortexm/ug/cmsis-conditions-for-dsp-system-objects-to-support-arm-cortex-m-processors.html> (dostęp 01.07.2023 r.)

[8.8] M. Gaidica, Implementing a Biquad Cascade IIR Filter on an ARM Cortex M4 using CMSIS DSP, 2020 (dostęp przez [gaidi.ca/weblog](http://gaidi.ca/weblog), 01.07.2023 r.)

[8.9] D. Ho, R.B. Randall, Optimization of Bearing Diagnostic Techniques Using Simulated and Actual Bearing Fault Signals, Mechanical Systems and Signal Processing, Volume 14, Issue 5, 2000, Pages 763-788

## Dodatek 1

[D1.1] J. Szabatin, Podstawy teorii sygnałów, WKŁ, 2000

[D1.2] <https://www.mathworks.com/matlabcentral/fileexchange/52848-matlab-support-for-mingw-w64-c-c-compiler> (dostęp 08.07.2023 r.)

## **Dodatek 5**

[D5.1] <https://www.mathworks.com/help/supportpkg/armcortexm/ug/supported-cmsis-functions.html> (dostęp 01.07.2023 r.)

[D5.2] <https://www.mathworks.com/help/supportpkg/armcortexm/ug/cmsis-conditions-for-dsp-blocks-to-support-arm-cortex-m.html>, (dostęp 01.07.2023 r.)

[D5.3] <https://www.mathworks.com/help/supportpkg/armcortexm/ug/cmsis-conditions-for-dsp-system-objects-to-support-arm-cortex-m-processors.html>, (dostęp 01.07.2023 r.)

[D5.4] <https://www.mathworks.com/help/supportpkg/armcortexm/ug/supported-simulink-blocks-and-matlab-system-objects-for-cmsis-library.html>, (dostęp 01.07.2023 r.)

[D5.5] Maxim Integrated Products APPLICATION NOTE 1762, A Beginner's Guide to Filter Topologies, 2003 (dostęp ze strony ANALOG DEVICES INC. , 01.07.2023 r.)

[D5.6] G. S. Moschytz, Analog circuit theory and filter design in the digital world: with an introduction to the morphological method for creative solutions and design, Springer, Cham, Switzerland, 2019

[D5.7] <https://www.mathworks.com/help/supportpkg/armcortexm/ug/supported-simulink-blocks-and-matlab-system-objects-for-cmsis-library.html> (dostęp 01.07.2023 r.)

[D5.8] <https://www.advsolned.com/implementing-biquad-iir-filters-with-the-asn-filter-designer-and-the-arm-cmsis-dsp-software-framework/> (dostęp 01.07.2023 r.)

## **Dodatek 6**

[D6.1] A. Jablonski, T. Barszcz, M. Bielecka, Automatic validation of vibration signals in wind farm distributed monitoring systems, Measurement, Volume 44, Issue 10, 2011

[D6.2] A. Jablonski, T. Barszcz, Validation of vibration measurements for heavy duty machinery diagnostics, Mechanical Systems and Signal Processing, Volume 38, Issue 1, 2013

## Spis ilustracji

Rysunek 1.1: Porównanie wpływu skalowalności na koszt obsługi systemu dla systemów stacjonarnych oraz przenośnych.....	22
Rysunek 1.2: Obraz z mikroskopu elektronowego (SEM) inercyjnego akcelerometru MEMS [https://www.analog.com/en/technical-articles/mems-accelerometers-for-condition-monitoring.html] .....	23
Rysunek 2.1: Zasada pomiaru przyspieszenia drgań.....	35
Rysunek 2.2: Ścieżki analizy pomiaru drgań .....	36
Rysunek 2.3: Po lewej: ścieżka identyfikacji uszkodzeń, po prawej: ścieżka detekcji uszkodzeń, dla pojedynczego pakietu danych.....	41
Rysunek 2.4: Pośrednie wykrywanie błędów poprzez identyfikację uszkodzeń .....	42
Rysunek 2.5: Typy i zadania systemów CMS .....	47
Rysunek 2.6: Graficzna reprezentacja głównych elementów architektury CMS .....	51
Rysunek 2.7: Ogólny schemat wdrożenia stacjonarnego CMS.....	53
Rysunek 2.8: Częściowe drzewo konfiguracji CMS [Dzięki uprzejmości AMC Vibro®]56	
Rysunek 2.9: Przykładowe ustawienia progów – prosty przypadek [Dzięki uprzejmości AMC Vibro®].....	57
Rysunek 2.10: Przykładowe ustawienia progów – trudny przypadek [Dzięki uprzejmości AMC Vibro®].....	58
Rysunek 3.1: Niewyważenie wału w widmie rzędów [Dzięki uprzejmości AMC Vibro®] .....	61
Rysunek 3.2: Niewyważenie wału w widmie częstotliwości [Dzięki uprzejmości AMC Vibro®].....	61
Rysunek 3.3: Przykładowa sygnatura wibracji przekładni wokół GMFx1 [Dzięki uprzejmości AMC Vibro®].....	62
Rysunek 3.4: Sygnał wysokiej częstotliwości REB z obwiednią.....	64
Rysunek 4.1: Przykłady architektury urządzenia wbudowanego, mającej kluczowy wpływ na przyszłą funkcjonalność CMS .....	70
Rysunek 4.2: Schemat wbudowanego DAQ jako autonomicznego urządzenia w lokalnym systemie monitoringu .....	85
Rysunek 4.3: Schemat akcelerometru MEMS.....	89
Rysunek 4.4: Schemat protokołu komunikacyjnego I <sup>2</sup> C .....	90
Rysunek 4.5: Schemat protokołu komunikacyjnego SPI .....	92

Rysunek 4.6: Schemat protokołu komunikacyjnego UART .....	94
Rysunek 4.7: Schemat interfejsów w module komunikacyjnym.....	96
Rysunek 4.8: Przegląd zastosowań dla wybranych akcelerometrów MEMS firmy Analog Devices Inc. [ <a href="https://www.analog.com/en/analog-dialogue/articles/choosing-the-most-suitable-mems-accelerometer-for-your-application-part-1.html">https://www.analog.com/en/analog-dialogue/articles/choosing-the-most-suitable-mems-accelerometer-for-your-application-part-1.html</a> ] .....	121
Rysunek 5.1: Rozważane położenie czujnika MEMS na płytce PCB akcelerometru.....	130
Rysunek 5.2: Charakterystyka częstotliwościowa czujnika z MEMS umieszczoną na środku płytki PCB.....	132
Rysunek 5.3: Charakterystyka częstotliwościowa czujnika z MEMS umieszczonym na spodzie płytki PCB.....	133
Rysunek 5.4: Porównanie odpowiedzi częstotliwościowej struktury w zależności od lokalizacji MEMS na płytce PCB .....	134
Rysunek 5.5: Konfiguracja eksperymentu przy użyciu wibrometru laserowego Polytec PSV400.....	136
Rysunek 5.6: Konfiguracja eksperymentu z użyciem wibrometru laserowego Polytec PSV400 – wiązka lasera skierowana na tarczę akcelerometru w kierunku pomiaru .....	137
Rysunek 5.7: Porównanie otrzymanej odpowiedzi częstotliwościowej z analizy (obliczonej dla prędkości w [m/s]) i eksperymentu z użyciem wibrometru laserowego PSV400 .....	138
Rysunek 5.8: Konfiguracja eksperymentu przy użyciu wzbudnika TIRAvib TV51144.	139
Rysunek 5.9: Surowy sygnał mierzony z czujników referencyjnych i MEMS na wzbudniku TIRAvib TV51144 .....	140
Rysunek 5.10: Odpowiedź częstotliwościowa obliczona na podstawie zmasowanego sygnału z czujnika MEMS na wytrząsarce TIRAvib TV51144.....	141
Rysunek 5.11: Proponowana uproszczona konstrukcja płytki czujnikowej z akcelerometrem MEMS .....	143
Rysunek 5.12: Pasma przenoszenia czujnika z akcelerometrem MEMS. Założone tłumienie 3% zostało uwzględnione podczas symulacji (wykres tłumienia analizy.....	144
Rysunek 7.1: Stanowisko badawcze Vibstand 2 (laboratorium AGH, WIMiR).....	161
Rysunek 7.2: Schemat kinetostatyczny urządzenia Vibstand 2.....	162
Rysunek 7.3: Tarcza ze śrubami wprowadzającymi niewyrównowazenie na wale wyjściowym stanowiska.....	163
Rysunek 7.4: Demontaż łożyska tocznego na stanowisku Vibstand 2.....	164
Rysunek 7.5: Uszkodzenie lokalne bieżni zewnętrznej łożyska tocznego .....	165

Rysunek 7.6: Schemat prototypowego układu akwizycji danych .....	166
Rysunek 7.7: Przykładowy widok aplikacji do wizualizacji drgań z prototypowego czujnika MEMS.....	167
Rysunek 7.8: Montaż prototypu i czujnika referencyjnego na obudowie łożyska. Lewa: Prototypowy czujnik bez obudowy, prawa: montaż czujników na obudowie łożyska .....	168
Rysunek 7.9: Przebiegi czasowe dla czujnika referencyjnego PCB.....	170
Rysunek 7.10: Przebiegi czasowe dla czujnika referencyjnego PCB.....	171
Rysunek 7.11: Widmo uśrednione z sygnałów z czujnika referencyjnego PCB.....	171
Rysunek 7.12: Widmo uśrednione z sygnałów z czujnika prototypowego MEMS .....	172
Rysunek 7.13: Przebiegi czasowe sygnału referencyjnego i badanego.....	173
Rysunek 7.14: Widma w pełnej rozdzielczości sygnału referencyjnego i sygnału badanego.....	174
Rysunek 7.15: Przebiegi czasowe sygnałów obwiedni dla czujnika referencyjnego i czujnika MEMS.....	175
Rysunek 7.16: Widma sygnału obwiedni dla czujnika referencyjnego oraz czujnika MEMS .....	176
Rysunek 8.1: Zastosowanie nowego cyfrowego czujnika drgań wykonanego w technologii MEMS .....	179
Rysunek 8.2: Schemat rozwoju oprogramowania na wbudowany układ nowego czujnika drgań z wykorzystaniem narzędzia MATLAB Embedded Coder i modułu CodeReplacement Library wraz z bibliotekami CMSIS dla procesorów ARM Cortex-M.....	181
Rysunek D1.1: Widok braku wyników tożsamy z opracowaną metodologią .....	210
Rysunek D1.2: Sygnał syntetyczny w środowisku MATLAB.....	211
Rysunek D1.3: Uruchomienie narzędzia MATLAB Coder .....	216
Rysunek D1.4: Przykładowy widok wyboru funkcji i konwersji precyzji danych .....	216
Rysunek D1.5: Wprowadzenie skryptu uruchamiającego funkcję właściwą.....	217
Rysunek D1.6: Wykorzystanie skryptu testowego do testowania kodu.....	217
Rysunek D1.7: Konfiguracja środowiska docelowego – widok ogólny.....	218
Rysunek D1.8: Ustawienia metod alokacji pamięci .....	219
Rysunek D1.9: Ustawienia wizualizacji dla bibliotek CMSIS .....	219
Rysunek D1.10: Ustawienia wyboru zastępczych bibliotek CMSIS.....	220
Rysunek D1.11: Ideowy wynik działania modułu Embedded Coder.....	221
Rysunek D1.12: Widok okna raportu z przejściem do zakładki „CodeReplacement”....	221
Rysunek D1.13: Analiza działania modułu „CodeReplacement” .....	223

Rysunek D1.14: Ilustracja metody CMSIS w ramach CodeReplacement Library .....	223
Rysunek D1.15: Wywołanie bibliotek CodeReplacement Library .....	224
Rysunek D1.16: Widok klucza metody arm_cmsis_fft_do_real z poziomu CRV .....	224
Rysunek D1.17: Widok referencyjny dla CMSIS DSP Software Library, v.1.10.0.....	225
Rysunek D1.18: Przykładowy wynik generacji kodu w MATLAB Embedded Coder...	226
Rysunek D1.19: Okno zakończenia procedury generacji kodu.....	226
Rysunek D1.20: Wybór platformy docelowej STM32F4.....	227
Rysunek D5.1: Widok braku wyników wsparcia dla badanego zastosowania.....	233
Rysunek D5.2: Komunikat MATLAB Embedded Coder: coder.Constant zalecany .....	234
Rysunek D5.3: Komunikat MATLAB Embedded Coder: coder.Constant niewspierany (w tym scenariuszu).....	234
Rysunek D5.4: Ilustracja działania CodeReplacement Library dla filtracji .....	235

## Spis tabel

Tabela 2.1: Typowe zadania systemu CMS .....	45
Tabela 2.2: Przykładowe elementy specyfikacji .....	55
Tabela 4.1: Przykład tabeli morfologicznej .....	69
Tabela 4.2: Tablica morfologiczna przedstawiająca wszystkie najważniejsze elementy składowe rozpatrywanego CMS.....	71
Tabela 4.3: Analiza dostępnych źródeł zasilania.....	73
Tabela 4.4: Analiza dostępnych przetworników analogowo-cyfrowych .....	76
Tabela 4.5: Analiza możliwości przechowywania danych pomiarowych.....	77
Tabela 4.6: Analiza możliwości komunikacji wewnętrznej pomiędzy elementami przetwornika.....	79
Tabela 4.7: Analiza możliwości transmisji danych pomiędzy przetwornikiem, a urządzeniem odbiorczym .....	81
Tabela 4.8: Analiza mocy obliczeniowej procesora .....	83
Tabela 4.9: Porównanie protokołów połączeń CMS stosowanych w części systemu wbudowanego.....	103
Tabela 4.10: Porównanie protokołów połączeń CMS używanych między systemem wbudowanym a bramą sieciową (lub użytkownikiem końcowym).....	104
Tabela 4.11: Parametry techniczne układu ESP32 .....	109
Tabela 4.12: Porównanie różnych modeli STM32 z różnymi rdzeniami arm cortex M. ....	111
Tabela 4.13: Porównanie modeli Raspberry Pi dostępnych na rynku .....	113
Tabela 4.14: Porównanie komputerów NVIDIA Jetson TK1 i TX1 .....	115
Tabela 4.15: Porównanie wybranych systemów wbudowanych .....	116
Tabela 4.16: Klasa akcelerometru i typowy obszar zastosowania .....	119
Tabela 5.1: Uzyskane wyniki analizy modalnej i harmonicznej dla dwóch rozważanych lokalizacji czujnika MEMS .....	131
Tabela 5.2: Specyfikacja techniczna akcelerometru MEMS wybranego do prototypu czujnika .....	135
Tabela 5.3: Własności mechaniczne zaproponowanego modelu .....	144
Tabela 5.4: Zmierzony poziom szumu sygnału .....	145
Tabela 7.1: Porównanie wartości RMS sygnału przyspieszenia szumu.....	169
Tabela 7.2: Poziomy RMS dla czujnika referencyjnego i prototypowego w osiach XYZ .....	172

