

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Сумський державний університет**  
**Факультет технічних систем та енергоефективних технологій**  
**Кафедра комп'ютерної механіки імені Володимира Марцинковського**

«До захисту допущено»

Завідувач кафедри  
\_\_\_\_\_ Андрій ЗАГОРУЛЬКО  
(підпис)  
\_\_\_\_\_ 2023 р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
**на здобуття освітнього ступеня магістр**

зі спеціальності 131 Прикладна механіка, освітньо-професійної програми «Комп'ютерна механіка», на тему: Моделювання динамічних процесів у роторних машинах за допомогою системи комп'ютерного моделювання MATLAB.

Здобувача групи КМ.м-21 ТАРАСЕНКО Діани Романівни.

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

\_\_\_\_\_ Діана ТАРАСЕНКО  
(підпис)

Керівник: доцент, к.т.н., доцент Євген САВЧЕНКО \_\_\_\_\_  
(підпис)

## РЕФЕРАТ

Кваліфікаційна робота бакалавра: 74 с., 59 рис., 15 дж.

Мета роботи: дослідження можливостей системи MATLAB для моделювання динамічних процесів у роторних машинах.

Об'єкт дослідження: динамічні процеси у роторних машинах.

Предмет дослідження: можливості системи комп'ютерного моделювання програмного комплексу MATLAB для моделювання динамічних процесів у роторних машинах.

Методи дослідження: комп'ютерне моделювання за допомогою програмного комплексу MATLAB.

Дослідження проводиться за допомогою комп'ютерного моделювання з використанням програмного комплексу MATLAB. Актуальність даної теми проявляється у вдосконаленні існуючих методів та у пошуку та розробці нових підходів до моделювання динамічних процесів у роторних машинах. Це здійснюється використовуючи нові можливості, які виникають з розвитку сучасних програмних комплексів, зокрема, таких як MATLAB.

Аналіз програмних можливостей системи MATLAB вказує на те, що вона є унікальною прикладною системою програмування та моделювання. Ця система дозволяє проводити різноманітні розрахунки у будь-якій галузі науки та техніки. Особливо важливою є її роль в математичному моделюванні динамічних процесів у роторних машинах з урахуванням гідроаеродинамічних, віброакустичних та інших явищ, які мають місце у цих системах.

Вивчення особливостей системи MATLAB підтверджує, що вона має спеціальні засоби з потужними можливостями, які сприяють розробці високоефективних віртуальних приладів для вирішення різноманітних завдань механіки.

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ, MATLAB, РОТОРНІ МАШИНИ,  
ДИНАМІЧНІ ПРОЦЕСИ

## ЗМІСТ

ВСТУП	6
1 МОДЕЛЮВАННЯ ДИНАМІЧНИХ ПРОЦЕСІВ ЗА ДОПОМОГОЮ СИСТЕМИ MATLAB .....	8
1.1 Система MATLAB і її можливості по обробці сигналів	8
1.2 Програма Signal Processing Tool (SPTool): головне вікно	10
1.3 Завантаження сигналу	11
1.4 Перегляд графіка сигналу	12
1.5 Спектральний аналіз сигналу	13
1.6 Розрахунок фільтра	14
1.7 Перегляд параметрів фільтра	17
1.8 Фільтрування сигналу	18
1.9 Збереження результатів роботи	20
2 ІНСТРУМЕНТ SIMULINK LTI-VIEWER: ВИЗНАЧЕННЯ І ПОБУДОВА ЧАСТОТНИХ ХАРАКТЕРИСТИК СИСТЕМИ .....	22
2.1 Робота з Simulink LTI-Viewer	22
2.2 Налаштування Simulink LTI-Viewer	26
2.3 Експорт моделі	30
3 ЗНЯТТЯ ЧАСТОТНИХ ХАРАКТЕРИСТИК ОБ'ЄКТА РЕГУЛЮВАННЯ З ВИКОРИСТАННЯМ ВІРТУАЛЬНИХ ФАЗОМЕТРА ТА ВИМІРЮВАЧА ЗГАСАННЯ .....	31
3.1 Опис автоматичного пікового вимірювача згасання	31
3.2 Робота з приладом	35
3.3 Опис фазочастотоміра гармонійних коливань	36
3.4 Етапи вимірювання	37
3.5 Влаштування та функціонування фазо частотоміра	37
4 МОДЕЛЬ СИСТЕМИ, РЕАЛІЗОВАНА У ПІДСИСТЕМІ SIMULINK	41
5 ОСНОВНІ ПРИЙОМИ ПІДГОТОВКИ ТА РЕДАГУВАННЯ МОДЕЛІ .....	51

