

Міністерство освіти і науки України
Сумський державний університет
Факультет технічних систем та енергоефективних технологій
Кафедра загальної механіки і динаміки машин

КВАЛІФІКАЦІЙНА РОБОТА

магістра

На тему: «Дослідження динамічних характеристик механічних систем з використанням можливостей програмного комплексу LabVIEW»

Напрямок підготовки: **6.040202 «Механіка»**

Фахове спрямування: **«Комп'ютерна механіка»**

Виконав: студент групи **КМ – 51: Логоша В.В.**

Керівник: Савченко Є. М.

Рецензент: Савченко Є. М.

Суми – 2020

РЕФЕРАТ

Кваліфікаційна робота спеціаліста: 62 с., 22 рис., 1 табл., 10 дж.

Об'єкт дослідження: моделювання реальних механічних систем та отримання їх частотних характеристик.

Предмет дослідження: використання програмного комплексу LabVIEW для моделювання реальних механічних систем та отримання їх частотних характеристик.

Мета роботи: створення у програмному комплексі LabVIEW, як середовищі візуального програмування, віртуального приладу для дослідження частотних характеристик реальних механічних систем.

Методи дослідження – комп'ютерне моделювання, засноване на візуальному графічному програмуванні засобами програмного комплексу LabVIEW.

В даній роботі розглянуто можливості середовища візуального програмування LabVIEW для моделювання реальних механічних систем і дослідження їх характеристик, розглянуто приклади побудови віртуальних приладів у програмному комплексі LabVIEW та створено за його допомогою віртуальний прилад для моделювання та дослідження частотних характеристик реальних механічних систем.

КЛЮЧОВІ СЛОВА: МЕХАНІЧНА СИСТЕМА, ЧАСТОТНІ ХАРАКТЕРИСТИКИ, КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ, ВІРТУАЛЬНЕ ПРОГРАМУВАННЯ, ВІРТУАЛЬНИЙ ПРИЛАД, LABVIEW.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ I. ОСНОВИ ПРОГРАМУВАННЯ В ГРАФІЧНОМУ СЕРЕДОВИЩІ LABVIEW.....	8
1.1. Основні характеристики LabVIEW та перспективи розвитку.....	8
1.2. Мова графічного програмування і потоки даних	10
1.3. Специфіка роботи LabVIEW.....	11
Висновки до розділу I.....	15
РОЗДІЛ II. МОДЕЛЮВАННЯ МЕХАНІЧНИХ СИСТЕМ ЗА ДОПОМОГОЮ ПРОГРАМНОГО КОМПЛЕКСУ LABVIEW.....	16
2.1. Класифікація моделей в LabVIEW.....	16
2.2. Математична модель і математичне моделювання.....	18
2.3. Характеристика досліджуваного завдання.....	20
2.4. Поняття гомоморфізму і ізоморфізму в теорії моделювання.....	22
2.5. Імітаційне моделювання. Верифікація та валідація моделі.....	23
Висновки до розділу II.....	24
РОЗДІЛ III. ЗАСТОСУВАННЯ ПРОГРАМНОГО КОМПЛЕКСУ LABVIEW ДЛЯ ДОСЛІДЖЕННЯ ЧАСТОТНИХ ХАРАКТЕРИСТИК МЕХАНІЧНИХ СИСТЕМ.....	25
3.1. Частотні характеристики механічних систем.....	25
3.2. Дослідження частотних характеристик механічних систем за допомогою багатофункціонального вібровимірювального приладу “Vibroport”.....	27
3.2.1. Аналіз роботи багатофункціонального вібровимірювального приладу "VIBROPORT".....	27
3.2.3. Лабораторна установка для дослідження частотних характеристик механічних систем.....	32
3.3. Дослідження частотних характеристик механічних систем за допомогою програмного комплексу LabVIEW.....	35

3.4. Дослідження частотних характеристик механічної системи віртуальний приладом в палітрі Signal Processing – Transforms.....	41
Висновки до розділу III.....	45
ОСНОВНІ ВИСНОВКИ.....	46
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	47

ВСТУП

LabVIEW (англ. Laboratory Virtual Instrumentation Engineering Workbench) – це середовище графічного програмування, яке використовують технічні фахівці, інженери, викладачі та вчені по всьому світу для швидкого створення комплексних програм в задачах вимірювання, тестування, управління, автоматизації наукового експерименту.

В основі LabVIEW лежить концепція графічного програмування – послідовне з'єднання функціональних блоків на блок – діаграмі. LabVIEW містить великий список стандартних інструментів та готових рішень. До того ж, існують відповідні доповнення і модулі, які значно розширюють можливості середовища LabVIEW для обробки та аналізу сигналів.

Використовуваний в LabView графічна мова програмування G простий для освоєння людьми, що мають базові знання з програмування і не мають значного досвіду програмування. Наочне графічне представлення потоку даних в програмі і основних структурних елементів програми дозволяє відносно легко створювати і налагоджувати програми по управлінню приладом.

Поряд зі стандартними інструментами LabVIEW має додаткові модулі, які значною мірою розширюють можливості обробки та аналізу даних. Програма, написана в середовищі LabVIEW, називається віртуальним приладом (ВП). «Віртуальним» – тому, що її складові існують тільки в програмному коді і на екрані монітора. «Приладом» – тому, що користувач взаємодіє з нею, як з панеллю приладів, «намальованою» на екрані. Дані, з якими працює ВП є реальними, це не симуляція. LabVIEW містить повний набір інструментів для збору, аналізу, представлення і збереження даних.

Актуальність дослідження. LabVIEW є ідеальним програмним засобом для створення систем вимірювання, а також систем автоматизації управління на основі технології віртуальних приладів. LabVIEW – програма в комплексі з такими апаратними засобами, як багатоканальні вимірювальні аналого – цифрові перетворювачі, а також вимірювальні прилади, що підключаються до комп'ютера

через стандартні інтерфейси, дозволяє розробляти системи вимірювання, контролю, діагностики та управління практично будь – якої складності. LabVIEW незамінний для організації та проведення модельного експерименту, відмінність якого від реального полягає в тому, що в модельному експерименті можуть бути реалізовані будь – які ситуації, в тому числі «неможливі» та аварійні, що в силу різних причин буває неприпустимо при роботі з реальними об'єктами.

Більшість сучасних приладів оснащених комунікаційними інтерфейсами поставляються з драйверами, які є підпрограмами LabView, які можуть бути використані в створюваній програмі і забезпечують легке управління приладом на високому рівні. Крім того в LabView можливо встановлення управління приладом на низькому рівні, наприклад, за допомогою передачі команд, що відповідають стандарту SCPI.

Мета дослідження: створення у програмному комплексі LabVIEW, як середовищі візуального програмування, віртуального приладу для дослідження частотних характеристик реальних механічних систем.

Для досягнення даної мети були поставлені наступні задачі:

1. Вивчення можливостей середовища візуального програмування LabVIEW для моделювання реальних механічних систем і дослідження їх характеристик.
2. Відпрацювання прикладів побудови віртуальних приладів з використанням програмного комплексу LabVIEW для дослідження частотних характеристик.
3. Створення за допомогою програмного комплексу LabVIEW віртуального приладу для дослідження частотних характеристик реальних механічних систем.

Об'єкт дослідження: моделювання реальних механічних систем та отримання їх частотних характеристик.

Предмет дослідження: використання програмного комплексу LabVIEW для моделювання реальних механічних систем та отримання їх частотних характеристик.

Методи дослідження – комп'ютерне моделювання, засноване на візуальному графічному програмуванні засобами програмного комплексу LabVIEW.

Кваліфікаційна робота спеціаліста складається з вступу, 4 розділів, висновків та переліку посилань.

У першому розділі розглянуто можливості програмного комплексу LabVIEW. У другому розділі розглянуто принципи моделювання за допомогою комплексу LabVIEW. У третьому розділі розглянуто створений за допомогою програмного комплексу LabVIEW віртуальний прилад для моделювання та дослідження частотних характеристик реальних механічних систем. Четвертий розділ присвячено питанню охорони праці, у якому проаналізовано шкідливі та небезпечні фактори приміщення, заходи евакуації, прогнозування та оцінка можливої обстановки на об'єктах господарської діяльності

РОЗДІЛ I. ОСНОВИ ПРОГРАМУВАННЯ В ГРАФІЧНОМУ СЕРЕДОВИЩІ LABVIEW

1.1. Основні характеристики LabVIEW та перспективи розвитку.

LabVIEW (Laboratory Virtual Instrument Engineering Workbench – середовище розробки лабораторних віртуальних приладів) середовище розробки прикладних програм, створена фірмою National Instruments (США). У ній використовується інтуїтивно зрозумілу мову графічного програмування G. Його освоєння не вимагає знання традиційних текстових мов програмування. LabVIEW надає широкі можливості для проведення обчислень та математичного моделювання.

В LabVIEW програми створюються у вигляді графічних діаграм, подібних звичайним блок – схемами. Іноді можна створити додаток, взагалі не торкаючись клавіатури комп'ютера, на відміну від текстових мов, таких як C, Pascal та інші, де програми складаються у вигляді рядків тексту.

Компанія National Instruments була створена в 1976 році трьома засновниками – Джеффом Кодоскі (Jeff Kodosky), Джеймсом Тручардом (James Truchard) і Біллом Новлінім (Bill Nowlin) в американському місті Остін (Austin), штат Техас. Основною спеціалізацією компанії були інструментальні засоби для вимірювань і автоматизація виробництва.

Перша версія LabVIEW побачила світ через десять років після створення компанії – в 1986 році (це була версія для AppleMac). Інженери NI вирішили кинути виклик «традиційним» мов програмування і створили повністю графічне середовище розробки. Основним ідеологом графічного підходу став Джефф. Рік за роком випускалися нові версії.

Першою кроссплатформенною версією (включаючи Windows) була третя версія, випущена в 1993 році. В Остіні і по сьогоднішній день розташовується головний офіс компанії. Сьогодні в компанії працюють майже чотири тисячі осіб, а офіси знаходяться майже в сорока країнах.

В цьому відношенні середовище LabVIEW конкурентоспроможна з такими відомими системами комп'ютерної математики, як MATLAB, MathCAD, Mathematica, MAPLE допомогою якої ви можете створювати додатки, використовуючи графічне представлення всіх елементів алгоритму, що відрізняє її від звичайних мов програмування, таких як C, C++ або Java, де програмують, використовуючи текст. Однак LabVIEW являє собою значно більше, ніж просто алгоритмічний мову.

Це середовище розробки та виконання додатків, призначена для дослідників – учених і інженерів, для яких програмування є лише частиною роботи. LabVIEW функціонує на комп'ютерах, що працюють під управлінням всіх поширених операційних систем: Windows, MacOS, Linux, Solaris і HP – UX.

Останні версії LabVIEW орієнтовані на створення розподілених і дистанційних систем вимірювань. Це дозволяє забезпечити доступ на відстані до унікальним експериментальним стендів і організувати дистанційне навчання.

Можливості базового пакета можуть бути розширені за допомогою спеціалізованих модулів і функціональних бібліотек. Базовий пакет LabVIEW пройшов 20 – ти річний еволюційний шлях розвитку і оновлюється практично щороку. На момент початку роботи над даними посібником актуальною була версія LabVIEW 8.20.

Потужна графічний мова програмування LabVIEW дозволяє в сотні разів збільшити продуктивність праці. Створення закінченого додатка за допомогою звичайних мов програмування може відняти дуже багато часу – тижні або місяці, тоді як з LabVIEW потрібно лише кілька годин, оскільки пакет спеціально розроблений для програмування різних вимірів, аналізу даних і оформлення результатів. Так як LabVIEW має гнучкий графічний інтерфейс і простий для програмування, він також відмінно підходить для моделювання процесів, презентації ідей, створення додатків загального характеру і просто для навчання сучасного програмування.