Tabela D1.1: Funkcje zastąpione przez CodeReplacement Library.....	222
Tabela D1.2: Operator odejmowania zastąpiony przez CodeReplacement Library.....	222
Tabela D1.3: Działania macierzowe zastąpione przez CodeReplacement Library.....	222
Tabela D6.1: Automatyczna walidacja sygnałów drgań dla czujników cyfrowych.....	237

## Dodatek 1 – Metodologia rozwoju oprogramowania (CMSIS + CRL)

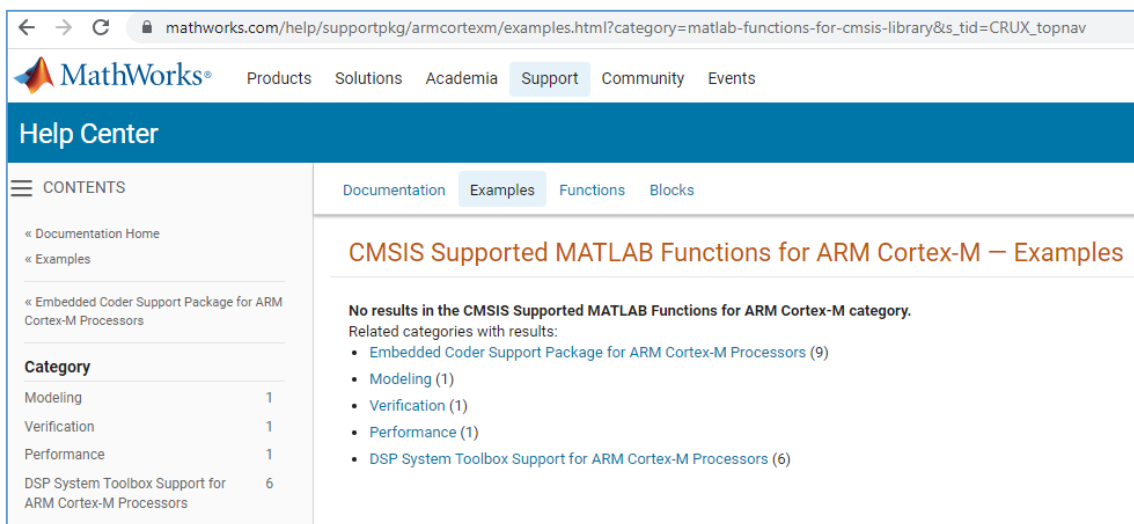
Dodatek 1 opisuje poszczególne kroki w ramach realizacji metody rozwoju oprogramowania przedstawionej schematycznie na Rysunku 8.2. W kolejnych krokach omówiono stworzenie przykładowego syntetycznego sygnału testowego, stworzenie przykładowego skryptu wywołującego (z ang. tzw. calling script), opracowanie właściwej funkcji wdrożeniowej (tu dla wskaźnika skalarnego VRMS, zgodnie z ISO 20816), opracowanie skryptu demonstracyjnego, który zapisuje wyniki referencyjne obliczone w środowisku MATLAB, utworzenie projektu w narzędziu MATLAB Embedded Coder® wraz w instrukcją wyboru właściwych parametrów dotyczących kompilacji i wygenerowania kodu źródłowego ze szczególnym uwzględnieniem wykorzystania interfejsu Code Replacement Library (CRL) umożliwiającego implementację wbudowanych funkcji dla procesorów ARM Cortex-M, analizę kodu z wykorzystaniem modułu Code Tracer w narzędziu MATLAB Coder Report Viewer oraz instrukcje dotyczące wdrażania kodu na konkretne urządzenie wbudowane. W ramach realizacji pracy wykorzystywano następujące funkcje z biblioteki CMSIS:

- |                                      |                                       |
|--------------------------------------|---------------------------------------|
| ▪ <i>arm_mean_f32</i>                | <i>obliczanie wartości średniej</i>   |
| ▪ <i>arm_rms_f32</i>                 | <i>obliczanie wartości skutecznej</i> |
| ▪ <i>arm_power_f32</i>               | <i>obliczanie potęgi</i>              |
| ▪ <i>arm_var_f32</i>                 | <i>obliczanie wariancji</i>           |
| ▪ <i>arm_biquad_cascade_df2T_f32</i> | <i>filtracja sygnału</i>              |
| ▪ <i>arm_rfft_fast_f32</i>           | <i>obliczanie współczynników FFT</i>  |

Szczególne znaczenie dla przedstawionego kodu ma funkcja *arm\_rfft\_fast\_32*. Pełny zestaw funkcjonalności środowiska MATLAB, które zostały wykorzystane w pracy doktorskiej są dostępne dopiero od wersji MATLABA R2023a (marzec 2023 r.). Jednakże, jak wskazuje strona [https://www.mathworks.com/help/supportpkg/armcortexm/matlab-functions-for-cmsis-library.html?s\\_tid=CRUX\\_lftnav](https://www.mathworks.com/help/supportpkg/armcortexm/matlab-functions-for-cmsis-library.html?s_tid=CRUX_lftnav) (dostęp 05.07.2023 r.), w zakładce „Topics” ogólne wsparcie jest dostępne, ale temat zastosowania funkcji z biblioteki CMSIS w ramach narzędzia Coder Replacement Library (CRL) w ujęciu holistycznym, które jest niezbędne w ramach opracowania kodów dla nowych czujników wykonanych w technologii MEMS jest nieopracowany:

- *CMSIS Conditions for MATLAB Functions to Support ARM Cortex-M Processors* - opracowany
- *CMSIS CRL conditions for MATLAB functions to support ARM Cortex-M* - nieopracowany

Rysunek D1 przedstawia zrzut ekranu z platformy Mathworks® z informacją o istnieniu hasła dla badanego tematu, ale z brakiem wsparcia dla badanego tematu (stan na dzień 08.07.2023 r). Brak wsparcia widoczny jest poprzez komunikat Support→Examples→„No results in the CMSIS Supported MATLAB Functions for ARM Cortex-M Category”, jak widać na Rysunku D1.1.

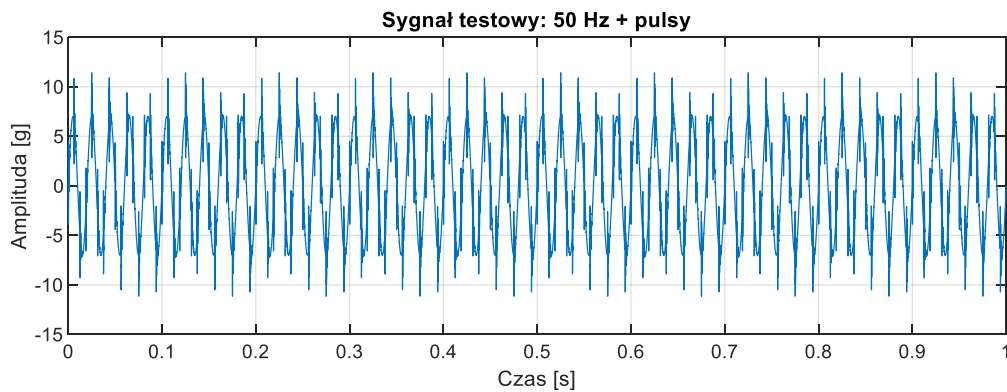


Rysunek D1.1: Widok braku wyników tożsamych z opracowaną metodologią

W związku z powyższym stwierdzeniem można uznać, że Dodatek 1 ma charakter innowacyjny. W ramach realizacji pracy doktorskiej przygotowano pełny zestaw kodów dla czujników drgań, natomiast w niniejszym dodatku przedstawiony jest kod uproszczony, przedstawiający obliczanie przykładowego wskaźnika wykorzystywanego w systemach oceny stanu technicznego, tj. wartość skuteczną sygnału prędkości drgań (z ang. VRMS), zgodnie z normą ISO 20816.

### **Krok 1: Stworzenie pliku testowego w środowisku MATLAB**

Poniższy skrypt przedstawia program generujący przykładowe dane testowe, zgodnie z parametrami akwizycji nowego czujnika. Rysunek D1.2 przedstawia ilustrację graficzną wyniku uruchomienia skryptu `generate_INPUT.m` umieszczonego poniżej.



Rysunek D1.2: Sygnał syntetyczny w środowisku MATLAB

```

% *****
% **** file name: generate_INPUT.m *****
% *****
fs = 32000;           % sampling frequency
T = 1;               % signal total time

Ac = 5;              % carrier amplitude
fc = 6000;           % carrier frequency
fp = 160;            % frequency of pulses
dr = -.25*fc;        % pulse decaying rate

dt = 1/fs;           % time resolution
N = T*fs;            % total number of samples in the signal
Tp = 1/fp;           % total time of a single pulse

t = dt:dt:T;         % time vector of the entire signal
tp = dt:dt:Tp;       % time vector of a single pulse
n = length(tp);      % length of one pulse (in No. of points)

x = exp(dr*tp)*Ac.*sin(2*pi*fc*tp); % single pulse

No_pulses = ceil(N/n)+1; % extra pulse for boundary conditions
data = repmat(x,1,No_pulses); % replicate and tile pulses
data = data(1:N); % trim part of last pulse if necessary
data = single(data);

data = data + 7*sin(2*pi*50*t); % add single sinusoidal component 50Hz
% *****
plot(t,data);
xlabel('Czas [s]');
ylabel('Amplituda [g]');
title('Sygnał testowy: 50 Hz + pulsy')
set(gca,'fontsize',8); grid on
% *****

```

Wynik działania skryptu *generate\_INPUT.m* jest wykorzystywany w ramach skryptu *script\_call.m* opisanego poniżej, w kroku 2 Dodatku 1.

## **Krok 2: Stworzenie przykładowego skryptu wywoławczego dla opracowywanej funkcji**

W ramach zautomatyzowanego generowania kodu źródłowego dla platform wbudowanych należy przygotować program, który jest wykorzystywany przez środowisko MATLAB to testowania wygenerowanego kodu źródłowego.

```
% *****
% ***** file name: script_call.m *****
% *****
generate_INPUT;
tic
[VRMS] = FD_VRMS10_fs32000_nFFF4096_T1s(data);
toc
```

W przedstawionym kodzie, nazwa funkcji *VRMS10\_fs32000\_nFFF4096\_T1s* jest wirtualna, gdyż definicja funkcji zostanie utworzona w kolejnym kroku. Podana nazwa funkcji informuje, że wartość VRMS będzie obliczana w dziedzinie częstotliwości (z ang. FD – *Frequency Domain*) standardowo, od 10 Hz dla częstotliwości próbkowania 32 kHz, wykorzystując bufor dla Transformaty Fouriera 4096 próbek oraz czasu sygnału 1 sekunda. Dodatkowe użycie metod zliczających czas, tj. *tic-toc* umożliwia finalne porównanie czasu wykonania kodu pomiędzy komputerem klasy PC i środowiskiem MATLAB a platformą wbudowaną. Takie dane pozwalają wstępnie oszacować czas realizacji innych kodów bez konieczności prototypowania, co zostało wykorzystane w pracy doktorskiej.

### **Krok 3: Opracowanie właściwej funkcji dla wdrożenia (z ang. *deployment*)**

W ramach opracowania kompletnego modelu nowego czujnika drgań wykorzystującego technologię MEMS, zgodnie z ISO 20816, program pracujący na czujniku w środowisku wbudowanym oblicza szereg analiz, w tym analizę VRMS przedstawioną w celu demonstracyjnym. Obliczenie wartości VRM przedstawiono poniżej.

```
% *****
% ***** file name: FD_VRMS10_fs32000_nFFF4096_T1s.m *****
% *****
function [VRMS] = FD_VRMS10_fs32000_nFFF4096_T1s(data) %codegen
data = data(:);
% DESCRIPTION
% VRMS obliczany jest jako najmniejsza wartosc z 4 fragemntow po 0.512[s] kazdy,
% z dedykowanym overlapem (FFT)
% -----
% ----- COMMON STATIC PARAMETERS -----
% -----
fs = 32000;          % sampling frequency
% -----
% ----- KONFIGURACJA STATYCZNA VRMS -----
% -----
w_len      = 4096;    % fixed for all FFT input; VRMS & ENV
limits     = [10 1000]; % pasmo VRMS [Hz]
No_frag    = 4;      % fixed for all fs (ilosc fragmentow z zakladka)
dec_factor = 4;      % Calculate_Decimation_Factor (static TARGET_1s=8192)
k          = 200;    % Detrend: number of points taken for calc. start & end
my_thr     = 0.1;    % Detrend: 10% difference start vs end requires lin. det.
N          = 4096;   % Detrend: N = w_len
```

```

% -----
% ----- VRMS -----
% -----
win          = tukeywin(w_len, .2);           % Generate_Tukey_Window
data_down    = data(1:dec_factor:fs);        % Downsample_Data:      <8000 x 1>
fs_new       = ceil(fs/dec_factor);          % Calculate_Fs_New:      32kHz/4=8kHz
df           = fs_new/w_len;                 % nowa sampling frequency: 1.953125 Hz
scaling_int  = 2*pi*(df:df:fs_new/2)';      % wektor skalujacy A-->V: <2048 x 1>

% NOTE: Dla zachowania statycznej alokacji pamieci, wykorzystywane jest
% obliczenie zakladki wg.:
% N_overlap  = floor(w_len-(fs_new-w_len)/(No_frag-1))+1;
% co daje zakladkę w ilości punktów: 2795

VRMS_frag = zeros(1,No_frag);
for i = 1:No_frag
    switch i
        case 1
            x = data_down(1:4096);
        case 2
            x = data_down(1302:5397);
        case 3
            x = data_down(2603:6698);
        otherwise
            x = data_down(3904:7999);
    end
    % -----
    % Detrend START
    % -----
    part_start = x(1:k);
    part_end   = x(end-k+1:end);
    avg_start  = mean(part_start);           % begining of signal
    avg_end    = mean(part_end);             % end of signal
    my_limits  = [avg_start-my_thr*avg_start, avg_start+my_thr*avg_start];
    linear = 0;                               % change to 1 if variation significant
    if avg_end > max(my_limits) || avg_end < min(my_limits)
        linear = 1;
    end
    if linear == 0                               % DC detrend
        x_det = x - mean(x);
    else                                           % linear detrend
        my_X = [k/2, N-k/2];
        my_Y = [avg_start, avg_end];
        c = [[1; 1] my_X(:)]\my_Y(:);          % Calculate Parameter Vector
        a = c(2);                               % y = a*x+b
        b = c(1);                               % y = a*x+b

        offset = a*(1:N)+b;
        offset = offset(:);
        x_det = x - offset;
    end
    % -----
    % Detrend STOP
    % -----
    x_det_win = x_det.*win;
    Y = abs(fft(x_det_win));
    Acc = Y(1:w_len/2+1)./(w_len/2);           % spectral Acc. amplitudes
    Vel = 9810*Acc(2:end)./scaling_int;        % spectral Vel. amplitudes
    i_start = round(limits(1)/df);             % static: 5
    i_stop = floor(limits(2)/df);             % static: 512
    VRMS_frag(i) = sqrt(1/2*sum(Vel(i_start:i_stop).^2)); % Parseval's T.
end

VRMS = min(VRMS_frag); % Practical consideration