5.1. Додавання текстових написів	52
5.2 Виділення об'єкта	52
5.3 Копіювання та переміщення об'єктів у буфер проміжного зберігання	53
5.4 Вставлення об'єктів із буфера проміжного зберігання	53
5.5 Видалення об'єктів	53
5.6 З'єднання блоків	53
5.7 Зміна розмірів блоків	55
5.8 Переміщення блоків	56
5.9 Використання команд Undo та Redo	56
5.10 Форматування об'єктів	56
5.11 Обмін даними між середовищем Matlab і S-моделлю	57
5.12 Запуск процесу моделювання S-моделі з середовища Matlab	62
5.13 Створення S-блоків з використанням програм Matlab	63
6 ВСТАНОВЛЕННЯ ПАРАМЕТРІВ РОЗРАХУНКУ І ЙОГО ВИКОНАННЯ .....	67
6.1 Встановлення параметрів розрахунку моделі	68
6.2 Встановлення параметрів обміну робочою областю	71
6.3 Встановлення параметрів діагностування моделі	72
6.4 Виконання розрахунків	73
ВИСНОВКИ .....	74
ПЕРЕЛІК ПОСИЛАНЬ .....	75

## ВСТУП

MATLAB представляє собою інтерактивну систему, яка дозволяє ефективно вирішувати різноманітні обчислювальні завдання, активно використовуючи матриці та вектори та оброблюючи їх у режимі реального часу. Цей підхід значно скорочує час вирішення завдань порівняно з написанням програм на мовах типу C або Fortran. Більше того, MATLAB є відкритою системою, придатною не лише для інтерактивного розв'язання завдань. За допомогою неї можна створювати окремі функції, які органічно доповнюють велику бібліотеку пакета, а навіть розробляти повноцінні програми.

Можливості MATLAB вражаюче розширені, і за швидкістю виконання завдань система часто перевищує своїх конкурентів. Вона є придатною для розрахунків в практично будь-якій галузі науки і техніки. Наприклад, вона широко використовується в математичному моделюванні механічних систем та пристроїв, зокрема у динаміці, гідродинаміці, аеродинаміці, акустиці, енергетиці та інших галузях. Цьому сприяють не лише розширені матричні операції та функції, але й наявність розширюваного пакета Simulink, спеціально призначеного для моделювання динамічних систем та пристроїв, а також багатьох інших пакетів розширень.

В комплексі команд, функцій та прикладних програм системи MATLAB, який постійно розширюється, містяться спеціальні засоби для електротехнічних та радіотехнічних розрахунків, обробки зображень, реалізації нейронних мереж та інших нових напрямків науки і техніки. Ці можливості ілюструються безліччю практично корисних прикладів.

Однією з ключових переваг системи є її відкритість і можливість розширення. Більшість команд і функцій реалізовані у вигляді текстових m-

файлів та файлів мовою С, і всі вони доступні для модифікації. Користувач може створювати окремі файли або бібліотеки файлів для вирішення специфічних завдань науки і техніки.

Загальна легкість модифікації та можливість адаптації системи до вирішення різних завдань наукового та технічного спрямування призвели до створення численних пакетів прикладних програм, які значно розширили області застосування MATLAB.

# 1 МОДЕЛЮВАННЯ ДИНАМІЧНИХ ПРОЦЕСІВ ЗА ДОПОМОГОЮ СИСТЕМИ MATLAB

## 1.1 Система MATLAB і її можливості по обробці сигналів

Signal Processing Toolbox (SPTool) є важливою складовою системи MATLAB, яка надає потужний інструментарій для аналізу, моделювання та проектування пристроїв обробки різноманітних сигналів, включаючи їх фільтрацію та різноманітні перетворення. Цей пакет є ключовим засобом для розробки програм обробки сигналів в сучасних наукових та технічних дослідженнях.

Signal Processing Toolbox надає широкий набір можливостей для розробки лінійних систем і аналізу часових рядів, використовуючи різноманітні техніки фільтрації та останні алгоритми спектрального аналізу. Цей пакет застосовується в різних галузях, таких як обробка аудіо- та відеоінформації, телекомунікації, геофізика, управління в реальному часі, економіка, фінанси та медицина.

Основні функції Signal Processing Toolbox включають моделювання сигналів та лінійних систем, проектування та реалізацію цифрових та аналогових фільтрів, швидке перетворення Фур'є та інші перетворення, оцінку спектрів, статистичну обробку сигналів, параметричне оброблення часових рядів та генерацію сигналів різної форми.