Вимірювальна система, створена в LabVIEW, має велику гнучкість в порівнянні зі стандартним лабораторним приладом, тому що вона використовує різноманіття можливостей сучасного програмного забезпечення. І саме ви, а не виробник обладнання, визначаєте функціональність створюваного приладу. Ваш комп'ютер, обладнаний вбудованою вимірювально – керуючою апаратною частиною, і LabVIEW складають повністю настроюється віртуальний прилад для виконання поставлених завдань. За допомогою LabVIEW допустимо створити необхідний тип віртуального приладу при дуже малих витратах в порівнянні з звичайними інструментами. При необхідності ви можете внести до нього зміни буквально за хвилини.

LabVIEW створений для полегшення роботи з програмування ваших завдань. Для цієї мети є розширена бібліотека функцій і готових до використання підпрограм, які реалізують велику кількість типових завдань програмування і тим самим рятують нас від рутинної метушні з покажчиками, розподілом пам'яті та іншого шаманства, притаманне традиційним мовам програмування. У LabVIEW також містяться спеціальні бібліотеки віртуальних приладів для введення / виведення даних зі вбудовуваних апаратних засобів (data acquisition – DAQ), для роботи з каналом загального користування (КОП, General Purposes Interface Bus – GPIB), управління пристроями через послідовний порт RS – 232, програмні компоненти для аналізу, представлення і збереження даних, взаємодії через мережі Internet. Бібліотека аналізу (Analysis) містить безліч корисних функцій, включаючи генерування сигналу, його обробку, різні фільтри, вікна, статистичну обробку, регресійний аналіз, лінійну алгебру і арифметику масивів.

Завдяки своїй графічній природі LabVIEW – це пакет ефективного відображення і подання даних. Вихідні дані можуть бути показані в будь – якій формі, яку ви побажаєте. Діаграми, графіки стандартного виду, а також оригінальна користувацька графіка (user – defined graphics) складають лише малу частину можливих способів відображення вихідних даних.

Програми LabVIEW легко портативна на інші платформи: ви можете створити додаток на Macintosh, а потім запустити його в Windows, для більшості

додатків практично нічого не змінюючи в програмі. Ви побачите, що додатки, створені на LabVIEW, якісно покращують роботу в багатьох сферах діяльності людини – як в автоматизації технологічних процесів, так і в біології, сільському господарстві, психології, хімії, фізики, освіти і безлічі інших.

1.2. Мова графічного програмування і потоки даних.

LabVIEW – вельми високорівнева мова. Однак ніщо не заважає включати «низькорівневі» модулі в LabVIEW – програми. Навіть якщо ви хочете використовувати асемблерні вставки – це теж можливо, треба лише згенерувати DLL і вставити виклики в код. З іншого боку, високорівнева мова дозволяє запросто виробляти досить нетривіальні операції з даними, на які в звичайній мові могли піти багато рядки (якщо не десятки рядків) коду. Втім, заради справедливості треба зазначити, що деякі операції низькорівневих мов (наприклад, роботу з покажчиками), не так просто реалізувати в LabVIEW зважаючи на його «високорівневих».

Зрозуміло, мова LabVIEW включає основні конструкції управління, що мають аналоги і в «традиційних» мовах:

- змінні (локальні або глобальні)
- розгалуження (case structure)
- For – цикли з перевіркою завершення і без
- While – цикли
- Угрупування операцій.

Розробка додатків в середовищі LabVIEW відрізняється від роботи в середовищах на основі C або Java однією дуже важливою особливістю. Якщо в традиційних алгоритмічних мовах програмування заснована на вводі текстових команд, послідовно утворюють програмний код, в LabVIEW використовується мова графічного програмування, де алгоритм створюється в графічній іконній формі (pictorial form), що утворює так звану блок – діаграму (block – diagram), що дозволяє виключити безліч синтаксичних деталей.

Графічний підхід до програмування дозволяє виключити безліч синтаксичних деталей, яким багаті текстові мови програмування, наприклад крапки з комою або фігурні дужки. (Якщо Ви не знаєте, навіщо вони потрібні, вам це в LabVIEW і не знадобиться.)

Застосовуючи цей метод, ви можете сконцентрувати увагу лише на програмуванні потоку даних; спрощений синтаксис тепер не відволікає вас від аналізу самого алгоритму. На рисунку 1.1 показаний простий користувальницький інтерфейс LabVIEW і який реалізує його графічний код.

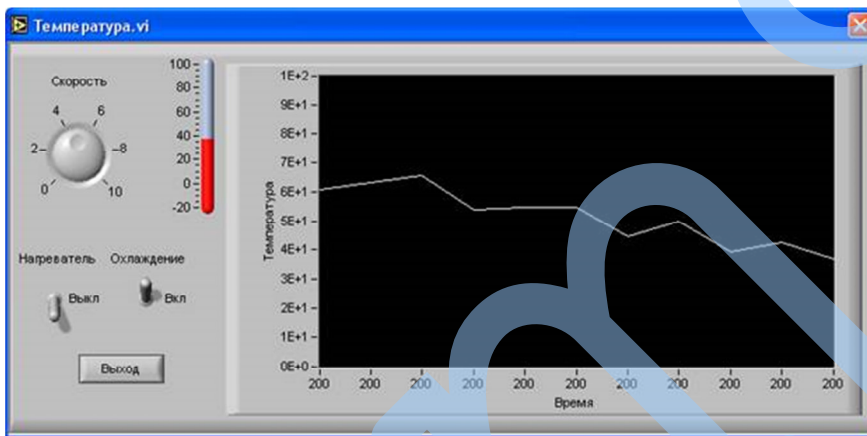


Рисунок 1.1 – Лицьова панель ВП

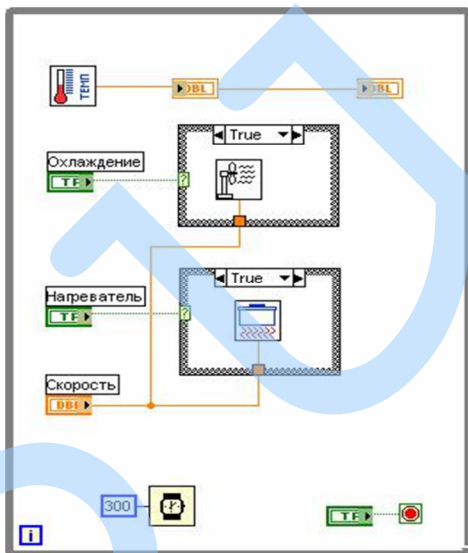


Рисунок 1.2 – Блок – діаграма ВП

В LabVIEW використовується термінологія, малюнки іконок і основні ідеї, знайомі вченим і інженерам. Ця мова базується на графічних символах, а не на тексті для опису програмованих дій. Основоположний для LabVIEW принцип потоку даних (dataflow), згідно з яким функції виконуються лише тоді, коли вони отримують на вхід необхідні дані, однозначно визначає порядок виконання алгоритму. Ви можете освоїти LabVIEW при невеликому або навіть відсутньому досвіді традиційного програмування, хоча знання його принципів було б дуже корисним.

У LabVIEW використовуються термінологія, малюнки іконок і основні ідеї, знайомі вченим і інженерам. Ця мова базується на графічних символах, а не на тексті для опису програмованих дій. Основоположний для LabVIEW принцип потоку даних (dataflow), згідно з яким функції виконуються лише тоді, коли вони отримують на вхід необхідні дані, однозначно визначає порядок виконання алгоритму. Ви можете освоїти LabVIEW при невеликому або навіть відсутній досвід традиційного програмування, хоча знання його принципів було б дуже корисним.

В основі LabVIEW лежить парадигма потоків даних. У вищенаведеному прикладі константа і термінал індикатора з'єднані між собою лінією. Ця лінія називається Wire. Можна назвати її «проводом». По проводам передаються дані від одних елементів іншим. Вся ця концепція називається Data Flow.

Суть Блок Діаграми – це вузли (ноди), виходи одних вузлів приєднані до входів інших вузлів. Вузол почне виконання тільки тоді, коли придуть всі необхідні для роботи дані. На діаграмі вгорі дві Ноди. Одна з них – константа. Цей вузол самодостатній – він починає виконання негайно. Другий вузол – індикатор. Він відобразить дані, які передає константа (але не відразу, а як тільки дані придуть від константи).

У LabVIEW VI виконуються під управлінням потоку даних, і вузол блок – діаграми виконується тільки тоді, коли отримає всі необхідні вхідні дані. По завершенні виконання вузол видає вихідні дані і передає їх наступного вузла на

шляху поширення потоку даних. Просування даних через вузли визначає порядок виконання VI і функцій на блок – діаграмі. Програми на Visual Basic, C++, JAVA і в більшості інших текстових мовах програмування виконуються відповідно до моделі управління потоком команд, згідно з якою послідовність програмних елементів визначає порядок виконання програми.

1.3. Специфіка роботи LabVIEW.

Програми LabVIEW називаються віртуальними приладами (ВП, virtual instruments – VI), оскільки вони функціонально і зовні подібні по реальним (традиційним) приладам. Однак вони настільки ж подібні програмам і функціям на популярних мовах програмування, таких як C або Basic. Тут і далі ми будемо називати програми LabVIEW віртуальними приладами або ВП, причому незалежно від того, співвідноситься їх вигляд і поведінку з реальними приладами чи ні.

Віртуальний прилад складається з трьох основних частин:

- Лицьова панель (Front Panel) являє собою інтерактивний користувальницький інтерфейс віртуального приладу і названа так тому, що імітує лицьову панель традиційного приладу. На ній можуть знаходитися ручки управління, кнопки, графічні індикатори та інші елементи керування (controls), які є засобами введення даних з боку користувача, а елементи індикації (indicators) – вихідні дані з програми. Користувач вводить дані, використовуючи мишу і клавіатуру, а потім бачить результати дії програми на екрані монітора;
- Блок – діаграма (Block Diagram) є вихідним програмним кодом ВП, створеним на мові графічного програмування LabVIEW, G (Джей). Блок – діаграма являє собою реально виконувану додаток. Компонентами блок – діаграми є: віртуальні прилади більш низького рівня, вбудовані функції LabVIEW, константи та структури управління виконанням програми. Для того щоб задати потік даних між певними об'єктами або, що теж саме, створити

зв'язок між ними, ви повинні намалювати відповідні провідники (wires). Об'єкти на лицьовій панелі представлені на блок – діаграмі у вигляді відповідних терміналів (terminals), через які дані можуть надходити від користувача в програму і назад;

– Для того щоб використовувати деякий ВП в якості підпрограми (подприлада) в блок – діаграмі іншого ВП, необхідно визначити його іконку (icon) і сполучну панель (connector). Віртуальний прилад, який застосовується всередині іншого ВП, називається, віртуальним подприладом (ВПП, SubVI), який аналогічний підпрограмі в традиційних алгоритмічних мовах. Іконка є однозначним графічним представленням ВП і може використовуватися в якості об'єкта на блок – діаграмі іншого ВП. Сполучна панель являє собою механізм передачі даних у ВП з іншої блок – діаграми, коли він застосовується в якості підприлада – ВПП. Подібно аргументів і параметрах підпрограми, сполучна панель визначає вхідні та вихідні дані віртуального приладу.

Віртуальні прилади є ієрархічними і модульними (modular). Ви можете використовувати їх як окремі програми (top – level programs), так і в якості віртуальних підприладів. Згідно з цією логікою, LabVIEW слід концепції модульного програмування (modular programming). Спочатку ви поділяєте велику прикладну задачу на ряд простих підзадач. Далі створюєте віртуальні прилади для виконання кожної з підзадач, а потім поєднуєте ці ВП на блок – діаграмі приладу більш високого рівня, який виконує прикладну задачу в цілому. Технологія модульного програмування дуже хороша, тому що ви можете працювати з кожним ВПП окремо, що полегшує налагодження програми. Більше того, ЗПС низького рівня часто виконують завдання, типові для декількох додатків і тому можуть використовуватися незалежно у багатьох окремих додатках.