```

W przedstawionym kodzie, w ramach realizacji prac badawczych B+R, wykonano następujące prace:

- opracowano wektor skalujący dla sygnału prędkości dla sygnału zdecydowanego,
- opracowano metodę podziału sygnału na fragmenty wg. klucza bufora FFT przy jednoczesnym kryterium minimalnej ilości fragmentów,
- opracowano metodę szybkiego detrendowania dla systemów wbudowanych (szczególnie istotne podczas uruchomienia czujników wykonanych w technologii MEMS),
- opracowano adaptację metody szybkiego wyznaczania współczynnika kierunkowego prostej pod kątem zautomatyzowanego generowania kodów źródłowych,
- opracowano adaptację Twierdzenia Parsewala [D1.1] do szybkiego wyznaczania wartości skutecznej sygnału ze współczynników widmowych.

Dodatkowo, na podstawie wyników empirycznych zaproponowano prosty statystyczny klasyfikator wyznaczania finalnej wartości VRMS z wektora wartości lokalnych dla całego bufora danych, jako alternatywy dla metod bazujących na obliczaniu średniej ruchomej. Jak wykazały testy na obiektach rzeczywistych, wymagana redukcja fałszywych alarmów wymagałaby zastosowania względnie długiego okna MA, co niekorzystnie wpływa na dynamikę trendów w systemie CMS.

Podczas pisania kodu w środowisku MATLAB, który ma być przekonwertowany na język C zalecane jest skorzystanie z materiałów z serii „*Best Practices for CodeGeneration*” dostępnych na stronie [mathworks.com](http://mathworks.com).

### **UWAGA**

Ważnym elementem pisania kodu referencyjnego w środowisku MATLAB jest analiza uwarunkowań dla danej platformy docelowej (z ang. *Conditions*). Poszczególne uwarunkowania/ograniczenia są dostępne na dedykowanych stronach wsparcia [mathworks.com](http://mathworks.com). W przedstawionym projekcie konieczna była analiza wartości granicznych parametrów wg. Materiałów wsparcia „*CMSIS Conditions for MATLAB Functions to Support ARM Cortex-M Processors*” (MATLAB R2023a). Podczas projektowania kodu wykorzystano wymagania dot. długości buforów rzeczywistych danych wejściowych, tj. {4096}-{16}, zgodnie z *arm\_rfft\_fast\_f32*, omówionej w dalszej części.

### **Krok 3: Opracowanie skryptu demo dla środowiska MATLAB**

Posiadając już skrypt do generowania sygnału testowego, właściwą funkcję obliczeniową oraz przykładowy skrypt wywołujący (wymagany przez MATLAB Toolbox Embedded Coder®), przed przystąpieniem do procesu generacji kodu źródłowego (tu w jęz. C), korzystnie jest przygotować skrypt demonstracyjny, który w ramach jednego programu uruchomi opracowane skrypty/funkcje oraz wyświetli/zapisze wzorcowe dane wynikowe. Przykładowy skrypt demonstracyjny przedstawiono poniżej.

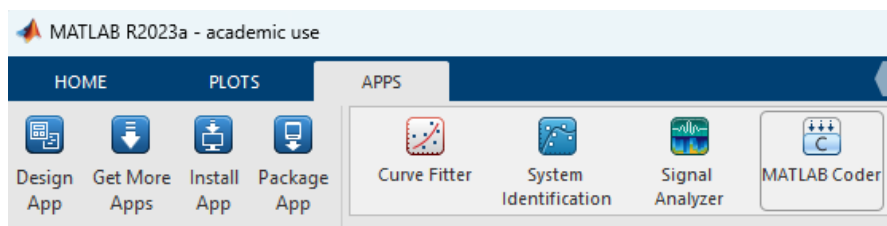
```
% *****  
% ***** file name: demo.m *****  
% *****  
clear;  
clc;  
script_call; % calls: "generate_INPUT.m"  
            % calls: "FD_VRMS10_fs32000_nFFF4096_T1s"  
% -----  
fid = fopen('INPUT_fs32000_T1s_comma_sep.txt', 'w');  
data = single(data);  
for k = 1:length(data)  
    fprintf(fid, '%f%s', data(k), ',');  
    fprintf(fid, '\n');  
end  
fclose(fid);  
% -----  
fid = fopen('OUTPUT_fs32000_T1s.txt', 'w');  
fprintf(fid, 'VRMS: %f\n', VRMS);  
fclose(fid);  
% -----
```

W rezultacie uruchomienia skryptu *demo.m* otrzymuje się dwa pliki tekstowe. Pierwszy plik zawiera wektor wartości sygnału testowego, a drugi plik tekstowy zawiera referencyjną wartość skalarną wskaźnika VRMS dla tego sygnału. Prawidłowo wykonany skrypt *demo.m* umożliwia realizację kolejnych kroków wdrożenia kodu na platformę wbudowaną. W wyniku uruchomienia skryptu otrzymano:

- plik *INPUT\_fs32000\_T1s\_comma\_sep.txt*, zawierający wektor danych <32000 x 1>
- plik *OUTPUT\_fs32000\_T1s.txt*, zawierający tekst „VRMS:144.909546”

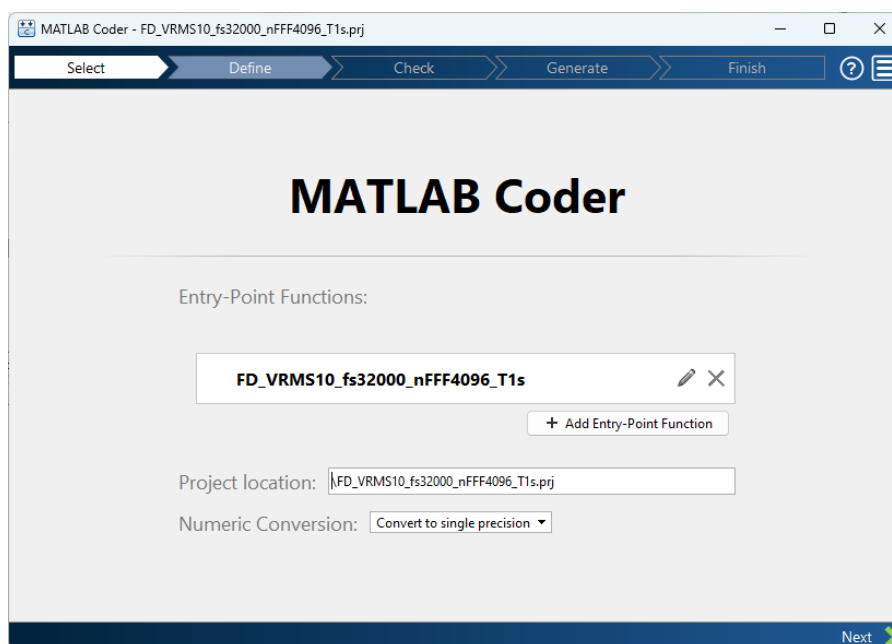
#### **Krok 4: Ustanowienie projektu w narzędziu MATLAB Embedded Coder**

W następnym kroku, uruchamiany jest interfejs MATLAB Coder z zakładki APPS, jak przedstawiono na Rysunku D1.3.



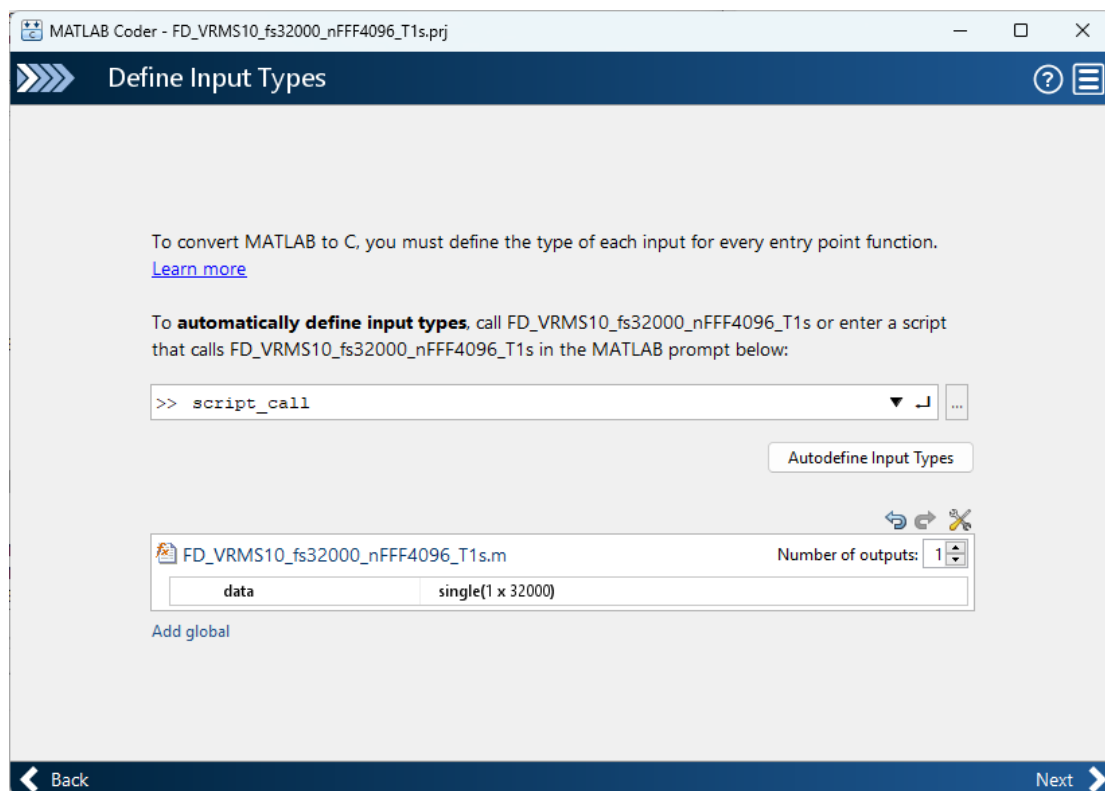
Rysunek D1.3: Uruchomienie narzędzia MATLAB Coder

Następnie, w pierwszym oknie dialogowym wybierane jest funkcja właściwa obliczeniowa, tj. „*FD\_VRMS10\_fs32000\_nFFF4096\_T1s.m*” oraz konwersja danych na pojedynczą precyzję dla badanych procesorów ARM Cortex-M, wg. Rysunku D1.4.



Rysunek D1.4: Przykładowy widok wyboru funkcji i konwersji precyzji danych

Wybór precyzji danych w tym miejscu powoduje, że nie trzeba pamiętać o wszystkich koniecznych miejscach użycia funkcji „*single*” w środowisku MATLAB z uwagi na domyślny format podwójnej precyzji *double*. Jak ilustruje Rysunek D1.5, opracowanie skryptu wywołującego funkcję właściwą umożliwia automatyczne wyznaczenie typu i rozmiaru danych wejściowych.

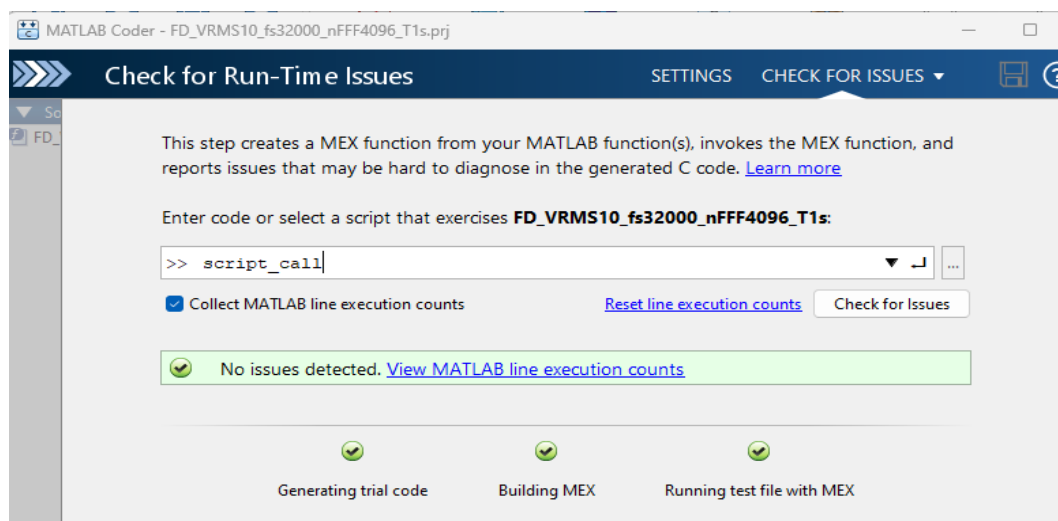


Rysunek D1.5: Wprowadzenie skryptu uruchamiającego funkcję właściwą

Zgodnie z założeniami, funkcja właściwa przyjmuje jedynie wartości bufora wejściowego, tj. wektor 32000 próbek.