Signal Processing Toolbox є ідеальним інструментом для аналізу та обробки сигналів, використовуючи перевірені алгоритми, обрані за критеріями ефективності та надійності. Він включає широкий спектр алгоритмів для представлення сигналів та лінійних моделей, надаючи

користувачеві гнучкість у створенні сценаріїв обробки сигналів.

Пакет Signal Processing Toolbox також дозволяє швидко розробляти різні цифрові фільтри з різноманітними характеристиками, використовуючи графічний інтерфейс для зручного проектування фільтрів. Заснований на оптимальному алгоритмі швидкого перетворення Фур'є, пакет забезпечує високу ефективність та точність при частотному аналізі та спектральних оцінках.

Новий графічний інтерфейс дозволяє користувачеві візуально оцінювати характеристики сигналів, проводити спектральний аналіз та проектувати фільтри, роблячи Signal Processing Toolbox важливим інструментом для вирішення завдань в області обробки сигналів, таких як вібродіагностика технічного стану машин і механізмів.



## 1.2 Програма Signal Processing Tool (SPTool): головне вікно

SPTool – це інструмент, який надає користувачеві графічне середовище для візуального аналізу сигналів та їхніх спектрів, виконання обчислень і аналізу фільтрів, а також для фільтрації сигналів. Для запуску SPTool, використовуючи командний рядок MATLAB, користувач повинен ввести команду "sptool". Після цього відкриється основне вікно програми, яке продемонстровано на рисунку 1.1.

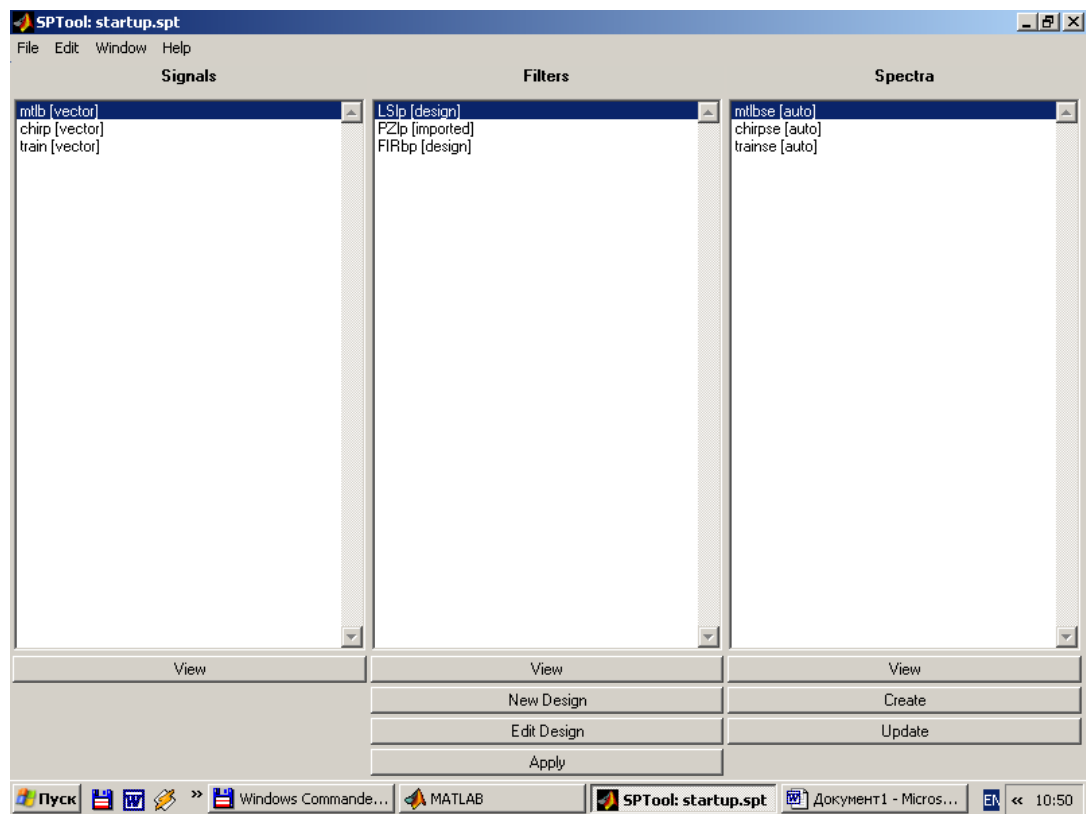


Рисунок 1.1 – Програма SPTool (головне вікно)

У трьох списках цього вікна перераховані завантажені в програму SPTool сигнали (Signals), фільтри (Filters) та спектри (Spectra). Кнопки, розташовані під списками, надають можливість виконувати різні операції, багато з яких будемо розглядати надалі.