У таблиці 1.1 наведено ряд основних термінів LabVIEW та їх загальноприйнятих еквівалентів для традиційних мов програмування.

Таблиця 1.1 – Терміни LabVIEW та їх еквіваленти для традиційних мов програмування

LabVIEW	Традиційні мови програмування
Віртуальний прилад (ВП)	Програма
Функція	Функція або метод
Віртуальний підприлад (ВПП)	Підпрограма, об'єкт
Лицьова панель	Інтерфейс користувача
Блок – діаграма	Програмний код
G або LabVIEW	C, C ++, Java, Basic і ін

Для створення коду віртуального приладу використовується палітра компонентів, що містить стандартний набір готових ВП, що дозволяють реалізовувати складні алгоритми. Панель інструментів показана на рисунку 1.3.



Рисунок 1.3 – Панель інструментів для створення блок – діаграми



Рисунок 1.4 – Панель інструментів для створення лицьової панелі

Для створення лицьовій панелі використовується інша панель інструментів, яка містить набір цифрових індикаторів і датчиків, кнопок, лампочок, рукояток, графічних індикаторів і т. д. Ця панель інструментів відображено на рисунку 1.4.

Висновки до розділу I:

LabVIEW є потужним і гнучким програмним пакетом для отримання, обробки та аналізу даних. Для створення програм, званих віртуальними приладами (ВП), в LabVIEW застосовується мова графічного програмування. Користувач взаємодіє з програмою через лицьову панель. Кожній лицьовій панелі відповідає блок – діаграма, яка є вихідним кодом віртуального приладу. LabVIEW має багато вбудованих функцій для полегшення процесу програмування; компоненти зв'язуються між собою провідниками, визначальними шляху потоку даних в межах блок – діаграми.

Програми LabVIEW легко портативна на інші платформи: ви можете створити додаток на Macintosh, а потім запустити його в Windows, для більшості додатків практично нічого НЕ змінюючи в програмі. Ви побачите, що додатки, створені на LabVIEW, якісно покращують роботу у багатьох сферах діяльності людини – як в автоматизації технологічних процесів, так і в біології, сільському господарстві, психології, хімії, фізики, освіти і безлічі інших.

Стандартний набір інструментів 'Signal Analysis Express VIs' дозволяє проводити аналіз, обробку, перетворення сигналів, а також імітувати різні типи частотних форм.

Для розширення можливостей середовища LabVIEW з обробки сигналів служать зовнішні модулі, такі як 'Advanced signal processing toolkit' й 'Sound and Vibration toolkit'. Дані доповнення дозволяють виробляти більш детальний і глибокий аналіз сигналів, включаючи метод спектра обвідної вібросигналу, реалізація якого є головним завданням конструкційного додатка

Визначено завдання та функції розроблюваної програми в середовищі LabVIEW. Додаток повинен відкривати і зчитувати звуковий файл, виводити амплітудну та частотну характеристику на графік.

РОЗДІЛ II. МОДЕЛЮВАННЯ МЕХАНІЧНИХ СИСТЕМ ЗА ДОПОМОГОЮ ПРОГРАМНОГО КОМПЛЕКСУ LABVIEW

2.1. Класифікації моделей в LabVIEW.

Модель являє собою абстрактне опис системи (об'єкта, процесу, проблеми, поняття) в деякій формі, відмінній від форми їх реального існування.

Моделювання являє собою один з основних методів пізнання, є формою відображення дійсності і полягає в з'ясуванні або відтворення тих чи інших властивостей реальних об'єктів, предметів і явищ за допомогою інших об'єктів, процесів, явищ, або за допомогою абстрактного опису у вигляді зображення, плану, карти, сукупності рівнянь, алгоритмів і програм.

В процесі моделювання завжди існує оригінал (об'єкт) і модель, яка відтворює (моделює, описує, імітує) деякі риси об'єкта.

Моделювання засноване на наявності в різноманітності природних і штучних систем, що відрізняються як цільовим призначенням, так і фізичним втіленням, подібності або подоби деяких властивостей: геометричних, структурних, функціональних, поведінкових. Ця схожість може бути повним (ізоморфізм) і частковим (гомоморфізм)[3].

Загальноприйнятої єдиної класифікації моделей поки ще немає, як немає і загальноприйнятого визначення моделі:

Статистичні моделі – моделі, які описують стан моделі в певний момент часу. У фізиці прикладом статистичної моделі може бути рух молекул, в хімії – моделі будови атомів, в біології – моделі класифікації тварин.

Динамічні моделі – моделі, що описують процеси зміни і розвитку системи з плином часу. У фізиці прикладом динамічної моделі може бути рух тіл, в хімії – процеси походження хімічних реакцій, в біології – розвиток організмів.

Матеріальні – відтворюють геометричні, фізичні та ін. Властивості об'єктів в матеріальній формі. Вони засновані на чомусь існуючому незалежно

від людської свідомості. Фізичні моделі: авто – і авіамоделі, глобус, будівлі і т. Д.

Аналогові моделі засновані на процесах, аналогічних досліджуваним процесам. Наприклад, електричний ланцюг як аналогія хімічних, соціальних мереж.

Фізичною моделлю процесу або явища називається його математична модель, складена з ідеальних фізичних об'єктів.

Інформаційна модель – опис реального об'єкта, процесу або явища на одній з мов (розмовному або формальному).

Образні інформаційні моделі – це зорові образи (малюнки, фотографії), зафіксовані на будь – якому носії інформації (папері, плівці та іншому). Вони широко використовуються в освіті і науках, де потрібно класифікація об'єктів за їх зовнішніми ознаками.

Знакові інформаційні моделі будуються з використанням різних мов. Знакова інформація може бути представлена у вигляді тексту, формул, таблиць і т. Д.

Геометричні моделі – моделі, представлені у вигляді об'ємних конструкцій або графічних форм.

Словесні моделі – моделі, виражені усно або письмово.

Математичні моделі – моделі, записані за допомогою рівнянь, математичних формул, нерівностей, систем і т. П.

Структурні моделі – схеми, графіки, таблиці і т. П.

Логічні моделі – моделі, в яких приймаються рішення на основі аналізу різних умов.

Спеціальні моделі – ноти, хімічні формули і т. П.

Комп'ютерні моделі – моделі, реалізовані на комп'ютері за допомогою програмного забезпечення.

Формалізація – процес побудови інформаційної моделі. Модель – це якийсь новий об'єкт, який відображає деякі істотні сторони досліджуваного об'єкта, явища або процесу.

Модель використовується як умовний образ, сконструйований для спрощення їх дослідження.

Природа моделей може бути різною: матеріальні моделі (наприклад, модель літака в аеродинамічній трубі), знакові моделі трьох типів: спеціальні (ноти, хімічні формули), математичні (наприклад, формула, що описує гравітаційну взаємодію двох тіл), алгоритмічні (програмний код комп'ютерного додатку); словесний опис об'єкта (явища, процесу) можна також розглядати як його модель.

Використовують види класифікацій моделей, що визначаються наступними ознаками:

- 1) областю використання;
- 2) обліком в моделі тимчасового фактора (динаміки);
- 3) галуззю знань;
- 4) способом представлення моделей.

По області використання моделі можуть бути наступними: навчальні, дослідні, науково – технічні, ігрові, імітаційні.

Якщо враховувати часовий фактор (динаміку процесу), то моделі можуть бути статичні і динамічні. Приклади класифікації моделей за галузями знань: математика, фізика, хімія, мовознавство і т.д.

Більш докладно розглянемо класифікацію за способами представлення моделей. У цьому випадку всі моделі діляться на матеріальні (предметні або фізичні) і на абстрактні (нематеріальні). У свою чергу абстрактні моделі складаються з

- 1) вербальних і образних;
- 2) інформаційних.

Інформаційні моделі включають в себе дві великі групи моделей:

- 1) образно – знакові;
- 2) знакові.

У свою чергу образно – знакові діляться на геометричні, структурні, словесні, алгоритмічні. Приклади геометричних моделей: креслення, план, карта.

Представниками структурних моделей є таблиці, графи, діаграми. Природні мови – найбільш типові приклади словесних моделей. Характерні примірники алгоритмічних моделей: нумеровані списки, блок – схеми, покрокове перерахування.

Знакові моделі складаються з математичних, спеціальних, алгоритмічних. Приклади спеціальних моделей: ноти, хімічні формули. Програма на алгоритмічній мові є представником алгоритмічних моделей.

Наведемо одне з обґрунтувань введення моделей [4]. Отже, модель потрібна:

- для того щоб зрозуміти, як влаштований конкретний об'єкт, яка його структура, основні властивості, закони розвитку і взаємодії з навколишнім світом;
- для того щоб навчитися управляти об'єктом (або процесом) і визначити найкращі способи управління при заданих цілях і критеріях;
- для того щоб прогнозувати прямі і непрямі наслідки реалізації заданих способів і форм впливу на об'єкт.

Характерні приклади обґрунтування введення моделей – ядерна реакція, управління виробничими процесами за допомогою технології віртуальних приладів LabVIEW.

За властивостями моделі виявляється можливим судити про властивості досліджуваного об'єкта – однак не про всіх, а лише про ті, які аналогічні і в моделі, і в об'єкті, і при цьому важливі для дослідження (такі властивості називаються істотними).

2.2. Математична модель і математичне моделювання.

Математична модель – наближений опис якого – небудь класу явищ зовнішнього світу, виражене за допомогою математичної символіки [5].

Важливо так «сконструювати» наближену математичну модель, щоб вона досить точно відображала характерні властивості даного явища. При цьому

можуть бути опущені несуттєві і другорядні властивості явища з тим, щоб ця модель була доступна для дослідження на даному рівні розвитку обчислювальної техніки.

Математичне моделювання – вивчення явища за допомогою математичної моделі [5].

Класичним прикладом математичного моделювання є опис та дослідження основних законів механіки І. Ньютона засобами математики.

З точки зору розвитку знань людства про природознавстві і дослідження законів природи, техніки, суспільства і науки на основі математичних моделей процес вирішення конкретної проблеми можна розбити на наступні етапи [5]:

- Формулювання законів, що зв'язують основні об'єкти моделі.
- Дослідження математичних задач, до яких призводять математичні моделі.
- Перевірка: чи задовольняє схвалена (гіпотетична) модель критерієм практики.
- Подальший аналіз моделі у зв'язку з накопиченням даних про досліджуваних явищах і модернізація моделі.

Перший етап вимагає широкого знання фактів, що відносяться до досліджуваних явищ, і глибокого проникнення в їх взаємозв'язку. Ця стадія завершується записом в математичних термінах сформульованих якісних уявлень про зв'язки між об'єктами моделі.

На другому етапі важливу роль набувають математичний апарат, необхідний для аналізу математичної моделі, і обчислювальна техніка – потужний засіб для отримання кількісної вихідної інформації як результату вирішення складних математичних завдань. Часто математичні завдання, що виникають на основі різних математичних моделей явищ бувають однаковими. Наведемо приклад: основне завдання лінійного програмування відображає ситуації різної природи. Це дає обґрунтування вважати ці різні математичні моделі гомоморфності по відношенню до математичних задачах, до яких ці

моделі наводяться. Такі типові математичні задачі досліджуються вченими та інженерами як самостійний об'єкт, абстрагуючись від досліджуваних явищ.

На третьому етапі перевіряється, чи узгоджуються результати спостережень з теоретичними наслідками моделі в межах точності спостережень. Застосування критерію практики до оцінки математичної моделі дозволяє робити висновок про правильність положень, що лежать в основі підлягає вивченню (гіпотетичної) моделі. Цей метод є єдиним методом вивчення недоступних нам явищ макро – і мікросвіту.

Перейдемо до четвертого етапу. У процесі розвитку науки і техніки дані про досліджуваних явищах все більш і більш уточнюються і настає момент, коли висновки, одержувані на підставі прийнятої математичної моделі, не відповідають нашим знанням про явище. Таким чином, виникає необхідність побудови нової, більш досконалої математичної моделі.