### **Krok 5: Sprawdzanie kodu pod kątem kompatybilności**

Jak pokazuje Rysunek D1.6, skrypt testowy wykorzystywane jest kolejno do generowania kodu próbnego, zbudowania plików mex oraz ich przetestowania.



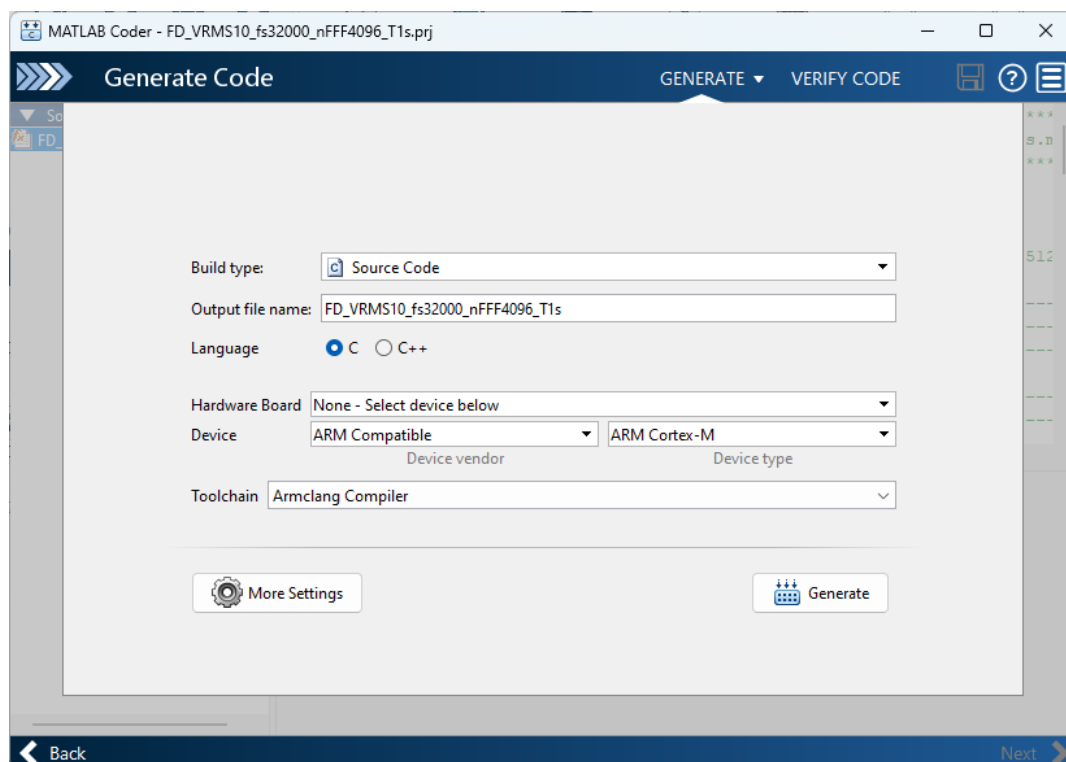
Rysunek D1.6: Wykorzystanie skryptu testowego do testowania kodu

Innymi słowy, na tym etapie MATLAB sprawdza, czy możliwe będzie wygenerowanie kodu w wybranym języku na platformy wbudowane wg. zainstalowanych bibliotek. W praktyce, ten etap może być najbardziej pracochłonny, gdyż wiele popularnych funkcji niskopoziomowych nie jest kompilowalnych.

### **Krok 6: Wybranie cech środowiska docelowego**

Na tym etapie należy kolejno wybrać: język docelowego kodu źródłowego (tu: język C), docelową platformę sprzętową (tu: ARM Cortex-M) oraz Toolchain (tu: Armclang), jak pokazano na Rysunku D1.7. W ramach przeprowadzonych testów, nie zauważono istotnych różnic pomiędzy zastosowaniem kompilatorów:

- GNU Tools for ARMEEmbedded Processors (kompilator domyślny)
- kompilator MinGW-w64 v 6.3 oraz v 8.1 (dodatkowa instalacja kompilatora minGW według [ref D1.2])
- Armclang Compiler (dodatkowa instalacja)

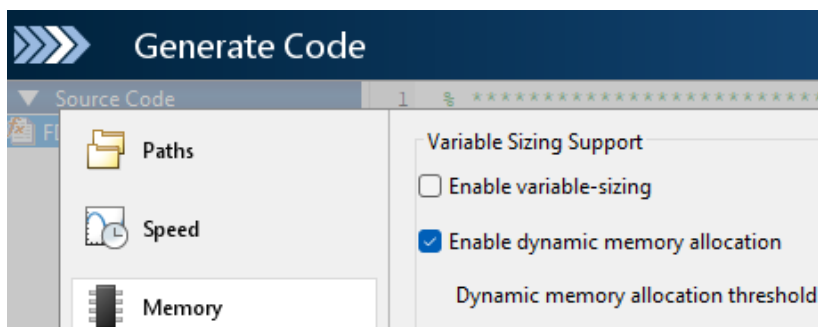


*Rysunek D1.7: Konfiguracja środowiska docelowego – widok ogólny*

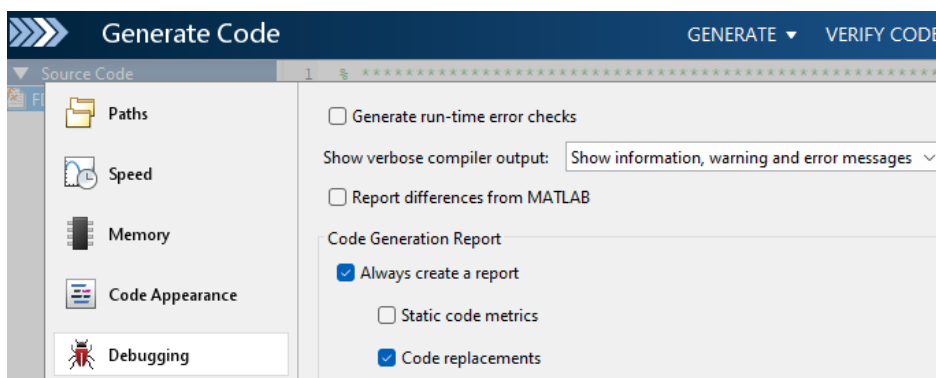
Następnie należy wybrać przycisk ustawień dodatkowych i kolejno:

- w zakładce **Memory** odznaczyć „Enablevariable-sizing”, jak pokazano na Rysunku D1.8,
- w zakładce **Debugging** zaznaczyć pole „Codereplacements”, jak pokazano na Rys. D1.9,
- w zakładce **CustomCode** wybrać pole ARM Cortex-M dla Codereplacementlibrary.

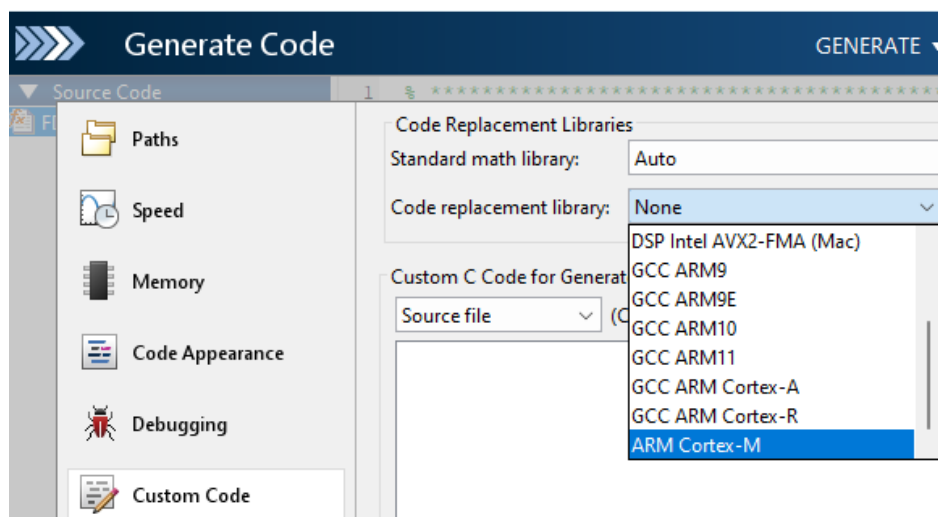
Wprowadzenie wskazanych modyfikacji jest kluczowe dla uzyskania kodu zoptymalizowanego zarówno na poziomie algorytmu, jak również na poziomie obliczeniowym.



Rysunek D1.8: Ustawienia metod alokacji pamięci



Rysunek D1.9: Ustawienia wizualizacji dla bibliotek CMSIS

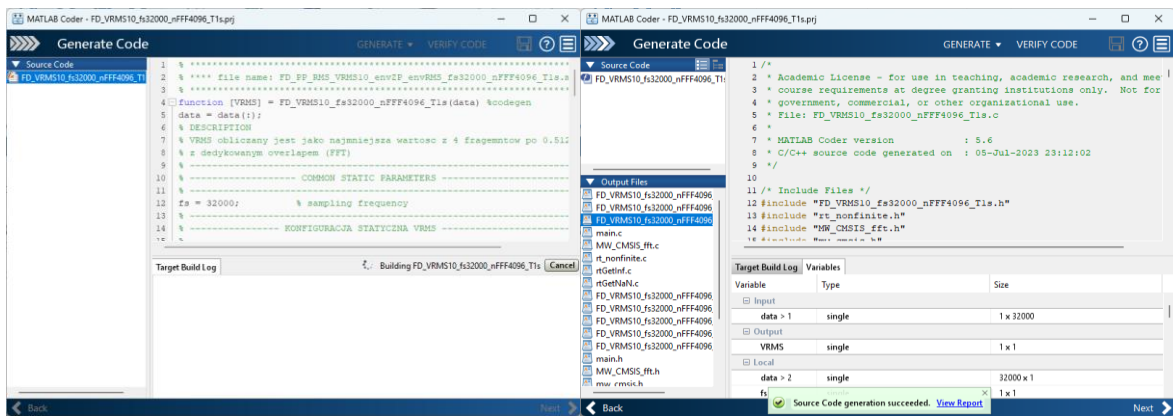


*Rysunek D1.10: Ustawienia wyboru zastępczych bibliotek CMSIS*

Zawartości pól przedstawionych w tym kroku, które użytkownik MATLABA może wybrać zależy od wielu czynników, głównie od zainstalowanych Hardware Support Packages, kompilatorów oraz bibliotek CMSIS. W chwili pisania pracy doktorskiej, najstabilniejsze wyniki kompilacji uzyskiwano dla wersji MATLAB 2022a. Po wprowadzeniu wskazanych zmian wybieramy pole „Generate”.

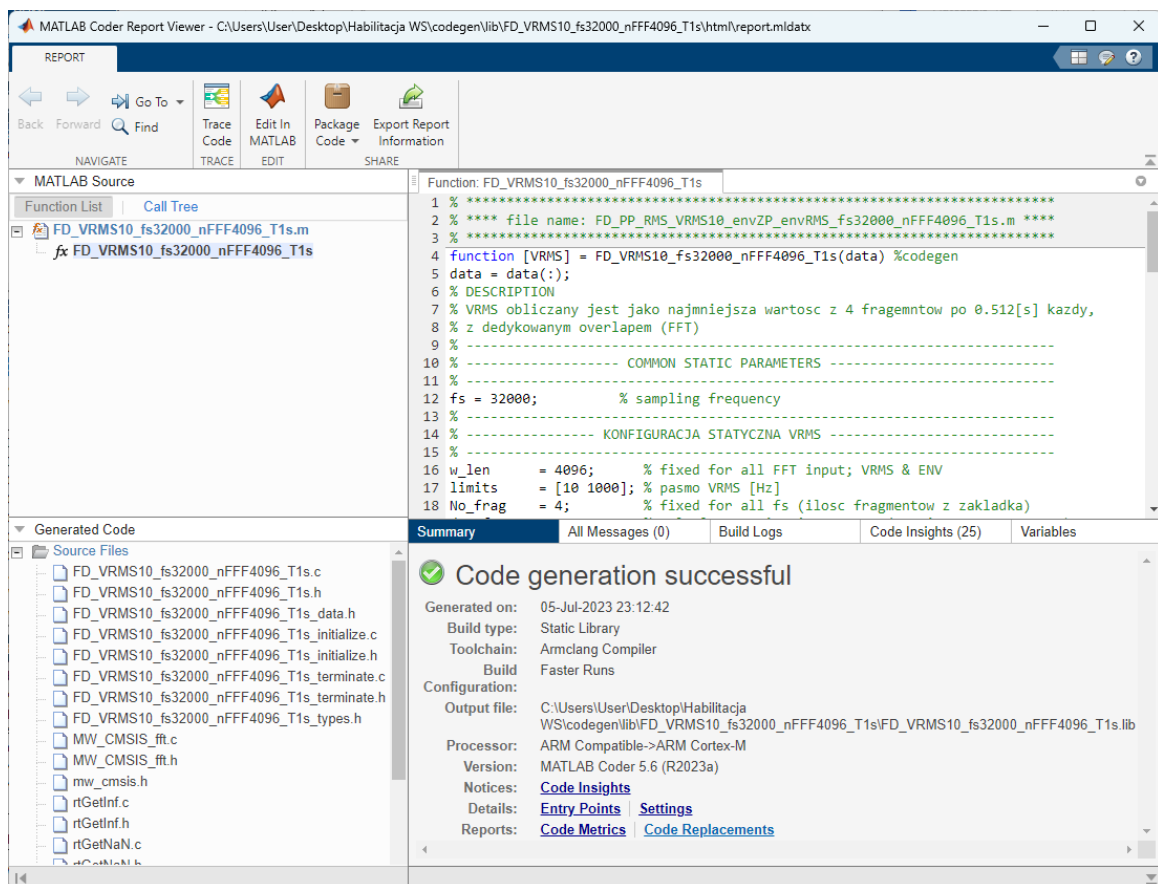
### **Krok 7: Budowa plików źródłowych**

W wyniku działania modułu Embedded Coder Toolbox, środowisko MATLAB generuje kompletny zestaw plików źródłowych na podstawie funkcji właściwej. Na Rysunku D1.11 przedstawiono ideowy widok okna na początku procesu (tylko funkcja w środowisku MATLAB) i na końcu procesu (widok uzupełniony o pliki w języku C).