Зазвичай використовуються наступні дії, які можна виконати за допомогою програми SPTool: завантаження сигналу, перегляд графіка сигналу,

спектральний аналіз сигналу, розрахунок фільтра та фільтрація сигналу. Далі ми коротко оглянемо кожен з цих етапів.

### 1.3 Завантаження сигналу

Для імпорту сигналу у SPTool, користувачу слід обрати опцію "Import" в меню "File" основного вікна програми. Це призведе до відкриття вікна "Import to SPTool", представленого на рисунку 1.2.

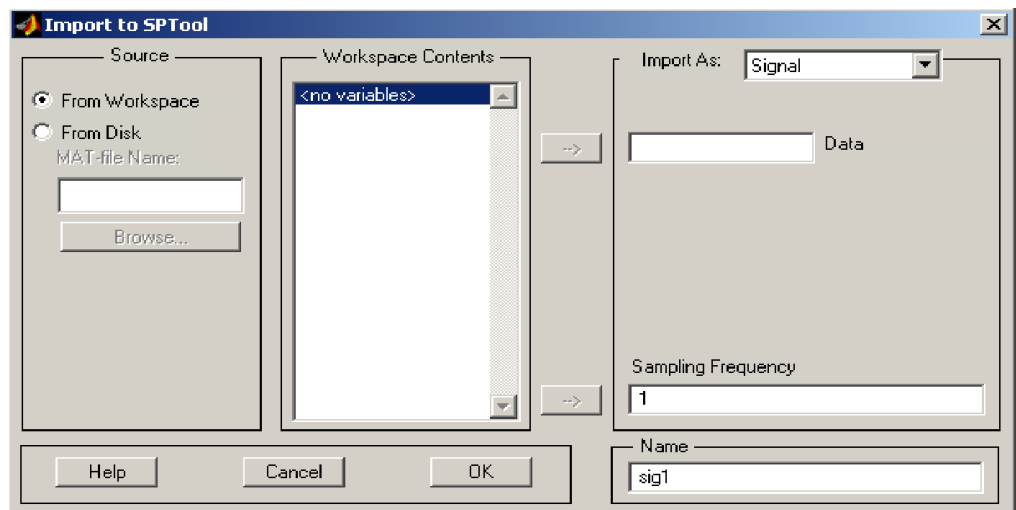


Рисунок 1.2 - Програма SPTool (вікно імпорту даних)

Перемикач "Source" дозволяє вибрати місце зберігання завантаженого сигналу: робочу область MATLAB ("From Workspace") або MAT-файл ("From Disk"). У випадку вибору другого варіанту стає доступним поле для введення імені MAT-файлу, яке можна ввести вручну або використувати кнопку "Browse" для вибору відповідного файлу.

У списку "Workspace Contents" перераховані змінні, що знаходяться у робочій пам'яті MATLAB на даному етапі. При виборі MAT-файлу цей список отримує назву "File Contents" та відображає змінні, які збереглися в обраному файлі.

В розкриваючому списку "Import As" слід вибрати опцію "Signal".

Інші два варіанти, "Filter" і "Spectrum", дозволяють імпортувати описи фільтрів та вже розраховані спектри для подальшого аналізу.

Далі слід обрати змінну зі списку, яка містить відліки завантаженого сигналу, і клацнути кнопку "-->", що розташована поруч із полем введення "Data". Також можна вручну ввести ідентифікатор цієї змінної.

У полі введення "Sampling Frequency" за замовчуванням міститься значення 1, яке можна відредагувати вручну або імпортувати, введши ідентифікатор змінної або вибравши його у списку та клацнувши по кнопці "->".

На останок, у полі введення "Name" можна відредагувати ім'я, під яким цей сигнал буде відображатися у програмі SPTool. Після цього слід натиснути кнопку "OK". Завантажений сигнал з'явиться у списку "Signals" головного вікна програми (рис. 1.1).

#### **1.4 Перегляд графіка сигналу**

Для перегляду графіка сигналу необхідно обрати його у списку "Signals" (рисунок 1.1) та натискати на кнопку "View", розташовану під цим списком. Це призведе до відкриття вікна "Signal Browser", яке представлено на рисунку 1.3.