Все вищесказане можна застосувати і до комп'ютерних технологій. Сучасні підприємства не зможуть вижити у всесвітній конкуренції, якщо не будуть випускати нові продукти кращої якості, більш низької вартості і за менший час [2]. Тому вони прагнуть використовувати величезні можливості пам'яті комп'ютерів, їх високу швидкодію і можливості зручного графічного інтерфейсу для того, щоб автоматизувати і зв'язати один з одним завдання проектування і виробництва, які раніше були вельми нудними і абсолютно не пов'язаними один з одним. Таким чином скорочується час і вартість розробки і випуску продукту. У зв'язку з цим постійно виходять нові версії програмних продуктів. З точки зору етапів математичного моделювання відбувається постійний перехід від четвертого етапу до першого, але вже йде дослідження заданої задачі по удосконаленій моделі програмного продукту. Наочний приклад. Зовсім недавно у LabVIEW була сьома версія, протягом півтора років з'явилися по черзі версії 8, 8.20, 8.50.

2.3. Характеристика досліджуваного завдання.

В ході виконання програми може бути створений масив даних, що містить всі випадкові числа, створені генератором випадкових чисел. можна створити масив даних, що містить числа, створювані лічильником ітерацій циклу.

Це будуть одномірні масиви. Можна створити масив в якому кожному номеру ітерації циклу буде відповідати випадкове число, сгенероване в ході цієї ітерації. В цьому випадку отримується масив буде двовимірним, тобто матрицею. У цій матриці буде два рядки: один рядок відповідає числам, створеним генератором випадкових чисел, другий рядок відповідає числам, створеним лічильником ітерацій. Число стовпців в матриці визначається числом ітерацій.

Впритул підійдемо до визначення похибки розв'язання досліджуваної задачі. Похибка рішення обумовлюється наступними причинами.

- Математичний опис задачі є неточним, зокрема, неточно задані вихідні дані.
- Вживаний для вирішення метод не є точним. Рішення більшості завдань природознавства на сучасному етапі при побудові їх математичних моделей важко отримати в кінцевому вигляді, а часто просто неможливо. Тому доводиться вдаватися до наближених методів таким як, чисельні, асимптотичні.
- При введенні даних в обчислювальну машину, при виконанні арифметичних операцій і при виведенні даних проводяться округлення.

Похибки, відповідні цих причин, називають:

- 1) непереборний похибкою;

- 2) похибкою методу;
- 3) обчислювальної похибкою.

На підставі цього визначимо повну похибка досліджуваного завдання природознавства. Отже, нехай Φ – точне рішення деякої задачі;

Φ_1 – рішення задачі, отримане за рахунок переходу до математичної моделі. За рахунок переходу до такої моделі утворюється невіправна похибка: $\varepsilon_1 = \Phi - \Phi_1$.

Нехай Φ_2 – рішення задачі Φ_1 , отримане за рахунок вибору методу рішення. Введемо похибка методу $\varepsilon_2 = \Phi_1 - \Phi_2$.

Нехай Φ_3 – рішення задачі Φ_2 , отримане при реалізації конкретного методу на конкретній ПЕОМ. На цьому етапі утворюється обчислювальна похибка $\varepsilon_3 = \Phi_2 - \Phi_3$. Знайдемо повну похибка:

$$\varepsilon = \Phi - \Phi_3 = (\Phi - \Phi_1) + (\Phi_1 - \Phi_2) + (\Phi_2 - \Phi_3). \quad (2.1)$$

У багатьох випадках під терміном похибка того чи іншого виду зручно розуміти не розглянуті вище різниці між наближеннями, а деякі міри близькості між ними. Наприклад, в скалярному випадку вважають

$$\varepsilon_1 = |\Phi - \Phi_1|, \varepsilon_2 = |\Phi_1 - \Phi_2|, \varepsilon_3 = |\Phi_2 - \Phi_3|, \varepsilon = |\Phi - \Phi_3|. \quad (2.2)$$

За таких позначеннях отримуємо оцінку:

$$\varepsilon \leq \varepsilon_1 + \varepsilon_2 + \varepsilon_3. \quad (2.3)$$

Зупинимося на першій похибки, непереборний. Більшість дослідників – інженерів з нею зазвичай миряться, вважають, що з нею нічого не можна вдіяти. Взагалі – то кажучи, це правильний підхід, поки ми знаходимося на другому і третьому етапах математичного моделювання. Але з плином часу гіпотетична модель перестане задовольняти критерієм практики. Відбувається перехід до вдосконаленої завдання, до задачі, яка ближче до реальної, і відповідно з

меншою непереборний похибкою. Цей перехід відбувається більш безболісно в середовищі віртуальних приладів LabVIEW.

Оцінку похибки, яку можна отримати до проведення обчислень, зазвичай називають апріорної, після проведення – апостеріорної. Отримати апріорну оцінку похибки для скільки змістовних обчислень – вкрай важке завдання.

2.4. Поняття гомоморфізму і ізоморфізму в теорії моделювання.

Поняття гомоморфізма, що є основним інструментом для практично точного опису різних процедур, що об'єднуються загальним терміном «моделювання», допускає ряд природних узагальнень, цікавих насамперед з огляду на велику різноманітність і гнучкості їх логіко – методологічних інтерпретацій. Якщо досліджувана система, як це часто буває, задана частково (в т.ч. частково визначені на ній операції і предикати), то можна доопределити систему – як варіант, ввівши якусь метрику і (або) топологію (т.зв. топологічна інтерпретація неповних ізоморфізмів).

Ізоморфне перетворення – процес формалізації вже побудованої таким чином концептуальної схеми у вигляді наукової теорії (І створюваних на її основі прикладних технологій і інструментів).

Якщо професійні математики в силу специфіки роду занять стосуються різною мірою понять ізоморфізму і гомоморфізму, то фахівці з імітаційного моделювання по ряду причин намагаються цього уникнути. Так, класична в усіх відношеннях монографія [5] В. Кельтона і А. Лоу не містить ніяких відомостей з цього питання, вкрай скупі сказано на цю тему і у Ю.І. Рижикова в [12].

Однак уже В.Н. Томашевський недвозначно фіксує роль ізоморфізму і гомоморфізма як заходи відповідності між вихідною системою і моделлю: система і модель ізоморфні, якщо існує взаємнооднозначне відповідність між ними, завдяки якому можна перетворити одне подання в інше; система і модель ізоморфні тільки в разі спрощення системи, тобто скорочення при моделюванні безлічі її властивостей і характеристик поведінки, що впливають на простір

станів системи; прикладом менш тісних зв'язків між системою і моделлю є гомоморфні зв'язку, які визначають однозначна відповідність тільки в одну сторону (від моделі до системи) [14, с. 21 – 22].

Стандартним чином введемо поняття гомоморфізму: кожному елементу і кожному відношенню між елементами першої системи відповідає один елемент і одне відношення другої системи (але не навпаки) [6].

Подібність моделі з оригіналом завжди неповна. Модель лише наближено відображає деякі властивості оригіналу. Причому реальна система може мати різні гомоморфні їй моделі.

Наведемо приклади. Тексту на екрані комп'ютера, збереженому у форматі Word з включеним режимом «попередній перегляд», відповідає кілька гомоморфних моделей в тому ж самому Word'і з включеною опцією «недруковані знаки». Другий приклад: креслення будинку є гомоморфною моделлю по відношенню до самого будинку (креслення зображене на площині, а будинок – об'ємний, тривимірний; креслення дає не всі деталі, припустимо, що на ньому не видно окремих цеглин і т.д.). Третій приклад: у LabVIEW одному терміналу, відповідальному за введення числових даних на Block Diagram відповідає кілька конструкцій введення на Front Panel, таких як Knob, Dial, Num Ctrl і т.д.. Четвертий приклад: у LabVIEW одному програмному коду в Block Diagram може відповідати кілька зображень на лицьовій панелі (Front Panel).

Гомоморфізм є фундаментальним теоретичним обґрунтуванням моделювання.

Суть поняття ізоморфізму: між елементами ізоморфних об'єктів існує взаємноодзначне співставлення, тобто кожному елементу (і відношенню між ними) одного об'єкта точно відповідає один елемент (і ставлення) іншого об'єкта і навпаки.

Останнє, зокрема, означає, що при ізоморфізмі одна система може бути моделлю іншої. У свою чергу, остання на всіх підставах може бути прийнята в якості моделі першої.

Модель ізоморфна по відношенню до деякого абстрактного образу, поданням про об'єкт, яке в свою чергу є його гомоморфним відображенням.

Розглянемо приклади. Текст, набраний в Word'i, з включеною опцією «попередній перегляд», може розглядатися як ізоморфна модель тексту, виведеного на принтер. У другому прикладі креслення будинку на дошці в аудиторії і те ж саме креслення в студентському конспекті можуть виступати як ізоморфні моделі по відношенню один до одного.

2.5. Імітаційне моделювання. Верифікація та валідація моделі.

Верифікація моделі – перевірка її істинності, адекватності [6]. Верифікація моделі при машинному (комп'ютерній) імітації є перевірка відповідності її поведінки припущеннями експериментатора. Коли модель організована в обчислювальну програму для ПЕОМ, то спочатку, як зазвичай, виправляють помилки в її запису на алгоритмічній мові, а потім переходять до верифікації. Процес налагодження програмного коду в LabVIEW часто відбувається швидше, ніж у відомих вже канонічних програмних середовищах, таких як Visual Basic, Delphi. Це пов'язано з тим, що середовище LabVIEW відразу пропонує можливі способи усунення помилок. Це перший етап дійсної підготовки до імітаційного експерименту. Підбираються деякі вихідні дані, для яких можуть бути передбачені результати розрахунку. Якщо виявиться, що ПЕОМ видає дані, що суперечать тим, які очікувалися при формуванні моделі, значить, модель невірна. У зворотному випадку переходять до наступного етапу перевірки працездатності моделі – її валідації.

Валідація моделі – перевірка відповідності даних, одержуваних у процесі машинної імітації, реальному ходу явищ, для опису яких створено модель. Виробляється тоді, коли експериментатор переконався на попередній стадії (верифікації) у правильності структури (логіки) моделі. Валідація полягає в тому, що вихідні дані після розрахунку на обчислювальній техніці зіставляються з наявними відомостями про моделюється системі. Процес

валідації пов'язаний з підтримкою роботи програмного продукту його виробника в промислових умовах. Тут саме середовище LabVIEW ратує за процес валідації.

Висновки до розділу II:

В даному розділі було розглянуто моделювання систем за допомогою програмного комплексу LabVIEW.

Розглянуто приклади застосування сучасних пакетів при математичному моделюванні пристроїв і систем. Сучасні програмні засоби дозволяють значно підвищити ефективність роботи розробника вже на початкових стадіях роботи над проектом.

Формалізація процедури гомоморфного відображення (і, відповідно, підтвердження її коректності) – питання більш ніж відкритий. У свою чергу, завдання повної автоматизації (комп'ютеризації) процедури формалізації не менше далека від вирішення в комерційних системах імітаційного моделювання (справді, не можна ж вважати, що застосування розрахованих на скорочення термінів розробки засобів візуального моделювання в змозі замінити логічно несуперечливі дії, які просувають дослідника від концептуальної моделі до програмного коду, придатного для вирішення реальних завдань). В Як один з варіантів вирішення проблеми пропонується сконцентрувати зусилля на створенні гібридних («інтелектуальних») імітаційних моделей, що використовують не тільки стали вже звичними методи системного аналізу, теорії масового обслуговування або реінжинірингу бізнес – процесів – але також спираються на логікоалгебраїчні схеми (в т.ч. апарат неklasичної логіки і алгебраїчні структури).