Rysunek D1.11: Ideowy wynik działania modułu Embedded Coder

Istotnym elementem widocznym na Rysunku D1.11 jest możliwość otwarcia interaktywnego okna, jak pokazano na Rysunku D1.12.



Rysunek D1.12: Widok okna raportu z przejściem do zakładki „CodeReplacement”

W wyniku użycia bibliotek CMSIS przez moduł CodeReplacement Library, narzędzie MATLAB Embedded CoderToolbox wykonało zautomatyzowaną podmianę poszczególnych fragmentów kodu na trzech poziomach. Poziom 1 obejmuje mapowanie funkcje, które

przedstawiono w Tabeli D1.1 Poziom 2 obejmuje pojedynczy operator odejmowania, jak pokazano w Tabeli D1.2 i działanie macierzowe przedstawione w Tabeli D1.3.

*Tabela D1.1: Funkcje zastąpione przez CodeReplacement Library*

Function	MATLAB Function
mw_cmsis_fft_do_real	<a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:3822-3836.#1</a>
fabsf	
memcpy	<a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:2291-2312.#1</a> <a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:2341-2365.#1</a> <a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:2394-2418.#1</a> <a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:2450-2474.#1</a>
mw_arm_sqrt_f32	<a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:4159-4196.#1</a>

*Tabela D1.2: Operator odejmowania zastąpiony przez CodeReplacement Library*

Function	MATLAB Function
mw_arm_sub_f32	<a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:3575-3585.#1</a>

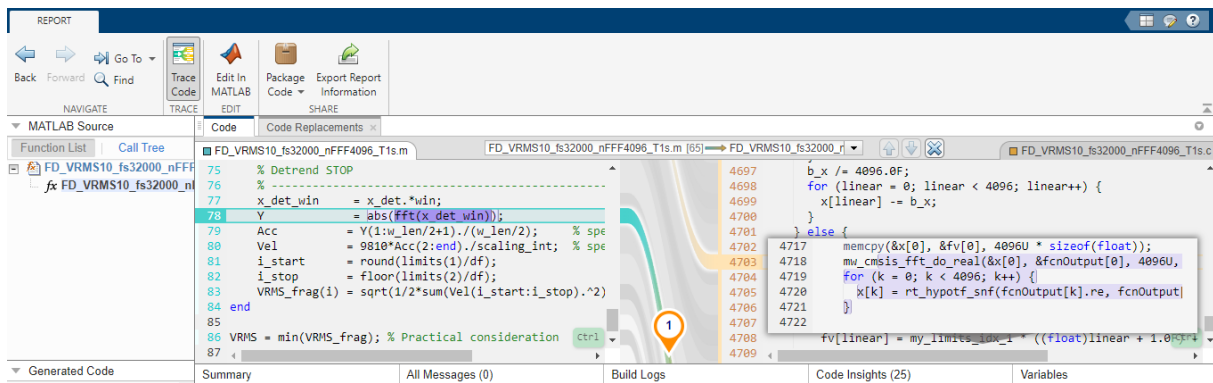
*Tabela D1.3: Działania macierzowe zastąpione przez CodeReplacement Library*

Function	MATLAB Function
mw_arm_mult_f32	<a href="#">FD VRMS10 fs32000 nFFF4096 T1s.m:3786-3796.#1</a>

W kolejnym kroku przedstawiono analizę działania bibliotek CodeReplacement Library w realizowanym programie.

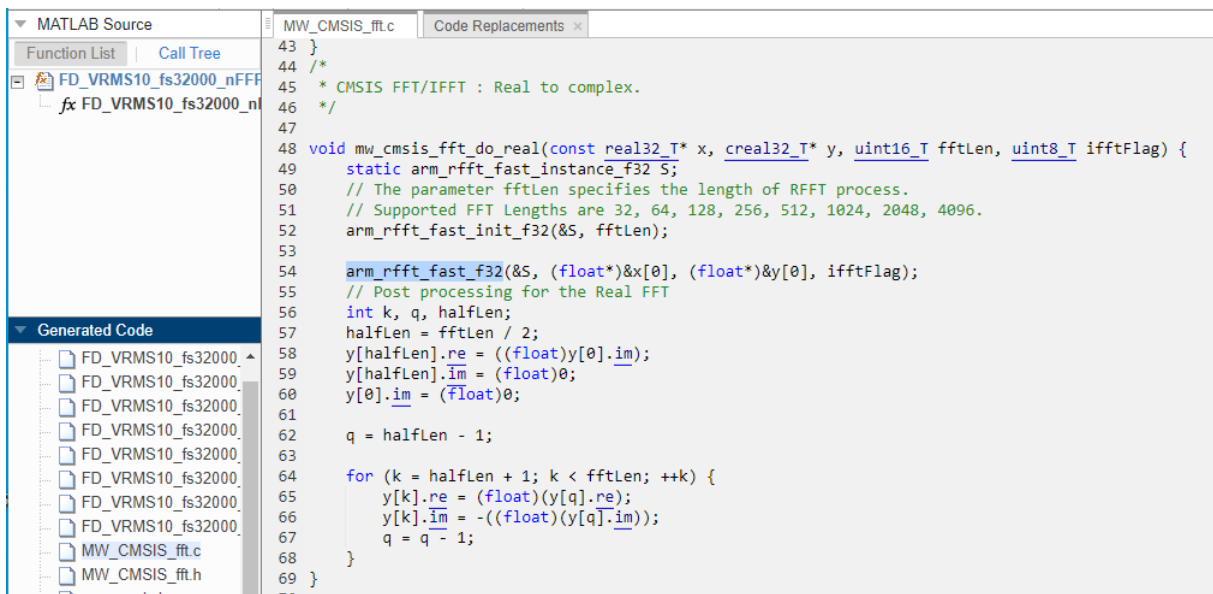
### **Krok 8: Analiza bibliotek zastępczych**

Interfejs MATLAB Coder Report Viewer umożliwia zautomatyzowaną analizę podmiiany kodu dzięki zakładce „TraceCoder”, co pokazano na Rysunku D1.13. Na wskazanym przykładzie widać, że narzędzie automatycznie skorzystało z funkcji *mw\_cmsis\_fft\_do\_real*, której prototyp z łatwością można dynamicznie odnaleźć w za pomocą głównego interfejsu *MATLAB Coder Report Viewer* w pliku *MW\_CMSIS\_fft.c*.



Rysunek D1.13: Analiza działania modułu „CodeReplacement”

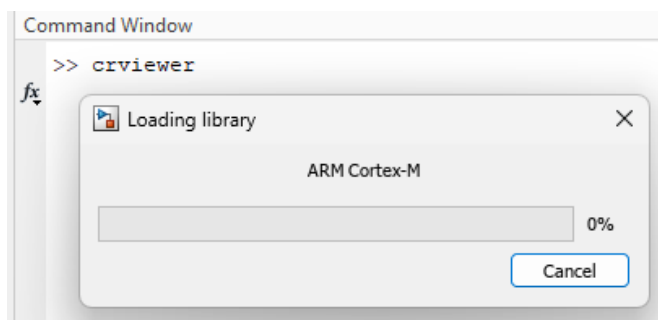
W następnym kroku, należy przeanalizować, które funkcje biblioteki CMSIS zostały użyte w MATLABIE. W tym celu, w rozpatrywanym fragmencie dotyczącym funkcji *fft* należy wybrać funkcję *MW\_CMSIS\_fft.c* w oknie *GeneratedCode*, co pokazano na Rysunku D1.14.



Rysunek D1.14: Ilustracja metody CMSIS w ramach CodeReplacement Library

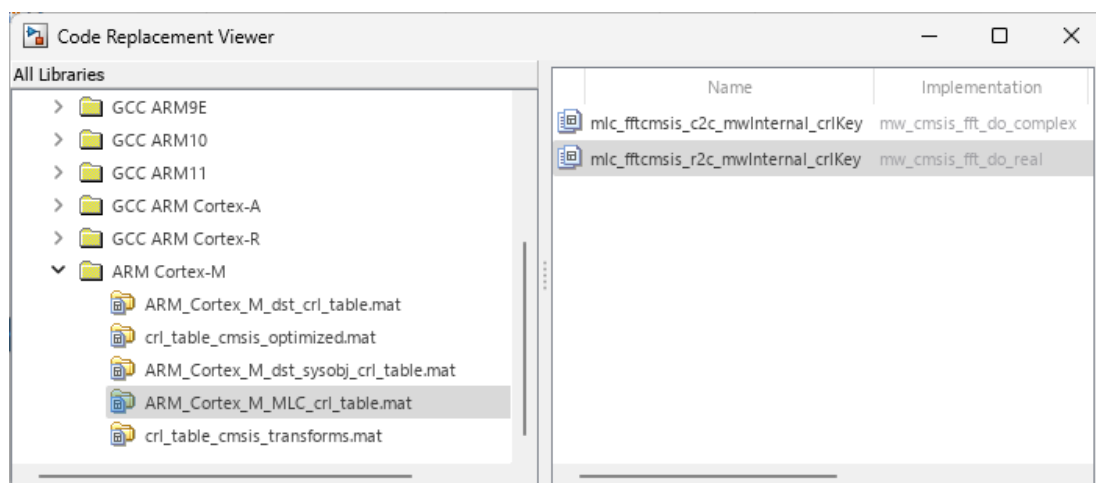
Ponieważ realizacja prac badawczych związanych z rozwojem programów docelowych przeznaczonych do działania na nowym czujniku wymaga dostosowania funkcji przetwarzania sygnałów do pozostałej struktury kodów, w praktyce korzystna jest możliwość modyfikacji kodu na różnych poziomach. W celu powiązania wskazanej funkcji z zestawem biblioteki CMSIS, należy wywołać metodę *crviewer* w wierszu poleceń środowiska MATLAB, jak pokazano na Rysunku D1.14. Środowisko MATLAB załaduje po kolei

wszystkie dostępne biblioteki, zgodnie z kompatybilnymi pakietami zainstalowanymi na komputerze.



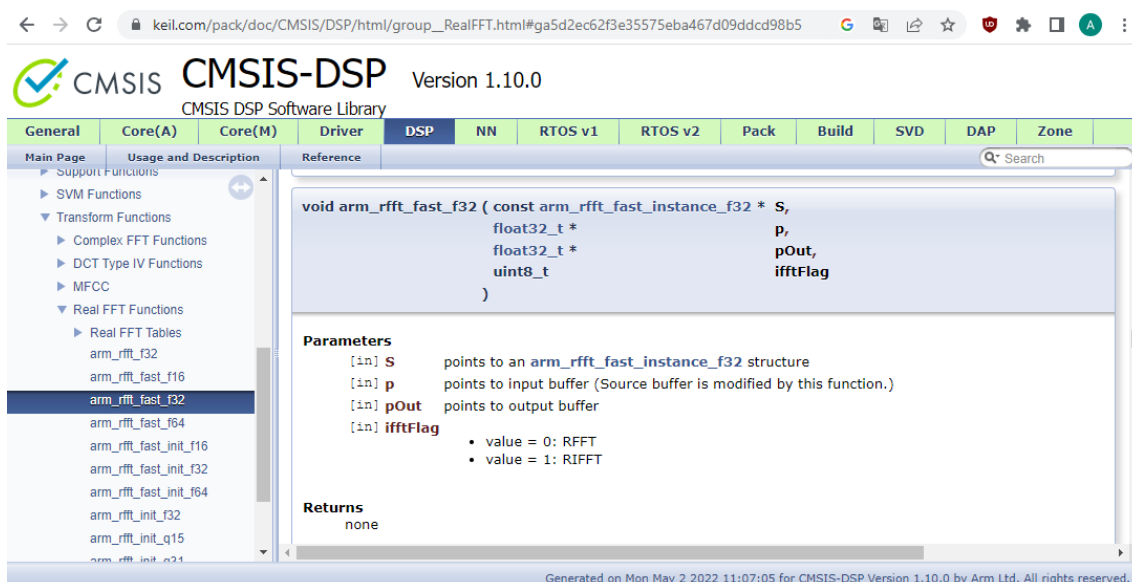
Rysunek D1.15: Wywołanie bibliotek CodeReplacement Library

W następnym kroku należy znaleźć odpowiednią metodę implementacyjną pasującą do kodu. W tym przypadku jest to implementacja `mw_cmsis_fft_do_real`, jak pokazano na Rysunku D1.16. Widok CRL (z ang. *CodeReplacement Library*) warto interpretować jednocześnie z plikiem nagłówkowym `mw_cmsis.h` (czytaj: *Wrapperfunctions for CMSIS functions*) wygenerowanym podczas procesu opisanego w bieżącym Dodatku 1.



Rysunek D1.16: Widok klucza metody `arm_cmsis_fft_do_real` z poziomu CRV

Chociaż widok interfejsu CMSIS z poziomu *MATLAB Coder Report Viewer* narzędzia *MATLAB Embedded CoderToolbox* w połączeniu z interfejsem *CodeReplacement Library* jest wystarczający na poziomie generowania kodu ze środowiska MATLAB, to w ramach prac badawczych korzystna jest możliwość jak najpełniejszego śledzenia kodu źródłowego. W tym celu należy zestawić uzyskane wyniki z danymi od producenta biblioteki DSP CMSIS, którą zilustrowano na Rysunku D1.17.

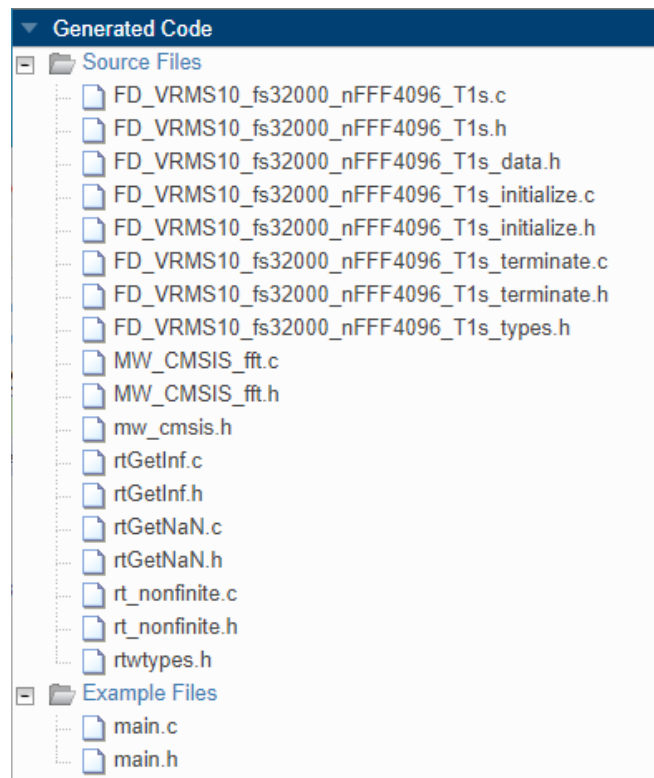


Rysunek D1.17: Widok referencyjny dla CMSIS DSP Software Library, v.1.10.0

Rysunek 17 przedstawia kompletny zestaw parametrów dla szybkiego wyznaczania zespolonych współczynników Transformaty Fouriera dla funkcji `arm_rfft_fast_f32` z biblioteki CMSIS dla procesorów ARM Cortex-M, który dzięki odpowiedniemu wykorzystaniu modułów MATLAB Embedded Coder oraz narzędziu CodeReplacement Library został wygenerowany w sposób zautomatyzowany w środowisku MATLAB.