У цьому вікні відображаються два графіки. Нижній графік із написом "Panor" показує панораму всього сигналу. Початково верхній графік також відображає весь сигнал, але масштаб можна змінити, і тоді верхній графік покаже лише фрагмент сигналу. Позначений прямокутником на нижньому графіку, він вказує на положення цього фрагмента на загальній панорамі. Можна перетягувати цей прямокутник мишею, щоб змінити область огляду.

Кнопки панелі інструментів вікна "Signal Browser" дозволяють

друкувати графік сигналу, відтворювати сигнал за допомогою звукової карти комп'ютера, управляти масштабом відображення, вибрати потрібний канал у випадку багатоканального сигналу та встановлювати потрібний режим відображення маркерів.

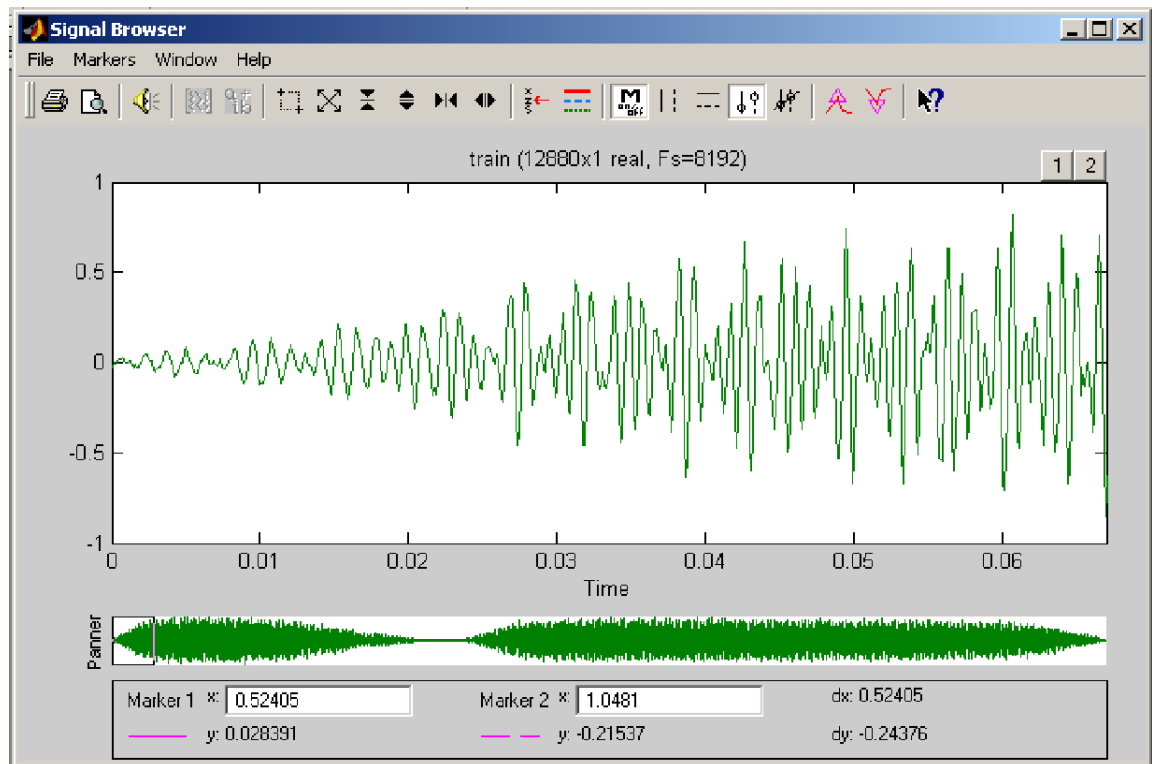


Рисунок 1.3 - Програма SPTool (вікно перегляду сигналів)

На графік можна додавати два маркери, що надають можливість здійснювати кількісні виміри над сигналом. Пересування маркерів здійснюється за допомогою миші, а інформація про відліки сигналу, які вони відзначають, відображається у нижній частині вікна. Цю інформацію можна експортувати до робочої області пам'яті MATLAB у вигляді структури за допомогою команди "Export" у меню "Markers".

### 1.5 Спектральний аналіз сигналу

Для аналізу діапазону сигналу, завантаженого в програму SPTool, необхідно вибрати потрібний сигнал у списку Signals головного вікна програми (рис. 1.1) і клацнути на кнопці Create, розташованій під списком

Spectra. З'явиться вікно Spectrum Viewer, показане на рисунку 1.4.