РОЗДІЛ III. ДОСЛІДЖЕННЯ ЧАСТОТНИХ ХАРАКТЕРИСТИК МЕХАНІЧНИХ СИСТЕМ ЗАСОБАМИ ПРОГРАМНОГО КОМПЛЕКСУ LABVIEW

3.1 Частотні характеристики механічних систем.

Частотні характеристики об'єкта визначаються його комплексним коефіцієнтом передачі $W(j\omega) = W(p) | p = j\omega$, який є Фур'є перетворенням ІХ. У радіотехніці і електроніці замість коефіцієнта передачі користуються поняттям комплексного коефіцієнта посилення $K(j\omega) = (\omega)$ [4].

Модуль комплексного коефіцієнта передачі $|W(j\omega)| = A(\omega)$ являє собою, як відомо, амплітудно частотну характеристику (АЧХ) об'єкта з функцією передачі $W(p)$, а аргумент $\arg(W(j\omega)) = \varphi(\omega)$ – фазо частотну характеристику (ФЧХ). В електроніці АЧХ і ФЧХ є найважливішими параметрами лінійних підсилювачів, виконаних на електронних лампах, транзисторах або інтегральних мікросхемах.

Графічне представлення $W(j\omega)$ на комплексній площині при зміні частоти ω від 0 до ∞ , тобто графік амплітудно фазової характеристики (АФХ) в полярних координатах, у вітчизняній літературі називається годографом, а в англомовній – діаграмою Найквіста. Існують і більш інформативні діаграми, наприклад діаграма Ніколса.

В теорії управління, так і в радіоелектроніці, часто використовується логарифмічна амплітудно частотна характеристика (ЛАЧХ), що визначається виразом $20 \lg |W(j\omega)|$. При цьому вісь частот також будується в логарифмічному масштабі.

Розглянемо механічну систему, що представляє собою тіло маси, поєднане лінійно пружно – демпфуючим зв'язком з жорсткістю і коефіцієнтом демпфування з нерухомою основою і має одну ступінь свободи (рисунк 1).

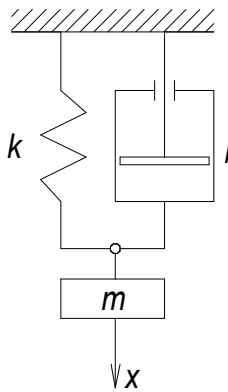


Рисунок 3.1 – Система з одним ступенем свободи

Докладемо до системи періодичне гармонійне вплив з частотою у вигляді сили, якщо мати на увазі, що дія постійної сили тяжіння викликає лише статичне зміщення, що не представляє для нас інтересу, рівняння руху системи можна записати у вигляді:

$$\ddot{x} + 2n\dot{x} + \omega_0^2 x = q_0 \cos(\omega t) \quad (3.1)$$

де:

$$\begin{cases} 2n = \frac{b}{m}; \\ \omega_0^2 = \frac{k}{m}; \\ q_0 = \frac{F_0}{m}. \end{cases} \quad (3.2)$$

Загальний розв'язок рівняння (3.1) має вигляд:

$$x = Ce^{-\lambda t} \cos(\omega_0 t - \phi_0) + A \cos(\omega t + \phi) \quad (3.3)$$

де перший член описує вільні затухаючі коливання з власною частотою системи, а другий – вимушені, з частотою змушуючої сили. З огляду на те, що власні

коливання загасають, перший член в загальному розв'язку далі враховувати не будемо.

З цих рівнянь визначаються амплітуда вимушених коливань та їх фазове запізнювання по відношенню до змушуючого впливу:

$$\begin{cases} A = \frac{q_0}{\sqrt{(\omega_0^2 - \omega^2)^2 + (2n\omega)^2}}; \\ \phi = \arctg \frac{2n\omega}{(\omega_0^2 - \omega^2)}. \end{cases} \quad (3.4)$$

При відхиленні системи дорівнює статичному:

$$A_c = \frac{q_0}{\omega_0} \quad (3.5)$$

Для безрозмірної амплітуди можна побудувати її функціональну залежність від частоти зовнішнього збурення, віднесеної до власної частоти системи :

$$a = \frac{1}{\sqrt{(1-p^2)^2 + \left(\frac{2np}{\omega}\right)^2}} \quad (3.6)$$

Вираз (3.6) називається амплітудною частотною характеристикою системи (АЧХ). При частотних збудженнях, близьких до частоти власних коливань системи, спостерігається локальне підвищення амплітуди коливань. При цьому, згідно (3.6):

$$A = A_c \frac{\omega}{2n} \quad (3.7)$$

Аналогічну функціональну залежність від частоти зовнішнього збурення можна побудувати для фази вимушених коливань ϕ , тобто для їх фазового

Отформатировано: русский

Отформатировано: Обычный (веб)

запізнювання по відношенню до впливу, що збудує – фазову частотну характеристику (ФЧХ).

3.2. Дослідження частотних характеристик механічних систем за допомогою багатофункціонального вібровимірювального приладу “Vibroport”.

3.2.1 Багатофункціональний вібровимірювальний прилад "VIBROPORT"

Протягом більше 20 років використовується багатофункціональний вібровимірювальний прилад "Vibroport" фірми «Schenck» (Німеччина) для різних вібродосліджень, вимірювань параметрів механічних коливань і, в тому числі, отримання амплітудних і фазових частотних характеристик механічних систем в лабораторії кафедри ЗМ і ДМ.

Прилад «Vibroport» зручний, легкий, універсальний, портативний прилад, який дозволяє вимірювати і аналізувати механічні коливання машин в зборі, а також досліджувати коливання з тим, щоб усунути їх. Зокрема, «Vibroport» дозволяє вирішувати наступні задачі:

- вимірювання інтенсивності коливань машин;
- безконтактне вимірювання коливань роторів;
- безконтактне вимірювання частоти обертання роторів;
- статичне і динамічне балансування роторів машин в експлуатаційних умовах;
- частотний аналіз коливань машин;
- гармонійний аналіз спектру коливань;
- запис кривих розгону і вибігу машин ("діаграми Найквіста") для визначення комплексних передавальних функцій до, власних частот, динамічної жорсткості, коефіцієнта демпфірування, власних форм коливань механічних систем і т. п.

Безконтактні датчики віброзміщення або електродинамічні датчики віброшвидкості можуть використовуватися у комплекті з приладом. В

останньому випадку вбудований інтегратор дозволяє отримувати при вимірюванні також значення вібропереміщення. Прилад дозволяє вимірювати значення віброшвидкості або вібропереміщення в межах 0, 0 1 200 мм / с і 1 2000 мкм відповідно. Лицьова панель приладу показана на рисунку 3.2.

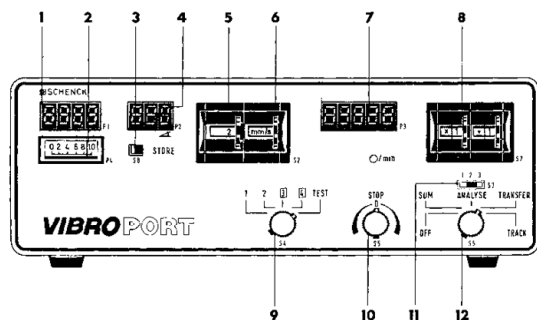


Рисунок 3.2 – Лицьова панель приладу «Vibroport»:

1 – цифровий індикатор вимірюваної величини; 2 – аналоговий індикатор тенденції зміни вимірюваної величини; 3 – запам'ятовування виміряного значення; 4 – показання фазового кута вимірюваної величини; 5 – атенюатор вхідного сигналу (вибір діапазону вимірювання); 6 – перемикач розмірності вимірюваної величини; 7 – індикатор частоти обертання ротора; 8 – перемикач кратності виділяються гармонік оборотної частоти; 9 – перемикач вимірювальних каналів; 10 – перемикач швидкості і напряму перебудови частоти фільтру і зміни частоти генератора, що задає; 11 – перемикач ширини смуги пропускання фільтру; 12 – перемикач виду роботи і вимикач живлення

Для вимірювання частоти обертання ротора може використовуватись або безконтактний оптоелектронний датчик, для роботи якого приклеюється спеціальна мітка на роторі машини з світловідзеркалючого паперу, або безконтактний вихрострумний датчик, для роботи якого необхідна наявність на роторі машини яких – небудь отворів або виступів, наприклад, шпонок або шпонкових пазів. Межа вимірювань становить до 100 000 об / хв з точністю 1 об / хв.

Частотний діапазон приладу складає 1 ... 1666 Гц. Він оснащений цифровим фільтром. Ширина смуги пропускання фільтра постійна за частотою і має 3 значення: 0.2, 1 .0 та 5 .0 Гц.

Для автоматичної реєстрації результатів вимірювань, зокрема, частотного аналізу, спільно з приладом може використовуватися двохкоординатний самописець. Для цих цілей «Vibroport» забезпечений відповідними виходами постійної напруги, пропорційної «сумарному» рівню коливань (середнє квадратичне значення – СКЗ), рівню обраної частотної складової, частоті настроювання фільтра або частоті обертання ротора, величині фазового кута, а також виходами змінної напруги, відповідного фільтрованого і нефільтрованого вимірювального сигналу. Крім того, є вихід опорного синусоїдального сигналу від внутрішнього задаючого генератора з частотою, рівною частоті налаштування фільтра.

У режимах вимірювання інтенсивності коливань машин і безконтактного вимірювання коливань роторів (режим «Sum») прилад працює як звичайний віброметр, а в режимі бесконтактного вимірювання частоти обертання роторів (режим «Track») – як електронний тахометр.

При частотному аналізі коливань машин (режим «Analyse») використовується перестроюваний цифровий фільтр, частота настройки якого може вибиратися як вручну, так і задаватися автоматично з певною швидкістю, що зручно при реєстрації результатів частотного аналізу на двокоординатному самописці. На вхід «Х» самописця в цьому випадку подається з відповідного виходу приладу сигнал, пропорційний частоті налаштування фільтра, а на вхід «У» – сигнал, пропорційний рівню відслідковуваної частотної складової.

Слід зазначити, що швидкість переналаштування фільтра, яка задається перемикачем 10, повинна відповідати ширині його смуги пропускання, встановленої перемикачем 11: швидкість 1 (0.013 Гц / с) – ширина смуги 1 (0,2 Гц), швидкість 2 (0.2 Гц / с) – ширина смуги 2 (1 Гц), швидкість 3 (3.5 Гц / с) – ширина смуги 3 (5 Гц).

Оскільки вібрації машини мають прямий зв'язок зі швидкістю обертання її ротора, важливою функціональною можливістю приладу є здійснення гармонійного аналізу спектра коливань, тобто виявлення складових з частотами, кратними обертовій частоті (частоті обертання ротора) – від 9 – ї субгармоніки до 9 – ї гармоніки включно.

Для реалізації даної можливості в режимі «Track» перемикача 8 кратності гармонік устанавлюється відповідний коефіцієнт (від 1/9 до 9/1), який і визначає необхідні конфігурації фільтра приладу по відношенню до частоти обертання ротора машини, що фіксується за допомогою безконтактного оптоелектронного або вихорострумовеого датчика. Важливо, що при цьому реєструється також фазовий кут кожної з гармонік, збільшуючи таким чином їх інформативність.

У цьому ж режимі «Track» проводиться запис амплітудних і фазових частотних характеристик а також годографа коливальної системи, якою є, наприклад, працююча машина в цілому або її ротор. Це необхідно при здійсненні більш детального аналізу її вібраційного стану (визначення комплексних передавальних функцій, власних частот, динамічної жорсткості, коефіцієнтів демпфірування і т. п.).

У даному випадку виходять з того, що вимушеною силою, що діє на систему, є відцентрова сила, що виникає при обертанні ротора з неминуче присутнім у ньому дисбалансом. Тому її частота дорівнює частоті обертання ротора, і якщо є можливість керувати цією частотою, можна побудувати залежності амплітуди і фази вимушених коливань системи від частоти цієї змушуючої сили, тобто частоти обертання ротора, отримавши таким чином АЧХ і ФЧХ.