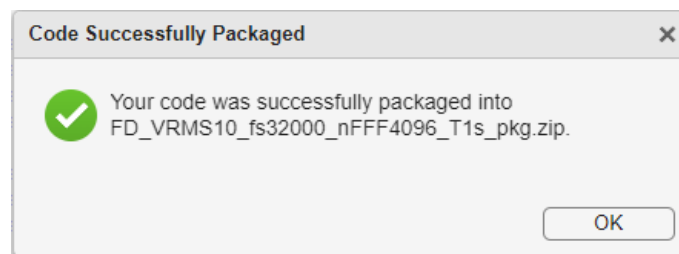
### **Krok 9: Zakończenie generowania kodu źródłowego z poziomu MATLAB**

Dzięki kontrolce graficznej „PackageCode” możliwe jest zebranie (i spakowanie) wygenerowanych plików źródłowych do pojedynczego pliku w formacie zip. MATLAB umożliwi wariant spakowania jako listę plików lub wariant hierarchiczny (z folderami zawierającymi przykłady użycia funkcji). Typowy zestaw plików wygenerowanych przez narzędzie *MATLAB Embedded Coder* wraz z nakładką *CoderReplacement Library* wykorzystującej funkcje biblioteki *CMSIS.DSP* dla pojedynczej przykładowej funkcji `FD_VRMS10_fs32000_nFFT4096_T1.m` pokazano na Rysunku D1.18.



Rysunek D1.18: Przykładowy wynik generacji kodu w MATLAB Embedded Coder

Przedstawiony proces generowania kodu jest zakończony sukcesem, o czym informuje okno modalne modułu *EmbeddedCoder*, przedstawione na Rysunku D1.19.



Rysunek D1.19: Okno zakończenia procedury generacji kodu

### **Krok 10: Utrzymywanie kodu**

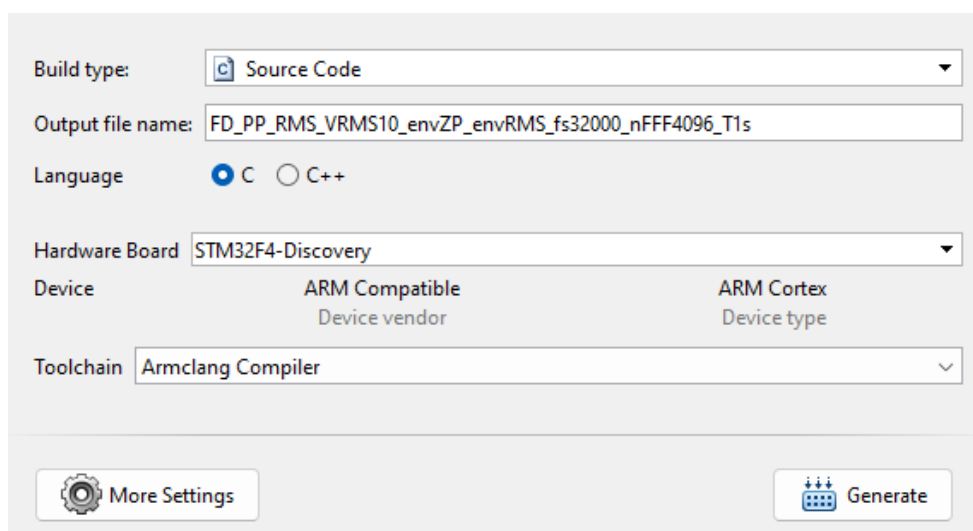
Wykorzystanie modułu spakowania projektu posiada dodatkową zaletę w postaci zapisu pliku *buildInfo.mat*, w którym zapisane są ustawienia użyte w trakcie generowania kodu. W przeciwieństwie do samego pliku projektu (tu domyślna nazwa *FD\_VRMS10\_fs32000\_nFFF4096\_T1s.prj*), plik ustawień zapisuje komplet metadanych umożliwiających odtworzenie wszystkich ustawień projektowych. Pozostałe pliki i ustawienia dotyczące użycia narzędzia *Embedded Coder* są opisane na stronie *mathworks.com*.

## **Krok 11: Wdrażanie kodu na wybraną platformę**

Procedura opisana w Krokach 1-10 wykorzystuje instalację nakładki Embedded CoderSupportPackage for ARM Cortex-M Processors. W przypadku, gdy dostępne są dodatkowe instalatory dla konkretnej platformy wbudowanej, proces generowania kodu może być wykonany jeszcze lepiej. W pracy doktorskiej wykorzystano platformy rodziny STM32F4x z procesorem ARM Cortex-M4, a więc możliwe było doinstalowanie nakładki Embedded CoderSupportPackage for STMicroelectronics STM32 Processors, która obecnie (stan na dzień 08.07.2023 r.) wspiera platformy STM z rodziny:

- STM32F4xx (wybrana w projekcie czujnika),
- STM32F7xx,
- STM32G4xx,
- STM32H7xx.

W celu uzyskania kodu zoptymalizowanego dla konkretnej rodziny platformy STM, należy zmodyfikować Krok 6 poprzez wybór platformy STM, jak pokazano na Rysunku D1.20.



The image shows a configuration window for an embedded code generation tool. It includes several dropdown menus and radio buttons. The 'Build type' is set to 'Source Code'. The 'Output file name' field contains a long alphanumeric string. The 'Language' section has 'C' selected with a radio button. The 'Hardware Board' dropdown is set to 'STM32F4-Discovery'. Below this, there are two columns for 'Device' configuration: 'ARM Compatible' under 'Device vendor' and 'ARM Cortex' under 'Device type'. The 'Toolchain' dropdown is set to 'Armclang Compiler'. At the bottom, there are two buttons: 'More Settings' with a gear icon and 'Generate' with a blue icon of a circuit board.

*Rysunek D1.20: Wybór platformy docelowej STM32F4*

Pozostałe kroki należy przeprowadzić, jak dla wcześniejszej platformy docelowej (tj. wybór ustawień w Kroku 6 oraz Kroki 7-10).

## Dodatek 2 – Analiza wyników liczbowych

Dodatek 2 przedstawia porównanie wyników uzyskanych z przetwarzania sygnału testowego o długości 32000 próbek, odpowiadającego sygnałowi o długości 1 sekundy zarejestrowanego z częstotliwością próbkowania 32kHz). W odniesieniu do wyników zilustrowanych w Dodatku 1, Dodatek 2 przedstawia szerszy zestaw najistotniejszych skalarnych wskaźników diagnostycznych obliczanych na platformie wbudowanej czujnika.

-----  
Wyniki z przetwarzania w środowisku MATLAB:  
-----

PP: 8.553005  
RMS: 0.815532  
VRMS: 0.576921  
envZP: 4.090857  
envRMS: 1.126711  
-----

-----  
Wyniki z przetwarzania na platformie STM32F413CHU6 FFT\_CMSIS + Armclang  
-----

PP: 8.553005  
RMS: 0.815532  
VRMS: 0.576920  
envZP: 4.082310  
envRMS: 1.106375  
-----

Przedstawione wyniki wskazują, że w zależności od ścieżki przetwarzania sygnałów różnica w wynikach się zmienia. W przypadku prostego porównania amplitud (wskaźnik PP), różnica jest niezauważalna (pojedyncza precyzja). Podobnie, pojedyncza operacja pierwiastka, dzielenia, sumy oraz pętli kwadratów (wskaźnik RMS) daje takie same wyniki. W pozostałych przypadkach, kiedy obliczenia wykorzystują transformaty matematyczne oraz mnożenia wektorowe (wskaźniki VRMS, envelope Zero-Pear raz envelope RMS), wyniki różnią się odpowiednio:  $< 1‰$ , 0.2% oraz  $< 2\%$ . Ponieważ zgodnie z normą ISO 20816 wartości graniczne drgań pomiędzy klasami stanu technicznego różnią się na poziomie pierwszego miejsca po przecinku, osiągnięte wyniki można uznać za dostatecznie dokładne.

### Dodatek 3 – Porównanie czasu wykonania FFT

Dodatek 3 przedstawia wynik analizy czasu wykonania funkcji obliczającej zespolone współczynniki transformaty Fouriera dla trzech wariantów w wersji finalnej modelu prototypu czujnika. W wyniku przeprowadzonych prac można było oszacować, jakie korzyści niesie zastosowanie kolejnych poziomów optymalizacji obliczeniowej. Testy wykonano na platformie STM32 z procesorem Cortex-M4. Podane czasy zostały wyznaczone jako średnia z kolejnych 10 prób.

```
- czas wykonania FFT_4096_MATLAB_CODER          0.3395 [ms]
- czas wykonania FFT_4096_MATLAB_EMBEDDED_CODER_M4 0.2335 [ms]
- czas wykonania FFT_4096_CMSIS                  0.1428 [ms]
```

Pierwsza próba dotyczy funkcji *fft* środowiska MATLAB (R2022a) przekonwertowanej bezpośrednio na język C za pomocą narzędzia MATLAB Coder. Aby taka konwersja była możliwa należy opakować funkcję *fft* w funkcję (m-plik) o innej nazwie – i to ona jest konwertowana. Druga próba dotyczy przekonwertowania tej samej funkcji z bardziej szczegółowym wyborem docelowego procesora, tu Cortex-M. Trzecia próba polega na dodatkowym użyciu nakładki CoderReplacement Library (proces opisany w Dodatku 1), która wykorzystuje bezpośrednią funkcję *arm\_rfft\_fast\_f32*, która jest częścią unormowanego interfejsu programistycznego CMSIS. Jak wykazały przeprowadzone próby, każda kolejne zastosowane narzędzie istotnie zmniejsza czas wykonania *fft*. Zastosowanie MATLAB Embedded Coder dla procesora ARM Cortex-M4 zmniejsza czas wykonania *fft* o 32% względem samego MATLAB Coder, natomiast dodatkowe użycie CodeReplacement Library zmniejsza czas wykonania *fft* aż o 58% względem samego użycia Toolboxa MATLAB Coder. W ramach wniosku z analizy można uznać, że użycie dedykowanych funkcji z interfejsu CMSIS.DSP jest bardzo korzystne.

## Dodatek 4 – Porównanie czasów obliczenia wskaźników

Dodatek 4 przedstawia krzyżowe porównanie czasu obliczenia wartości zdefiniowanych wskaźników diagnostycznych z sygnału 1-sekundowego przy częstotliwości próbkowania równej 32 kHz. W pierwszym porównaniu (pkt 1 vs pkt 2) zbadano jaki wpływ na łączny czas obliczenia wartości wskaźników diagnostycznych ma zastosowanie funkcji `arm_rfft_fast_f32` z biblioteki CMSIS.DSP poprzez użycie nakładki CodeReplacement Library. W drugim porównaniu (pkt 2 vs pkt 3) sprawdzono, jakie korzyści daje użycie kompilatora Armclang(Arm® Compiler) w porównaniu z kompilatorem GNU domyślnie używanym przez środowisko MATLAB.

```
1.
----- FFT_MATLAB + Armclang -----
Time elapsed: 750 [ms]
PP:      8.553005
RMS:     0.815532
VRMS:    0.576920
envZP:   4.082310
envRMS:  1.106375
-----

2.
----- FFT_CMSIS + Armclang -----
Time elapsed: 269 [ms]
PP:      8.553005
RMS:     0.815532
VRMS:    0.576920
envZP:   4.082310
envRMS:  1.106375
-----

3.
----- FFT_CMSIS + GNU -----
Time elapsed: 269 [ms]
PP:      8.553005
RMS:     0.815532
VRMS:    0.576920
envZP:   4.082310
envRMS:  1.106375
-----
```

Jak wykazały przeprowadzone testy w ramach porównania pierwszego, z punktu widzenia praktycznego, zastosowanie bibliotek CMSIS.DSP jest konieczne, aby system wbudowany mógł zdążyć wykonać inne operacje związane z komunikacją, gdyż użycie zrekonstruowanej funkcji `fft` środowiska MATLAB powoduje 75% zużycie czasu dostępnego w oknie 1-sekundowym. W przypadku użycia biblioteki CMSIS.DSP za pomocą MATLAB Embedded CoderToolbox wraz z nakładką CodeReplacement Library umożliwiającą bezpośrednio użycie funkcji `arm_rfft_fast_f32`, zużycie spada do niecałych 27%, co pozostawia wystarczającą ilość czasu na obsługę pozostałych wątków systemu wbudowanego. Z kolei drugie porównanie wykazuje, że dla opracowanych kodów przetwarzania sygnałów, zastosowanie dedykowanego kompilatora `armclang` nie przynosi korzyści względem

kompilatora GNU standardowo używanego przez Toolbox MATLAB  
Coder/EmbaddedCoder.