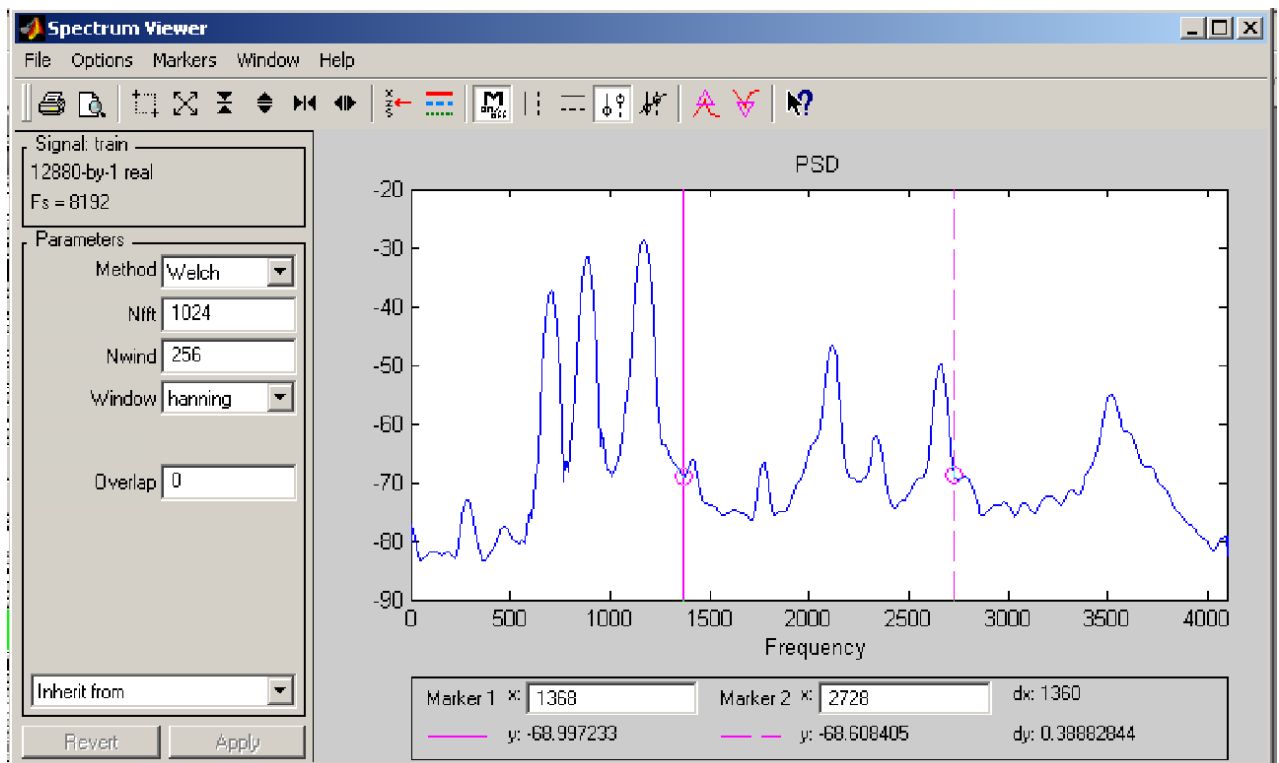


Рисунок 1.4 - Програма SPTool (вікно спектрального аналізу)

У лівій частині вікна можна вибрати метод спектрального аналізу та налаштувати його параметри. У розкриваючому списку, де спочатку вказаний рядок "Inherit from", можна скопіювати повний набір параметрів аналізатора спектра з іншого розрахунку, представленого у списку "Spectra" основного вікна програми.

Після виконання налаштувань параметрів аналізу слід клацнути кнопку "Apply". Система розрахує оцінку спектра сигналу та відобразить відповідний графік. Кнопки панелі інструментів дозволяють виконувати ті ж самі операції, які були вище зазначені для перегляду графіків сигналів. При огляді спектра також можна користуватися маркерами.

### 1.6 Розрахунок фільтра

Для створення нового дискретного фільтра слід натискати кнопку "New Design", розташовану під списком "Filters" у головному вікні SPTool

(рисунок 1.1). Також можна змінити параметри вже налаштованого фільтра, вибравши його в списку "Filters" та вибравши опцію "Edit Design". У будь-якому з цих випадків з'явиться вікно "Filter Designer", представлене на рисунку 1.5.