На практиці це здійснюється наступним чином:

– забезпечується плавна зміна частоти обертання ротора машини, яке контролюється по відсутності мигаючих точок на індикаторі 7 (допустима швидкість зміни частоти обертання залежить від ширини смуги пропускання фільтра приладу, встановленої перемикачем 11);

- реєструється рівень і фазовий кут коливань (віброшвидкість або віброзміщення) на відслідковуваній обертовій частоті;

- на вхід «Х» двокоординатного самописця з відповідного виходу приладу подається сигнал, пропорційний частоті обертання ротора, а на вхід «У»

- сигнал, пропорційний або рівню відслідковуваної обертової частотної складової, або величині її фазового кута, отримуючи таким чином АЧХ або ФЧХ відповідно.

В якості збудника гармонічних коливань із змінною частотою можуть застосовуватися зовнішні, наприклад, електродинамічні збудники коливань. Управлятися такі збудники можуть наявними в приладі генератором (за умови використання відповідного підсилювача потужності).

У цьому випадку використовується режим «Transfer» приладу, в якому:

- за допомогою перемикача 10 задається швидкість і напрямок зміни частоти генератора (а, відповідно, швидкість і напрямок переналаштування частоти фільтру) і з виходу приладу сформований синусоїдальний сигнал через підсилювач подається на віброзбудник, укріплений на машині;

- реєструється рівень і фазовий кут коливань (віброшвидкість або віброзміщення) на відслідковуваній частоті;

- на вхід «Х» двокоординатного самописця з відповідного виходу приладу подається сигнал, пропорційний частоті налаштування фільтра (дорівнює частоті коливань віброзбудника), а на вхід «У» – сигнал, пропорційний або рівню відслідковуваної частотної складової, або величині її фазового кута, отримуючи таким чином АЧХ або ФЧХ відповідно.

3.2.2. Лабораторна установка для дослідження частотних характеристик механічних систем.

Для вивчення способів отримання частотних характеристик механічних систем в лабораторії кафедри створена лабораторна установка, що представляє

собою найпростішу одномасову систему з одним ступенем свободи, яка власне і піддається дослідженням.

Основою лабораторної установки (рисунок 3. 3) є вібростіл типу ESE – 201 і багатофункціональний вимірювальний прилад "Vibroport".

Вібростіл – котушка мідного дроту, що знаходиться в магнітному полі, закріплена в нерухомому корпусі на пружному підвісі, допускаючи свободу осьових переміщень. Таким чином, рухлива система вібростола являє собою механічну динамічну систему другого порядку – коливальну ланку, що повністю відповідає схемі, представлений на рисунку 3.1, і рівнянням динаміки, розглянутим вище.

При проходженні через котушку електричного струму внаслідок електромагнітної взаємодії з'являються діючі на рухому систему осьові сили, пропорційні силі струму і викликають її осьовий зсув. Змінний струм викликає вимушені коливання котушки в магнітному полі, причому при незмінній амплітуді сили струму в межах частотного діапазону 5 – 2000 Гц амплітуда збурюючого силового впливу залишається практично постійною.

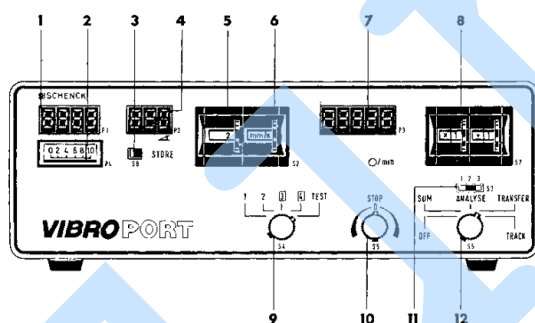


Рисунок 3.3 – Лабораторна установка для дослідження амплітудних і фазових частотних характеристик.

Плавно змінюючи частоту електричного струму і реєструючи реакцію системи – амплітуду і фазу коливань (щодо збурюючого впливу), можна побудувати її амплітудну і фазову частотну характеристики. Для забезпечення достатньої амплітуди коливань сигнал з приладу "Vibroport" надходить на

вібростіл через підсилювач типу LV – 102 потужністю 50 Вт, який підключається до вібростолів через резистор R, що знижує залежність струму в котушці вібростолу від частоти живлячої напруги.

Джерелом синусоїдальної напруги для вібростолу служить вбудований генератор приладу "Vibroport", що забезпечує плавну зміну частоти в діапазоні 1 ... 16 66 Гц з регульованою швидкістю. Цей же прилад містить комплект вихрострумів апаратури для вимірювання переміщень, за допомогою якої реєструються вібропереміщення рухомої системи вібростолу: ДВС – датчик вихрострумів і ПЗН – перетворювач величини зазору між датчиком і досліджуванним об'єктом в пропорційну йому напругу струму.

У комплект входить також двокоординатний самописець, горизонтальний канал «X» якого підключений до виходу частотного детектора, а вертикальний канал «Y» – до виходу амплітудного або фазового детектора приладу "Vibroport". Тобто, переміщення пера самописця по осі X буде пропорційним заданій приладом "Vibroport" частоті збудовуючої сили, а по осі Y – пропорційний відповідно виміряній приладом "Vibroport" амплітуді або фазі вимушених коливань досліджуваної системи, що і дозволяє отримувати таким чином відповідно АЧХ або ФЧХ системи.

Приклади отриманих на лабораторній установці за допомогою багатофункціонального вимірювального приладу "Vibroport" амплітудних і фазових частотних характеристик наведено на рис. 3.4 і 3.5 відповідно.

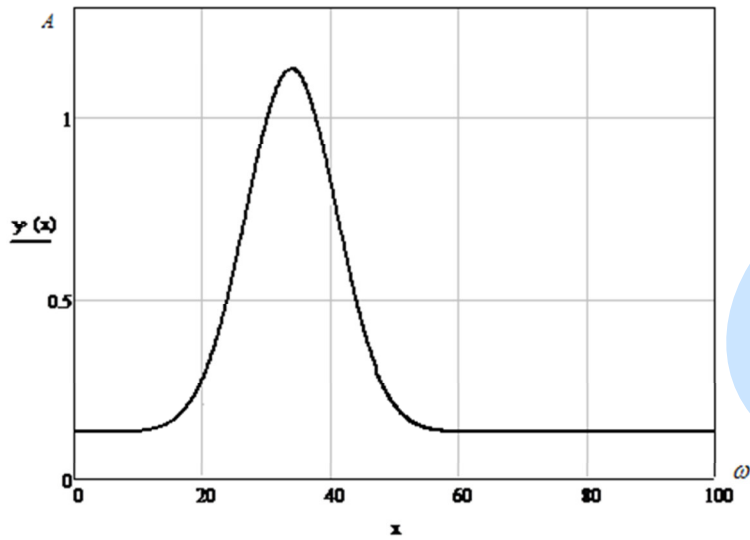


Рисунок 3.4 – Амплітудна частотна характеристика

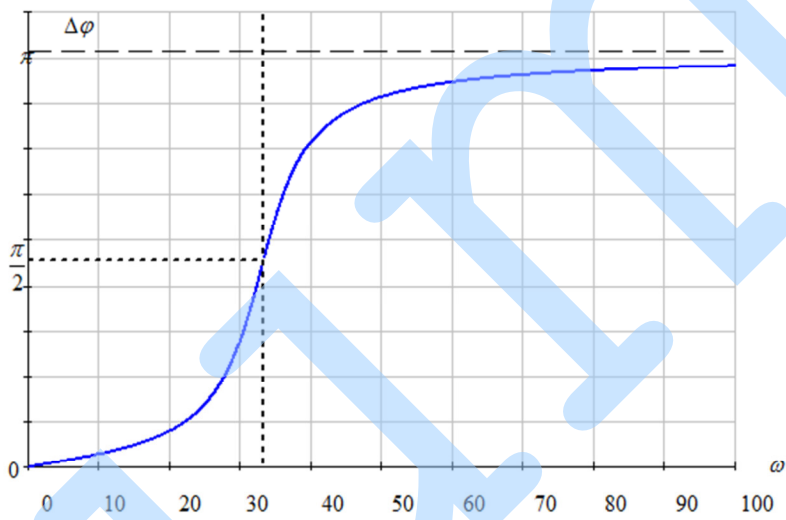


Рисунок 3.5 – Фазова частотна характеристика

3.3 Відпрацювання методів побудови віртуальних приладів та дослідження частотних характеристик з використанням програмного комплексу LabVIEW на прикладі аналогових фільтрів.

Завдання побудови та дослідження аналогових фільтрів залишається актуальною і затребуваною, незважаючи на успіхи цифрової фільтрації. Для її вирішення необхідні не тільки ефективні засоби синтезу та моделювання, які в даний час успішно реалізуються за допомогою відповідного програмного забезпечення, але і високопродуктивні інструменти для автоматизації експериментального визначення амплітудно – частотних (АЧХ) і фазо – частотних (ФЧХ) характеристик активних фільтрів.

Існуючі спеціалізовані прилади не завжди дозволяють забезпечити необхідну для вирішення практичних завдань співвідношення швидкодії, точності та вартості. Вони часто не володіють необхідною широтою функціональних можливостей. Авторами запропоновано вирішення завдання автоматизованого визначення АЧХ і ФЧХ аналогових фільтрів у вигляді апаратно – програмного комплексу (АПК), побудованого із застосуванням технологій NI.

Алгоритм роботи АПК побудований на методі аналізу АЧХ і ФЧХ, заснованому на порівнянні двох сигналів: тестового гармонійного сигналу на вході досліджуваного ланцюга (ДЛ) і сигналу відгуку.

АПК можна розділити на два основні блоки: віртуальний генератор (ВГ) і віртуальний прилад порівняння (ВПС), індукуючий значення отриманих амплітуди і фази. Структурна схема АПК показана на рисунку 3.6.

У встановленому користувачем діапазоні частот ВГ формує гармонійний тестовий сигнал заданої амплітуди. Перебудова частоти генератора визначається числом аналізованих точок у заданому діапазоні. Мінімальний крок, з яким може перебудовуватися генератор, становить 1 Гц.

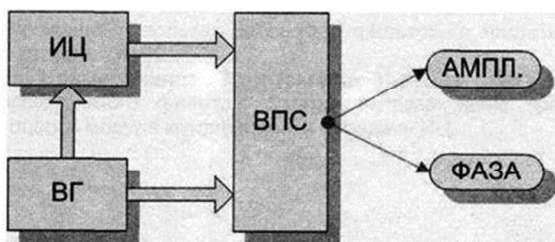


Рисунок 3.6 – Структурна схема АПК

ВПС організовує паралельне зчитування значень тестового сигналу і сигналу відгуку досліджуваної ланцюга. Порівняння результатів вимірювання сигналу відгуку з результатами вимірювання тестового сигналу (замість порівняння з його програмно заданими значеннями) дозволяє враховувати реальні параметри тестового сигналу і тим самим підвищити достовірність визначення АЧХ і ФЧХ. Для виключення впливу перехідних процесів, які можуть виникнути при перебудові генератора, збір даних відбувається з регульованою затримкою. Обчислення амплітуди і початкової фази отриманих сигналів здійснюється стандартною функцією Extract Single Tone Information. Так як при паралельному зборі даних входить до складу АПК плата DAQ 6251 використовує один АЦП на всі канали і мультиплексор (псевдопаралельний збір даних), то виникає постійна міжканальна затримка. Затримка додатково зміщує в часі сигнал відгуку щодо тестового сигналу, викликаючи помилку у визначенні фази, причому похибка вимірювання зростає з ростом частоти. Компенсація між канального затримки здійснюється за допомогою програмної корекції.

Для контролю працездатності ДЛ реалізований критичний запуск АПК. У разі відсутності сигналу відгуку виводиться повідомлення про помилку, а подальше виконання програми віртуального приладу припиняється. У разі відсутності помилки прилад працює в нормальному режимі.