## Dodatek 5 – Instrukcja przekształcenia *dsp.BiquadFilter* na *arm\_biquad\_cascade\_df2T\_f32*

Filtracja w środowisku MATLAB jest zagadnieniem niezwykle obszernym i obejmuje projektowanie filtrów cyfrowych i analogowych, dyskretyzację, różne operacje filtracji oraz graficzne interfejsy wsparcia do projektowania, analizy i wizualizacji filtrów. Również rodzaje filtrów dostępne w środowisku MATLAB można rozpatrywać w wielu kategoriach, m.in. dziedziny, składni, typu, możliwości adaptacji, struktury, reprezentacji, etc. W konsekwencji, projektując ten sam (analitycznie) filtr różnymi ścieżkami (zwłaszcza wykorzystując funkcje z różnych Toolboxów), filtry mogą być zapisane w przestrzeni roboczej środowiska MATLAB albo jako macierze albo jako obiekty systemowe różnych klas (*digitalFilter*, *df2sos*, etc.), co może utrudniać wybór najwłaściwiej ścieżki filtracji. Sam proces filtracji, jako operacji przetwarzania sygnału cyfrowego, również może przebiegać w środowisku MATLAB pośrednio (np. funkcje *step*, *filter*, *filtfilt*) albo bezpośrednio (np. funkcja *highpass*). Analizując dokumentację *mathworks.com* [D5.1 – D5.4], interfejs *crviewer* oraz interfejs programistyczny CMSIS wyciągnięto trzy istotne wnioski:

**Po pierwsze**, konieczność użycia CMSIS.DSP ogranicza zastosowanie filtrów do obiektów systemowych czterech klas:

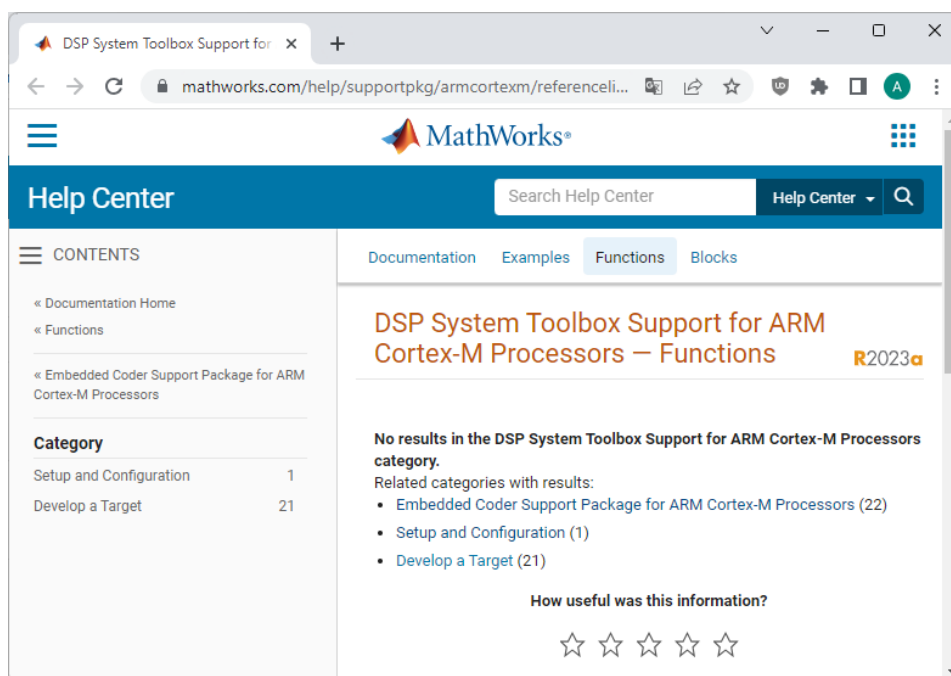
- filtr klasy *dsp.FIRFilter* (pominięty ze względu na dużą ilość współczynników),
- filtr klasy *dsp.LMSFilter* (pominięty z uwagi na charakter adaptacyjny),
- filtr klasy *dsp.BiquadFilter* (analizowany),
- filtr klasy *dsp.SOSFilter* (analizowany),

z których wszystkie są obiektami systemowymi *Digital Signal Processing (DSP) Toolbox*. Nie znaleziono innych funkcji, których można użyć. Należy zauważyć, że jest to sytuacja inna, niż w przypadku funkcji *fft*, która może być implementowana albo z poziomu *Signal Processing Toolbox* (maksymalnie 4096 próbek) albo z poziomu *DSP Toolbox* (maksymalnie 1024 próbki).

**Po drugie**, obiekty *dsp.SOSFilter* (wprowadzone w wersji R2020a z możliwością przekształcania na kod C/C++ od wersji 2022b) mogą być traktowane jako uzupełnienie obiektów *dsp.BiquadFilter*, które zostały wprowadzone w środowisku MATLAB już w wersji R2012a, gdyż obie klasy zwracają filtr *biquad* (skrót od ang. biquadratic, ) o nieskończonej

odpowiedzi (z ang. IIR, *Infinite Impulse Response*), złożony z kaskady filtrów SOS (z ang. *Second-Order-Section*) [D5.5 - D5.6]. Z uwagi na pojedynczą precyzję numeryczną oraz możliwy niewielki stosunek częstotliwości odcięcia do częstotliwości próbkowania (np. 2 Hz vs. 32kHz), preferowaną klasą jest klasa *dsp.BiquadFilter*, gdyż w odróżnieniu od klasy *dsp.SOSFilter* wykorzystującą strukturę „licznik i mianownik”, operuje na dokładniejszej reprezentacji zer, biegunów i wzmocnienia w dziedzinie  $Z$ ,  $[z,p,k]$ .

**Po trzecie**, dokumentacja mathworks.com w zakresie badanym w ramach realizacji pracy doktorskiej, podobnie jak w przypadku tamatu „*MATLAB Functions for ARM Cortex-M Category*” (zob. Dodatek 1 Rysunek 1) wyświetla informację o pustej liście rezultatów, co pokazano na Rysunku D5.1, stąd można uznać, że opracowane kody mają charakter rozwojowy i dlatego umieszczono je w pracy.



Rysunek D5.1: Widok braku wyników wsparcia dla badanego zastosowania

Zasadniczo, procedura zautomatyzowanego generowania kodu w języku C na podstawie kodu wzorcowego w środowisku MATLAB z wykorzystaniem Toolboxa MATLAB *Embedded Coder* oraz odpowiednich metod z interfejsu programistycznego CMSIS.DSP przez nakładkę *CodeReplacement Library* jest podobna do tej opisanej w Dodatku 1, jednak konieczne jest wprowadzenie zmian, które nie wynikają z dokumentacji środowiska MATLAB (stan do bieżącej wersji, tj. R2023a). Poniższy kod przedstawia składnię zgodną z wytycznymi dokumentacji [D5.7], jednak błędną podczas kompilacji. Należy podkreślić, że na etapie dewelopmentu, wygodnie jest pracować na jednej funkcji, w której definiowane są parametry

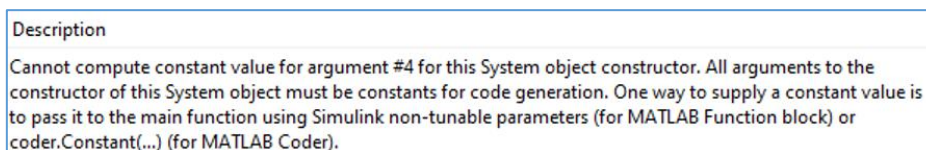
filtracji, tworzony jest filtr (tu obiekt systemowy) oraz przeprowadzana jest operacja filtracji, stąd badany kod wygląda następująco:

```
function y = filter_TD_dsp_BiquadFilter_HP10_fs32000_n4(x)
n = 4; % rzad filtru
fs = 32000; % czestotliwosc probkowania [Hz]
fc = 10; % czestotliwosc odciecia [Hz]
type = 'high'; % filtr gorno-przepustowy

[z,p,k] = butter(n, fc/(fs/2), type); % projekt filtru
[sos,g] = zp2sos(z,p,k); % konwersja
bqf = dsp.BiquadFilter('Structure', 'Direct form II transposed',...
'SOSMatrix', sos, ...
'ScaleValues', g); % obiekt systemowy

% DOMYSLNIE
% bqf.SOSMatrixSource = 'Property';
% bqf.InitialConditions = 0;
% bqf.ScaleValuesInputPort = 1;
y = bqf(x); % lub: y = step(bqf,x);
```

Podczas kompilacji narzędzie *MATLAB Embedded Coder* wskazuje błąd, polegający na tym, że generator wymaga, aby wszystkie argumenty konstruktora klasy obiektu systemowego były typu „constant” – i podpowiada, aby użyć funkcji „*coder.Constant*”, jak pokazano na Rysunku D.5.2.

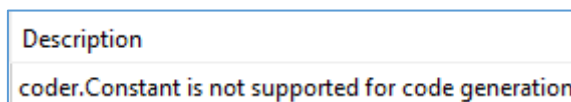


Rysunek D5.2: Komunikat MATLAB Embedded Coder: *coder.Constant* zalecany

Postępując zgodnie z zaleceniami narzędzia, podczas drugiej próby zmodyfikowano odpowiednie fragmenty kodu:

```
sos = coder.Constant(sos);
g = coder.Constant(g);
```

co generuje kolejny błąd, z uwagi na to, że klasa *Constant* nie jest wspierana przez narzędzie MATLAB Embedded Coder, jak pokazano na Rysunku D 5.3.



Rysunek D5.3: Komunikat MATLAB Embedded Coder: *coder.Constant* niewspierany (w tym scenariuszu)

Możliwą przyczyną niespójności jest podobieństwo składni w środowisku MATAB „*coder.const*” (pre-ewaluacja wyrażeń i zastąpienie ich wartościami statycznymi) oraz „*coder.Constant*” (klasa). W przyjętej architekturze kodu testującego problem można prosto rozwiązać podając parametry filtru w sposób statyczny wewnątrz funkcji, jako poniżej:

```

function y = filter_TD_dsp_BiquadFilter_HP10_fs32000_n4(x)
% n = 4; % rzad filtru
% fs = 32000; % czestotliwosc probkowania [Hz]
% fc = 10; % czestotliwosc odcięcia [Hz]
% type = 'high'; % filtr gorno-przepustowy
%
% [z,p,k] = butter(n, fc/(fs/2), type); % projekt filtru
% [sos,g] = zp2sos(z,p,k); % konwersja

% Wynikowe parametry filtru
sos = [
    1.0000000 -2.0000000 1.0000000 1.0000000 -1.9963746 0.9963785
    1.0000000 -2.0000000 1.0000000 1.0000000 -1.9984945 0.9984983];

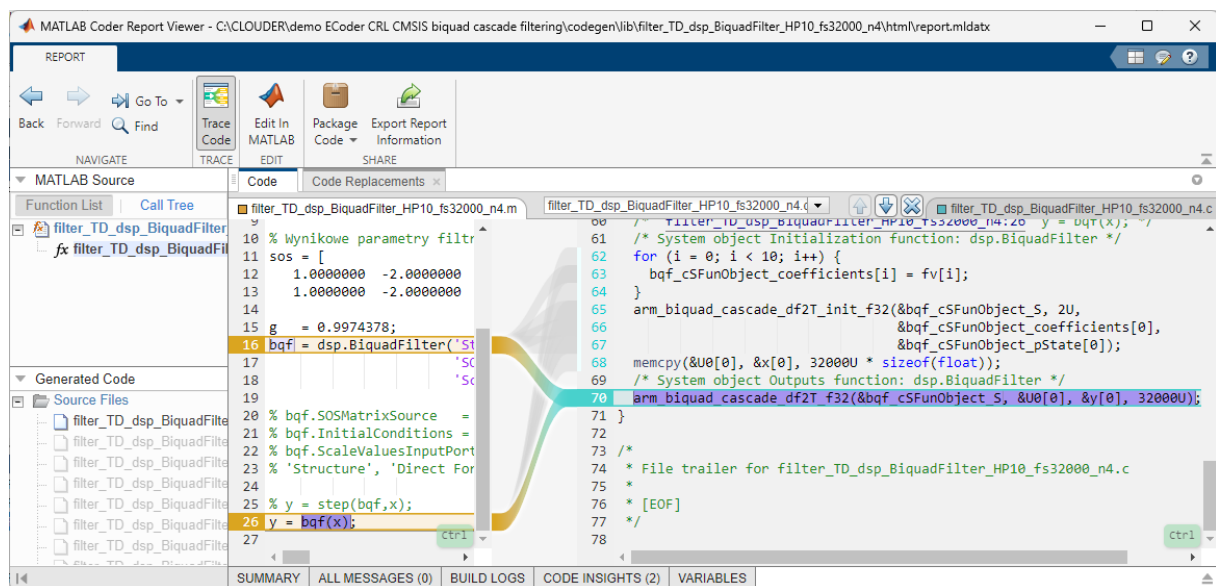
g = 0.9974378;
bqf = dsp.BiquadFilter('Structure', 'Direct form II transposed',...
    'SOSMatrix', sos, ...
    'ScaleValues', g);

% bqf.SOSMatrixSource = 'Property';
% bqf.InitialConditions = 0;
% bqf.ScaleValuesInputPort = 1;
% 'Structure', 'Direct Form I'

% y = step(bqf,x);
y = bqf(x);

```

Po dalszych krokach użycia narzędzia MATLAB EmbeddedCoder (zob. Dodatek 1), uzyskuje się kod w języku C wykorzystujący funkcję filtrującą, jak pokazano na Rysunku D5.4.



Rysunek D5.4: Ilustracja działania CodeReplacement Library dla filtracji

Rysunek D5.4 pokazuje zaznaczoną zamianę dwóch funkcji środowiska MATLAB, tj. funkcji projektowania obiektu systemowego klasy *dsp.BiquadFilter* z biblioteki *Signal Processing Toolbox* oraz funkcji filtrującej (której składania odpowiada użyciu funkcji *step*) na funkcję *arm\_biquad\_cascade\_df2T\_f32* interfejsu programistycznego CMSIS.DSP dla procesorów ARM Cortex-M dzięki nakładce *CodeReplacement Library* narzędzia MATLAB Embedded Coder. Pomimo licznych przeprowadzonych prób, nie udało się opracować programu, który

byłby przekształcalny w ten sam sposób, ale w którym projektowanie filtru odbywałoby się w pełni dynamicznie, wzorując się na [D5.8]. Postać skryptowa generowania kodu, zamieszczona dla referencji) wygląda następująco:

```
cfg = coder.config('lib','ecoder',true);
cfg.HardwareImplementation.ProdHWDeviceType = 'ARM Compatible->ARM Cortex-M';
cfg.HardwareImplementation.TargetHWDeviceType = 'ARM Compatible->ARM Cortex-M';
cfg.EnableVariableSizing = false;
cfg.GenerateReport = true;
cfg.GenerateCodeReplacementReport = true;
cfg.HighlightPotentialDataTypeIssues = true;
cfg.MATLABSourceComments = true;
cfg.ReportPotentialDifferences = false;
cfg.CodeReplacementLibrary = 'ARM Cortex-M';
cfg.Toolchain = 'Armclang Compiler';
cfg.GenCodeOnly = true;
my_ARGS{1} = [];
ARGS{1}{1} = coder.typeof(single(0), [32000 1]);
codegen -config cfg -singleC filter_TD_dsp_BiquadFilter_HP10_fs32000_n4 -args ARGS{1}
```

Powyższy kod wymaga funkcji *filter\_TD\_dsp\_BiquadFilter\_HP10\_fs32000\_n4* zamieszczonej w Dodatku 5.