У розкриваючомуся списку "Algorithm" обирається метод розрахунку, а в розділі "Specifications" визначаються параметри синтезованого фільтра.

Після вибору методу розрахунку та визначення параметрів фільтра слід натискати кнопку "Apply" для виконання синтезу. У центрі вікна буде відображено графік АЧХ фільтра, а в розділі "Measurements" вказані його параметри.

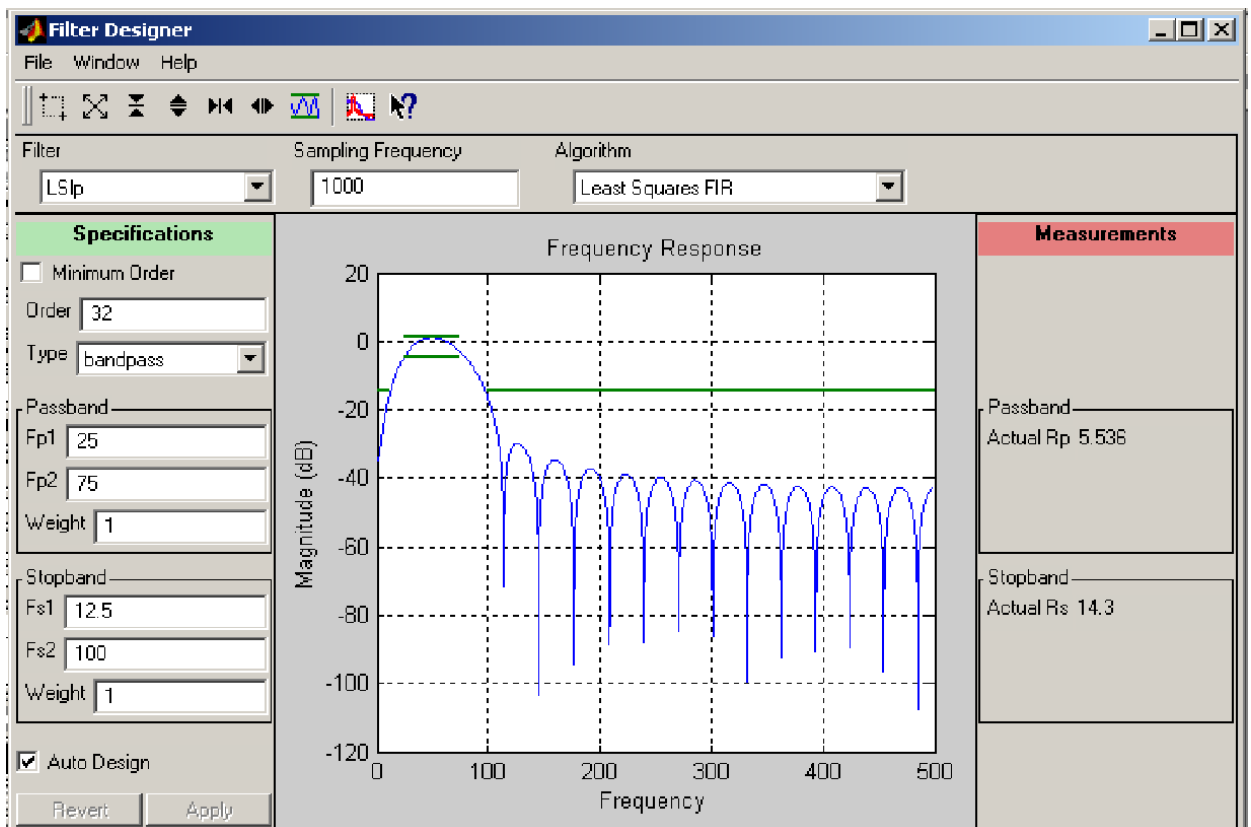


Рисунок 1.5 - Програма SPTool (вікно розрахунку фільтрів)

Кнопки на панелі інструментів надають можливість регулювати масштаб графіку. Крім того, можна вивести деталі графіку амплітудно-частотної

характеристики (АЧХ) у смузі пропускання, а також наложити на графік АЧХ одного із спектрів, які вказані в списку Spectra головного вікна програми.

Можливості програми SPTool для синтезу фільтрів обмежені у порівнянні з програмою FDATool. Єдиний функціонал, який присутній у SPTool і відсутній у FDATool, - це можливість безпосереднього редагування положення нулів та полюсів фільтра. Для використання цієї можливості виберіть в списку Algorithm останній рядок – Pole/Zero Editor. Вікно Filter Designer прийме вигляд, показаний на рисунку 1.6.