За рахунок організації безперервного буферизованого процесу генерації і зчитування значень сигналів вдалося істотно підвищити швидкодію АПК. Для зручності експлуатації в програмі реалізовані два режими роботи: одноразовий і

безперервний. На відміну від одноразового, безперервний режим передбачає циклічний перезапуск приладу, що зручно при швидкому аналізі АЧХ і ФЧХ досліджуваних схем по невеликому числу точок (20 – 30). Цей режим можна використовувати при ручному налаштуванні досліджуваного фільтра. При аналізі характеристик з великою кількістю досліджуваних точок, краще використовувати одноразовий режим, коли після закінчення вимірювання прилад зупиняється, а всі результати вимірювання і настройки зберігаються до наступного запуску. Лицьова панель приладу (рисунок 3.7) містить наступні вкладки: АЧХ і ФЧХ досліджуваного приладу, Нормована АЧХ, Осцилограми, Установки. Для підвищення зручності експлуатації даного приладу деякі органи управління та індикатори доступні на всіх вкладках меню.

За допомогою регулятора Вибір діапазону встановлюються верхня і нижня межі аналізованого діапазону частот. Для зручності користувача масштаб графіків автоматично змінюється у відповідності з обраним діапазоном частот. Індикатор Прогрес відображає завершеність процесу вимірювання. На вкладці Кількість точок для дослідження вибирається кількість аналізованих точок у вказаному користувачем частотному діапазоні.

На вкладці меню АЧХ і ФЧХ досліджуваного приладу (малюнок 3.7) відображаються отримані графіки АЧХ і ФЧХ досліджуваного фільтра. Таке розташування на одній вкладці створює зручність при роботі з АПК, що має важливу перевагу в порівнянні з традиційними інструментами. Верхній графік відображає залежність напруги у вольтах на виході фільтра від частоти в Гц. На нижньому графіку будується ФЧХ.

На вкладці меню Нормована АЧХ (рисунок 3.8) відображається нормований графік АЧХ досліджуваного фільтра. Дана вкладка меню застосовується при необхідності оцінки модуля коефіцієнта передачі Ш.

На вкладці меню Осцилограми (малюнок 3.9) показуються тимчасові діаграми сигналу, що відображають хід експерименту. Ця функція дозволяє візуально контролювати амплітудні значення і форму вхідного і вихідного сигналів досліджуваного фільтра.

На вкладці меню Налаштування (рисунок 3.10) розташовані органи управління та індикатори приладу. Вкладка містить настройку і контроль основних параметрів роботи АПК: значення амплітуди на вході досліджуваного фільтра, значення амплітуди напруги на виході досліджуваного фільтра, частотний діапазон виміру, крок зміни частоти, величину внесеної корекції ФЧХ в градусах і час, витрачений на проведення останнього вимірювання, а також опцію збереження результатів у файл.

Крім того, тут же вказуються входні та вихідні канали DAQ, використовувані АПК, і налаштування тригера для старту АПК.

Технічні характеристики АПК:

- діапазон досліджуваних частот: 250 – 250 000 Гц;
- мінімальний крок перебудови генератора: 1 Гц;
- похибка установки частоти генератора (не більше): $\pm 0,0015\%$;
- можливе число точок для дослідження: 4 – 249750;
- час дослідження 50 точок: 4,6 сек.

Амплітуда тестового сигналу ВГ: рекомендована 0,1 – 3В; макс. – 10 В.

За результатами серії експериментів відносна похибка вимірювання АЧХ склала (не більше):

- в діапазоні 250 – 100 000 Гц: 0,5%;
- в діапазоні 100 кГц – 200 кГц: 1,4%;
- в діапазоні 200 кГц – 250 кГц: 2,5% Абсолютна похибка вимірювання

ФЧХ (не більше):

- в діапазоні 250 – 100000 Гц: 2 °;
- в діапазоні 100 кГц – 250 кГц: 5 °.

На рисунку 3.11 наведено графіки АЧХ і ФЧХ при роботі АПК в режимі короткого замикання виходу на вхід.

3.4. Дослідження частотних характеристик механічної системи віртуальний приладом в палітрі Signal Processing – Transforms.

Програми, спроектовані в LabVIEW, називаються віртуальним вимірювальним приладами, або VI, оскільки їх зовнішній вигляд і функціонування імітують реальні вимірювальні прилади, такі, як осцилографи і мультиметри. LabVIEW містить повний набір інструментів для збору, обробки, відображення та реєстрації даних, а також кошти, які допомагають налагодити розроблювану програму.

Поряд з набором ВП генерації сигналів Signal Processing®Signal Generation, розглянутим в параграфі 2.1., Бібліотека Signal Processing містить підрозділи Transforms, Signal Operation, Filters і Windows. ВП, які обчислюють ДПФ і ОДПФ сигналу, знаходяться в підрозділі Signal Processing® Transforms. При $N = 2m$ ці ВП реалізують алгоритм БПФ, в іншому випадку – обчислюється ДПФ. На вхід ДПФ можна подати дійсний і комплексний (аналітичний) сигнали. На виході як ДПФ, так і ОДПФ, завжди вийде послідовність комплексних чисел. ДПФ і ОДПФ приймають на вхід одно – і двовимірні послідовності.

На рисунку 3.12 показана структурна схема експериментальної установки для дослідження частотних характеристик механічної системи, створено на основі віртуального приладу LabVIEW. Об'єктом дослідження є динамічний вібростенд, як одномасова коливальна система з одним ступенем свободи. Згенерований віртуальним приладом синусоїдальний сигнал, частота якого може змінюватись в заданих межах, подається на об'єкт дослідження через підсилювач сигналів ПС. Амплітуда коливань рухомої маси об'єкту досліджень фіксується датчиком Д та через аналого – цифровий перетворювач АЦП типу NI USB 6008 вводиться у віртуальний прилад, змодельований на комп'ютері.

Блок – діаграма віртуального приладу наведена на рисунку 3.13. Віртуальний прилад функціонально складається з трьох основних частин: генерація частоти вимушених коливань об'єкта досліджень, взаємодія з АЦП та

введення сигналу з об'єкта дослідження, побудову амплітудно – частотної характеристики.

Більшість сигналів, породжуваних фізичними об'єктами, дійсно – значний. Але в радіо – і гідролокації, в телекомунікаційних системах в процесі модуляції / демодуляції сигнали дуже часто представляють у вигляді «синфазной» і «квадратурной» компонент. В цьому випадку синфазна компонента повинна грати роль дійсної складової вхідного сигналу для БПФ, а квадратурная – уявної. Спектр такого сигналу асиметричен, в цьому випадку він вже не містить надлишкової інформації.

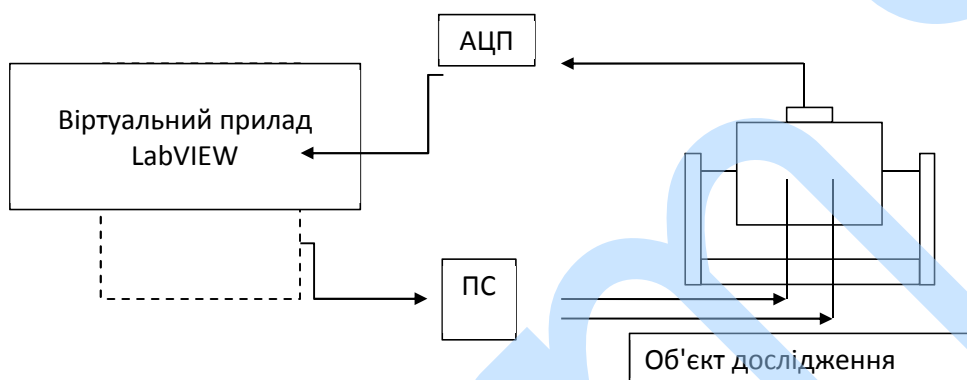


Рисунок 3.12 – Структурна схема експериментальної установки на основі віртуального приладу LabVIEW

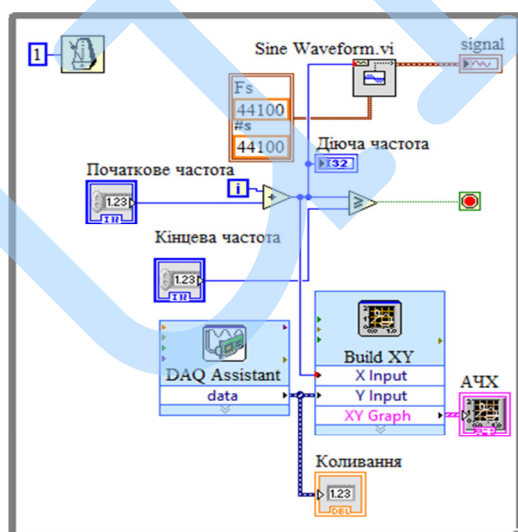


Рисунок 3.13 – Блок – діаграма віртуального приладу

Генерація частоти у віртуальному приладі забезпечується за допомогою , який знаходиться – Signal Processing >> Waveform Generation >> Sine Waveform. Для зміни частоти від заданої до кінцевої, потрібно помістити генератор у цикл (Programming >> Structures >> While Loop), в якому відповідає за початок циклу, а за його зупинку при виконанні умов циклу. В даному випадку початкова частота буде зростати до встановленої "кінцевої частоти" , а не більше завдяки математичної функції (Programming >> Comparison >> Greater or Equal?), після чого програма завершить свою роботу. – так званий Numeric Control, за їх допомогою можна змінювати, в даному ВП, початкову та кінцеві частоти. Numeric Indicator , на блок – діаграмі, відображає на лицьовій панелі частоту, яка генерується у певний відрізок часу. Як і Control, Indicator будується на лицьовій панелі: Modern >> Numeric. На блок – діаграмі відповідає за виведення на лицьову панель графічне зображення сигналу, який створюється генератором. Всі елементи діаграми між собою зв'язані, окрім функції часу . Функція часу відповідає за швидкість виконання процесу (Programming >> Timing)

У LabVIEW передбачено взаємодія віртуально приладу з АЦП та введення сигналу, яка забезпечується за допомогою DAQ Assistant (Express >> Input >> DAQ Assistant). Прилад створений для здійснення збору та генерації даних. Після того як фізична величина перетворена на електричну, її вже можна вимірювати для отримання необхідної інформації.

Побудова амплітудно – частотної характеристики у побудованому віртуальному приладі здійснюється за допомогою XY Graph, який створюється Modern >> Graph >> Ex XY Graph. Графік можна редагувати вже на лицьовій панелі як для візуальної зручності так і для практичності. Для побудови АЧХ було поєднано два сигнали. До "X Input" був направлений сигнал зміни частоти, а до "Y Input" сигнал, отриманий за допомогою DAQ Assistant. Отримавши два сигнали, Build XY Graph генерує їх в один з двох змінними X та Y.