## Dodatek 6 – Adaptacja metod automatycznej walidacji sygnałów drgań

W warunkach przemysłowych, sygnały drgań mogą zawierać różnego rodzaju artefakty, których źródłem mogą być zjawiska mechaniczne, elektroniczne, skutki uboczne przetwarzania sygnałów lub inne, nieokreślone czynniki [D6.1-2]. W pracy [D6.2] przedstawiono propozycję szeregu metod przetwarzania sygnałów, które mają na celu automatyczną walidację sygnałów drgań, czyli stwierdzenie, czy dany sygnał drgań jest istotny – tj. czy jego analiza i przetwarzanie mogą być przydatne w procesie oceny stanu technicznego maszyny (dotyczy zarówno informacji o uszkodzeniu, jak i o jego braku). Odwołując się do Rysunku 8.1, w przypadku, gdy cyfrowy czujnik drgań przesyła surowy przebieg czasowy, automatyczna walidacja może spowodować, że w bazie zapisanych sygnałów nie będzie sygnałów, które są bezużyteczne (albo szkodliwe podczas dalszego przetwarzania, np. trenowania modelu lub wnioskowania diagnostycznego). W przypadku, gdy cyfrowy czujnik drgań przesyła obliczone wartości wskaźników diagnostycznych, automatyczna walidacja może bezpośrednio zredukować omówiony wcześniej współczynnik fałszywych alarmów FAR.

W ramach przeprowadzonych analiz i testów, dla cyfrowych czujników drgań zaproponowano użycie trzech metod spośród algorytmów opisanych w [D6.1 - 6.2], które były zaprojektowane dla sygnałów zarejestrowanych przez analogowe piezoelektryczne czujniki drgań. Przeprowadzone badania miały na celu również dobór odpowiednich parametrów algorytmów automatycznej walidacji, ze względu na różne parametry akwizycji badanych czujników względem pracy [D6.2], tj. inną długość sygnałów i inną częstotliwość próbkowania oraz ze względu na różnice w charakterystyce odpowiedzi układów, zwłaszcza inną stałą czasową cyfrowych czujników drgań opracowanych w technologii MEMS. Opracowane kody źródłowe wraz z odpowiednikami w języku C przedstawiono w Tabeli D6.1.

*Tabela D6.1: Automatyczna walidacja sygnałów drgań dla czujników cyfrowych*

### **Metoda „N-point”**

Metoda *N-point* wyznacza największą ilość elementów wektora o tej samej wartości następujących po sobie, np. dla wektora  $([0,1,0,1,1,1,0,1])$  metoda „N-point” zwraca wartość „3”. Zgodnie z [D6.2], metoda *N-point* jest przydatna do wykrywania przebiegów o zbyt niskim SNR (tj. niskim poziomie amplitud względem zakresu

pomiarowego).

#### Kod w środowisku MATLAB

```
function [No_Npoint] =  
Validation_No_Npoint(x)  
% Function calculates maximum number  
% of consecutive points of the same  
% value, e.g.: [3] = ...  
% Validation_No_Npoint ...  
% ([0,1,0,1,1,1,0,1])  
counter_1 = 1;  
counter_2 = 1;  
for i = 2:length(x)  
    if x(i) == x(i-1)  
        counter_2 = counter_2 + 1;  
    else  
        counter_2 = 1;  
    end  
    if counter_2 > counter_1  
        counter_1 = counter_2;  
    end  
end  
No_Npoint = max(counter_1,  
counter_2);
```

#### Kod w języku C

```
double Validation_No_Npoint(const  
emxArray_real_T *x)  
{  
    const double *x_data;  
    double counter_1;  
    double counter_2;  
    int b_i;  
    int i;  
    x_data = x->data;  
    counter_1 = 1;  
    counter_2 = 1;  
    i = x->size[1];  
    for (b_i = 0; b_i <= i - 2; b_i++) {  
        if (x_data[b_i + 1] == x_data[b_i]) {  
            counter_2++;  
        } else {  
            counter_2 = 1;  
        }  
        if (counter_2 > counter_1) {  
            counter_1 = counter_2;  
        }  
    }  
    return fmax(counter_1, counter_2);  
}
```

### Metoda „ZCR”

Metoda ZCR (z ang. *Zero-Crossing-Rule*) wyznacza największą ilość elementów wektora następujących po sobie, które nie przekraczają wartości „0”, np. dla wektora  $([-1,1,2,3,2,1,-1,-2])$  metoda ZCR zwraca wartość „5”. Zgodnie z [D5.2], metoda ZCR jest przydatna do wykrywania przebiegów zawierających zbyt długie czasy stabilizacji drań (zwłaszcza pojedyncze odpowiedzi impulsowe generowane zakłóceniami mechanicznymi, np. przypadkowym uderzeniem w czujnik).

```
function [No_ZCR] =  
Validation_No_ZCR(x)  
No_ZCR =  
max(max(diff([0, find(~[0, x < 0, 0])]) -  
1), ...  
max(diff([0, find(~[0, x > 0, 0])]) -  
1));  
end  
% Zrodlo: [D5.2]
```

```
#include "diff.h" //MATALB ECoder  
#include "find.h" //MATALB ECoder  
#include "minOrMax.h" //MATALB ECoder  
#include "rt_nonfinite.h" //MATALB ECoder  
#include <math.h> //MATALB ECoder  
  
double Validation_No_ZCR(const  
emxArray_real_T *x)  
{  
    emxArray_boolean_T *r;  
    emxArray_int32_T *r2;  
    emxArray_real_T *r3;  
    emxArray_real_T *r6;  
    emxArray_real_T *r8;  
    const double *x_data;  
    double No_ZCR;  
    double *r5;  
    double *r7;  
    double *r9;
```

```

int i;
int loop_ub;
int *r4;
boolean_T *r1;
x_data = x->data;
emxInit_boolean_T(&r);
i = r->size[0] * r->size[1];
r->size[0] = 1;
r->size[1] = x->size[1] + 2;
emxEnsureCapacity_boolean_T(r, i);
r1 = r->data;
r1[0] = true;
loop_ub = x->size[1];
for (i = 0; i < loop_ub; i++) {
    r1[i + 1] = !(x_data[i] < 0.0);
}
emxInit_int32_T(&r2);
emxInit_real_T(&r3, 2);
r1[x->size[1] + 1] = true;
eml_find(r, r2);
r4 = r2->data;
i = r3->size[0] * r3->size[1];
r3->size[0] = 1;
r3->size[1] = r2->size[1] + 1;
emxEnsureCapacity_real_T(r3, i);
r5 = r3->data;
r5[0] = 0.0;
loop_ub = r2->size[1];
for (i = 0; i < loop_ub; i++) {
    r5[i + 1] = r4[i];
}
emxInit_real_T(&r6, 2);
diff(r3, r6);
r7 = r6->data;
i = r->size[0] * r->size[1];
r->size[0] = 1;
r->size[1] = x->size[1] + 2;
emxEnsureCapacity_boolean_T(r, i);
r1 = r->data;
r1[0] = true;
loop_ub = x->size[1];
for (i = 0; i < loop_ub; i++) {
    r1[i + 1] = !(x_data[i] > 0.0);
}
r1[x->size[1] + 1] = true;
eml_find(r, r2);
r4 = r2->data;
i = r3->size[0] * r3->size[1];
r3->size[0] = 1;
r3->size[1] = r2->size[1] + 1;
emxEnsureCapacity_real_T(r3, i);
r5 = r3->data;
r5[0] = 0.0;
loop_ub = r2->size[1];
emxFree_boolean_T(&r);
for (i = 0; i < loop_ub; i++) {
    r5[i + 1] = r4[i];
}
emxFree_int32_T(&r2);
emxInit_real_T(&r8, 2);
diff(r3, r8);
r9 = r8->data;
i = r3->size[0] * r3->size[1];
r3->size[0] = 1;
r3->size[1] = r6->size[1];
emxEnsureCapacity_real_T(r3, i);
r5 = r3->data;

```

	<pre> loop_ub = r6-&gt;size[1]; for (i = 0; i &lt; loop_ub; i++) {     r5[i] = r7[i] - 1.0; } i = r6-&gt;size[0] * r6-&gt;size[1]; r6-&gt;size[0] = 1; r6-&gt;size[1] = r8-&gt;size[1]; emxEnsureCapacity_real_T(r6, i); r7 = r6-&gt;data; loop_ub = r8-&gt;size[1]; for (i = 0; i &lt; loop_ub; i++) {     r7[i] = r9[i] - 1.0; } emxFree_real_T(&amp;r8); No_ZCR = fmax(maximum(r3), maximum(r6)); emxFree_real_T(&amp;r3); emxFree_real_T(&amp;r6); return No_ZCR; } </pre>
--	--

### **Metoda “Unique”**

Metoda *Unique* wyznacza łączną ilość elementów wektora o różnej amplitudzie, np. dla wektora  $([-1,0,1,2,1,2])$  metoda *Unique* zwraca wartość „4”. Zgodnie z [D5.2], metoda *Unique* jest przydatna do wykrywania przebiegów zarejestrowanych czujnikami drgań o zbyt małej czułości.

#### *Kod w środowisku MATLAB*

```

function [No_Unique] =
Validation_No_Unique(x)
No_Unique = length(unique(x));
end

```

#### *Kod w języku C*

```

double Validation_No_Unique(const
emxArray_real_T *x)
{
    emxArray_int32_T *idx;
    emxArray_int32_T *iwork;
    emxArray_real_T *b;
    const double *x_data;
    double No_Unique;
    double b_x;
    double *b_data;
    int b_i;
    int i;
    int i2;
    int j;
    int k;
    int n;
    int na;
    int nb;
    int qEnd;
    int *idx_data;
    int *iwork_data;

    boolean_T exitg1;
    x_data = x->data;
    emxInit_real_T(&b, 2);
    emxInit_int32_T(&idx, 2);
    na = x->size[1];
    n = x->size[1] + 1;
    i = idx->size[0] * idx->size[1];
    idx->size[0] = 1;
    idx->size[1] = x->size[1];
    emxEnsureCapacity_int32_T(idx, i);

```

```

idx_data = idx->data;
b_i = x->size[1];
for (i = 0; i < b_i; i++) {
    idx_data[i] = 0;
}

if (x->size[1] != 0) {
    emxInit_int32_T(&iwork, 1);
    i = iwork->size[0];
    iwork->size[0] = x->size[1];
    emxEnsureCapacity_int32_T(iwork, i);
    iwork_data = iwork->data;
    i = x->size[1] - 1;
    for (k = 1; k <= i; k += 2) {
        b_x = x_data[k];
        if((x_data[k-1]<=b_x)||rtIsNaN(b_x)){
            idx_data[k - 1] = k;
            idx_data[k] = k + 1;
        } else {
            idx_data[k - 1] = k + 1;
            idx_data[k] = k;
        }
    }
}
if ((x->size[1] & 1) != 0) {
    idx_data[x->size[1]-1] = x->size[1];
}
b_i = 2;
while (b_i < n - 1) {
    i2 = b_i << 1;
    j = 1;
    for (nb=b_i+1; nb < n; nb=qEnd+b_i) {
        int kEnd;
        int p;
        int q;
        p = j;
        q = nb - 1;
        qEnd = j + i2;
        if (qEnd > n) {
            qEnd = n;
        }
        k = 0;
        kEnd = qEnd - j;
        while (k + 1 <= kEnd) {
            b_x = x_data[idx_data[q] - 1];
            i = idx_data[p - 1];
            if((x_data[i-1]<=b_x)||rtIsNaN(b_x)){
                iwork_data[k] = i;
                p++;
                if (p == nb) {
                    while (q + 1 < qEnd) {
                        k++;
                        iwork_data[k]=idx_data[q];
                        q++;
                    }
                }
            } else {
                iwork_data[k] = idx_data[q];
                q++;
                if (q + 1 == qEnd) {
                    while (p < nb) {
                        k++;
                        iwork_data[k]=idx_data[p-1];
                        p++;
                    }
                }
            }
        }
        k++;
    }
}

```

```

    }
    for (k = 0; k < kEnd; k++) {
        idx_data[(j+k)-1]= iwork_data[k];
    }
    j = qEnd;
}
b_i = i2;
}
emxFree_int32_T(&iwork);
}
i = b->size[0] * b->size[1];
b->size[0] = 1;
b->size[1] = x->size[1];
emxEnsureCapacity_real_T(b, i);
b_data = b->data;
for (k = 0; k < na; k++) {
    b_data[k] = x_data[idx_data[k] - 1];
}
emxFree_int32_T(&idx);
k = 0;
while ((k + 1 <= na) &&
rtIsInf(b_data[k]) && (b_data[k] < 0.0)) {
    k++;
}
b_i = k;
k = x->size[1];
while ((k>=1) && rtIsNaN(b_data[k-1])){
    k--;
}
i2 = x->size[1] - k;
exitg1 = false;
while ((!exitg1) && (k >= 1)) {
    b_x = b_data[k - 1];
    if (rtIsInf(b_x) && (b_x > 0.0)) {
        k--;
    } else {
        exitg1 = true;
    }
}
nb = 0;
if (b_i > 0) {
    nb = 1;
}
while (b_i + 1 <= k) {
    b_x = b_data[b_i];
    do {
        b_i++;
    } while (!((b_i + 1 > k) ||
(b_data[b_i] != b_x)));
    nb++;
    b_data[nb - 1] = b_x;
}
emxFree_real_T(&b);
if ((x->size[1] - k) - i2 > 0) {
    nb++;
}
for (j = 0; j < i2; j++) {
    nb++;
}
if (nb < 1) {
    No_Unique = 0.0;
} else {
    No_Unique = nb;
}
return No_Unique;
}
}

```

--	--