На рисунку 3.14 наведено лицьову панель віртуального приладу, на якій є 2 графічних вікна для відображення згенерованого сигналу та побудованої

амплітудно – частотної характеристики, інтерактивні вікна для задання початкової та кінцевої частоти, а також вікна для відображення поточного значення згенерованої частоти та величини амплітуди вхідного сигналу. Запуск початку побудови частотних характеристик ініціюється натисненням кнопки Run на панелі інструментів LabVIEW

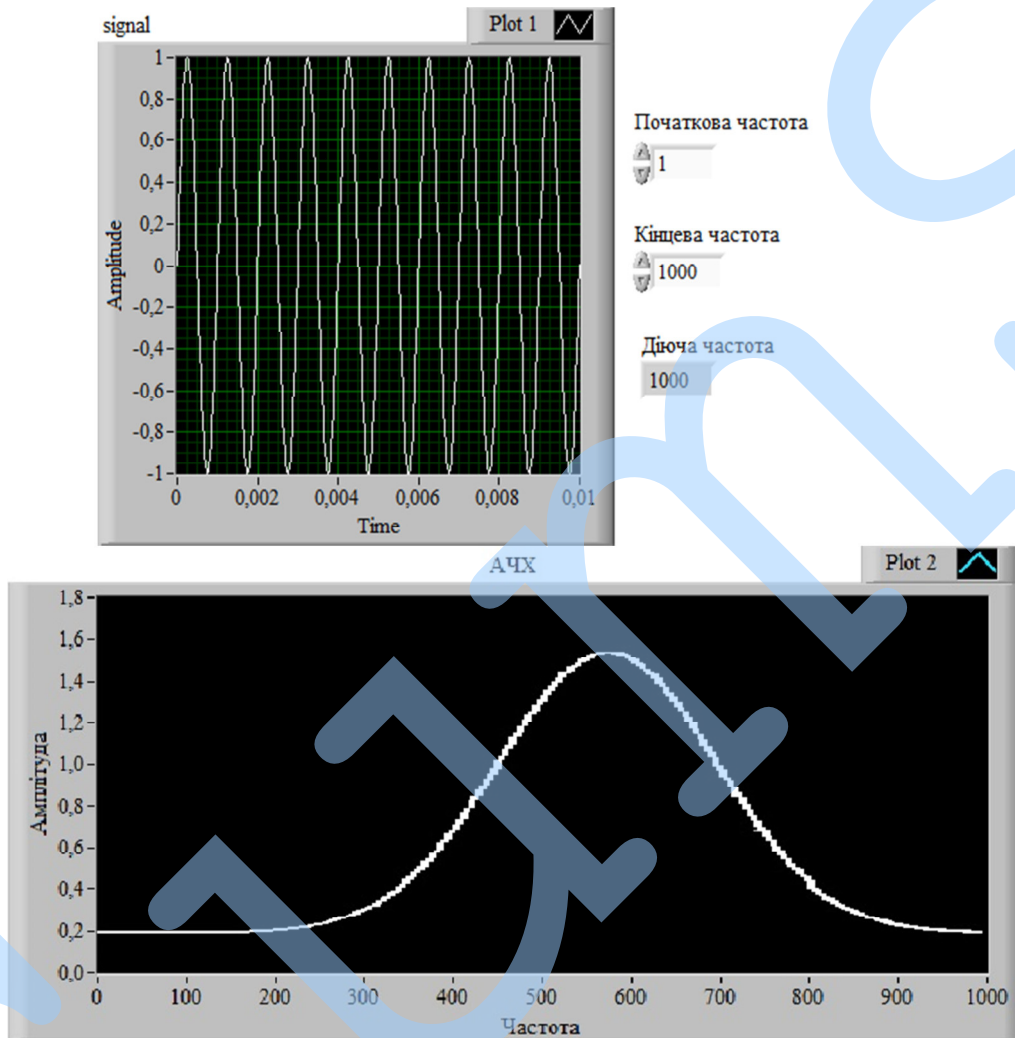


Рисунок 3.14 – Лицьова панель віртуального приладу

Висновки до розділу III:

Для контроля и управления каким – либо процессом или тестирования какого – либо устройства могут быть использованы разнообразные технические средства измерений. И если ясен принцип действия этих устройств, то, как только они будут подключены к компьютеру, они могут стать составной частью виртуальной измерительной системы.

Використання середовища LabVIEW дозволило створити простий і зручний користувальницький інтерфейс, що може динамічно керувати параметрами роботи віртуального приладу як апаратно – програмного комплексу відповідно до завдання користувача. Елементи інтерфейсу апаратно – програмного комплексу підкоряються вимогам ергономіки і функціональності, а не обмеженням, що накладається апаратною побудовою традиційних приладів.

Застосування технологій National Instruments, зокрема АЦП типу NI USB 6008, дозволяє засобами програмного комплексу LabVIEW реалізувати алгоритм, який не пред'являє особливих вимог до апаратних ресурсів персонального комп'ютера і в той же час має високу швидкість і точність.

Даний апаратно – програмний комплекс може знайти застосування в різноманітних наукових дослідженнях, зокрема для досліджень частотних характеристик механічних систем.

ОСНОВНІ ВИСНОВКИ

LabVIEW – потужний і гнучкий програмний пакет для отримання, обробки та аналізу даних. В LabVIEW застосовується мова графічного програмування для створення програм, званих віртуальними приладами (ВП). Користувач взаємодіє з програмою через лицьову панель. Кожній лицьовій панелі відповідає блок – діаграма, яка є вихідним кодом віртуального приладу. LabVIEW має багато вбудованих функцій для полегшення процесу програмування; компоненти зв'язуються між собою провідниками, визначальними шляху потоку даних в межах блок – діаграми.

Програмне забезпечення LabVIEW ідеально підходить для будь – яких вимірювальних систем і систем контролю і є серцем платформ National Instruments. Інтегруючи всі інструменти, які необхідні інженерам і вченим, LabVIEW дозволяє створювати широкий круг додатків в найкоротші терміни. LabVIEW – середовище розробки для вирішення проблем, збільшення продуктивності і створення інновацій.

LabVIEW - потужна інтуїтивно зрозуміла графічна середовище програмування. Вам більше не треба друкувати текст і розбиратися в семантиці мови. Тепер Ви можете повністю зосередитися на завданні, замість того щоб розбиратися в інструменті її рішення.

Для створення зручного і сучасного інтерфейсу в LabVIEW є велика кількість інструментів управління та індикації (осцилограми, графіки, кнопки, перемикачі та ін). Більш того в LabVIEW є можливість створення своїх елементів управління.

LabVIEW має понад тисячу вбудованих функцій для обробки і аналізу сигналу. Тепер Вам не доведеться витратити час на написання стандартних програм, таких як перетворення Фур'є або Лапласа, апроксимація кривих, обчислення коренів полінома і т.д.

Стандартний набір інструментів 'SignalAnalysis Express VIs' дозволяє проводити аналіз, обробку, перетворення сигналів, а також імітувати різні типи

частотних форм. Для розширення можливостей середовища LabVIEW з обробки сигналів служать зовнішні модулі, такі як 'Advancedsignalprocessingtoolkit' й 'SoundandVibrationtoolkit'. Дані доповнення дозволяють виробляти більш детальний і глибокий аналіз сигналів, включаючи метод спектра обвідної вібросигналу, реалізація якого є головним завданням конструкційного додатка

Розглянуто приклади застосування сучасних пакетів при математичному моделюванні пристроїв і систем та відпрацьовано методи побудови віртуальних приладів та дослідження частотних характеристик з використанням програмного комплексу LabVIEW на прикладі аналогових фільтрів.

Використання середовища LabVIEW дозволило створити простий і зручний користувальницький інтерфейс, що може динамічно керувати параметрами роботи віртуального приладу як апаратно – програмного комплексу відповідно до завдання користувача. Елементи інтерфейсу апаратно – програмного комплексу підкоряються вимогам ергономіки і функціональності, а не обмеженням, що накладається апаратною побудовою традиційних приладів.

Застосування технологій National Instruments, зокрема АЦП типу NI USB 6008, дозволяє засобами програмного комплексу LabVIEW реалізувати алгоритм, який не пред'являє особливих вимог до апаратних ресурсів персонального комп'ютера і в той же час має високу швидкодію і точність.

Розроблений апаратно – програмний комплекс може знайти застосування в різноманітних наукових дослідженнях, зокрема для досліджень частотних характеристик механічних систем.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Батоврин В. К., Бессонов А. С., Мошкин В. В., Папуловский В. Ф. LabVIEW: практикум по основам измерительных технологий / под ред. В. К. Батоврина. 2 – е изд, переработ. и доп. – М.: Д М К Пресс. – 232 с.
2. Виноградова Н.А., Лістратов Я.І., Свиридов Є.В. «Розробка прикладного програмного забезпечення в середовищі LabVIEW». Навчальний посібник./ Н.А. Виноградова, Я.І. Лістратов, Є.В. Свиридов. – М.: Видавництво МЕІ, 2005.
3. Глушков В.М., Цейтлин Г.Е., Ющенко Е.Л. Алгебра. Языки. Программирование. – К.: Наукова думка, 1974. – 328 с.
4. Гусак О.Г. Сучасні технології у промисловому виробництві : матеріали науковотехнічної конференції викладачів, співробітників, аспірантів і студентів факультету технічних систем та енергоефективних технологій, м. Суми, 23 – 26 квітня 2013 р.: у 2 – х ч. / Ред.кол.: О.Г. Гусак, В.Г. Євтухов. – Суми : СумДУ, 2013. – Ч.1. – С. 150.
5. Гупаленко, Е.Д. Применение программного комплекса LABVIEW для исследования частотных характеристик механических систем [Текст] / Е.Д. Гупаленко, Е.Н. Савченко // Сучасні технології у промисловому виробництві : матеріали науково – технічної конференції викладачів, співробітників, аспірантів і студентів фак – ту технічних систем та енергоефективних технологій, м. Суми, 23 – 26 квітня 2013 р.: у 2 – х ч. / Ред.кол.: О.Г. Гусак, В.Г. Євтухов. – Суми : СумДУ, 2013. – Ч.1. – С. 150.
6. Дьяконов В. П. Simulink 5/6/7: Самоучитель. – М.: ДМКПресс, 2008. – 784 с.: ил
7. Джеффри Тревис. LabVIEW для всех /: Пер. с англ. Клушин Н. А. ~ М.: ДМК Пресс; ПриборКомплект, 2005. ~ 544 с.; ил.
8. Кухтенко А.И. Кибернетика и фундаментальные науки. – К.: Наукова думка, 1987. – 141 с.

9. Кельтон В., Лоу А. Имитационное моделирование. Классика CS. 3 – е изд. – СПб.: Питер; К.: ВHV, 2004. – 847 с.
10. Колесов Ю.Б., Сениченков Ю.Б.. Моделирование систем. Динамические и гибридные системы. Учеб. пособие/ Ю.Б. Колесов, Ю.Б. Сениченков – СПб.: БХВ – Петербург, 2006. 224 с.
11. Климентьев Е.К. Основы графического программирования в среде LabVIEW. Учебное пособие. Самара: Самар. гос. аэрокосм. ун – т, 2002 г. – 65 с.
12. Ли К. Основы САПР (САР/САМ/САЕ)./СПб: Питер, 2004. 560 с.
13. Лазарев Ю. Моделирование процессов и систем в MATLAB: Учебный курс. – СПб.: Питер; Киев: Изд. группа ВHV, 2005.
14. Математическая Энциклопедия. – Т1, Т.3, Т5./ М.: Советская энциклопедия.
10. Маликов Р.Ф. Основы разработки компьютерных моделей сложных систем: учеб. пособие./ Р.Ф. Маликов– Уфа: Изд – во БГПУ, 2012. – 257с.
11. Новиков Ф.А. Дискретная математика для программистов. – СПб.: Питер, 2002. – 304 с.
12. Рыжиков Ю.И. Имитационное моделирование. Теория и технологии. – СПб.: КОРОНА принт; М.: Альтекс – А, 2004. – 384 с.
13. Солонина А. И. Основы цифровой обработки сигналов: курс лекций / А. И. Солонина, Д.А. Улахович, С.М. Арбузов, Е.Б. Соловьева. 2_е изд. испр. и перераб. – СПб.: БВХ_Петербург, 2005. – 768 с.
14. Томашевский В.М. Моделирование систем. – К.: Видавнична група ВHV, 2005. – 332 с.
15. Сергиенко А.Б. Цифровая обработка сигналов: учебник для вузов. 2_е изд. – СПб.: Питер, 2006. – 751 с.
16. Федосов В. П. Цифровая обработка сигналов в LabVIEW: учеб. пособие / под ред. В. П. Федосова. – М.: ДМК Пресс, 2007. – 456 с.
17. <http://www.labview.ru>.
18. <http://www.ni.com/russia>