У режимі редагування фільтра можна переміщати нулі та полюси за допомогою миші, додавати та видаляти їх, а також автоматично створювати комплексно-сполучені пари.

Так як вікно Filter Designer не виводить графік АЧХ фільтра під час редагування нулів і полюсів, під час використання цього режиму слід одночасно відкривати вікно Filter Viewer, яке буде розглянуто далі. Під час редагування фільтра у вікні Filter Designer його характеристики, що відображаються в вікні Filter Viewer, будуть одночасно оновлюватися. Загалом, вікна програми SPTool не є модальними, що дозволяє відкривати будь-яку їх кількість та вільно перемикатися між ними.

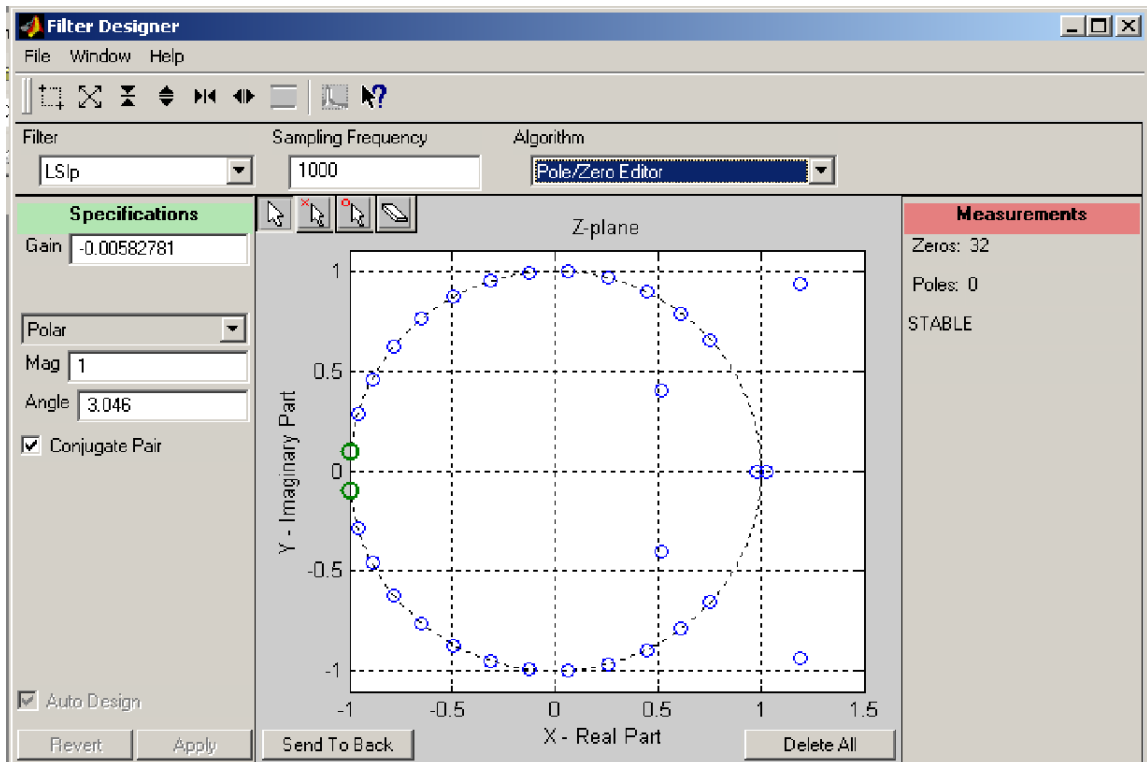


Рисунок 1.6 - Програма SPTool  
(радагування розміщення нулів і полюсів фільтра)

### 1.7 Перегляд параметрів фільтра

Для перегляду характеристик завантаженого в програму SPTool фільтра, потрібно обрати його у списку Filters головного вікна програми (рис. 1.1) та натискати кнопку View, яка розташована під цим списком. Відкриється вікно Filter Viewer, представлене на малюнку 1.7.

У лівій частині вікна Filter Viewer розташована група прапорців Plots для вибору складу відображуваних графіків. В секції Frequency Axis можна визначити частотний діапазон для аналізу характеристик і встановити тип шкали - лінійний або логарифмічний.

Панелі інструментів мають кнопки для управління режимом відображення та виведенням маркерів для поточного графіка, осі якого виокремлена червоним кольором. Інформація про розміщені маркери відображається у нижній частині вікна, як завжди.



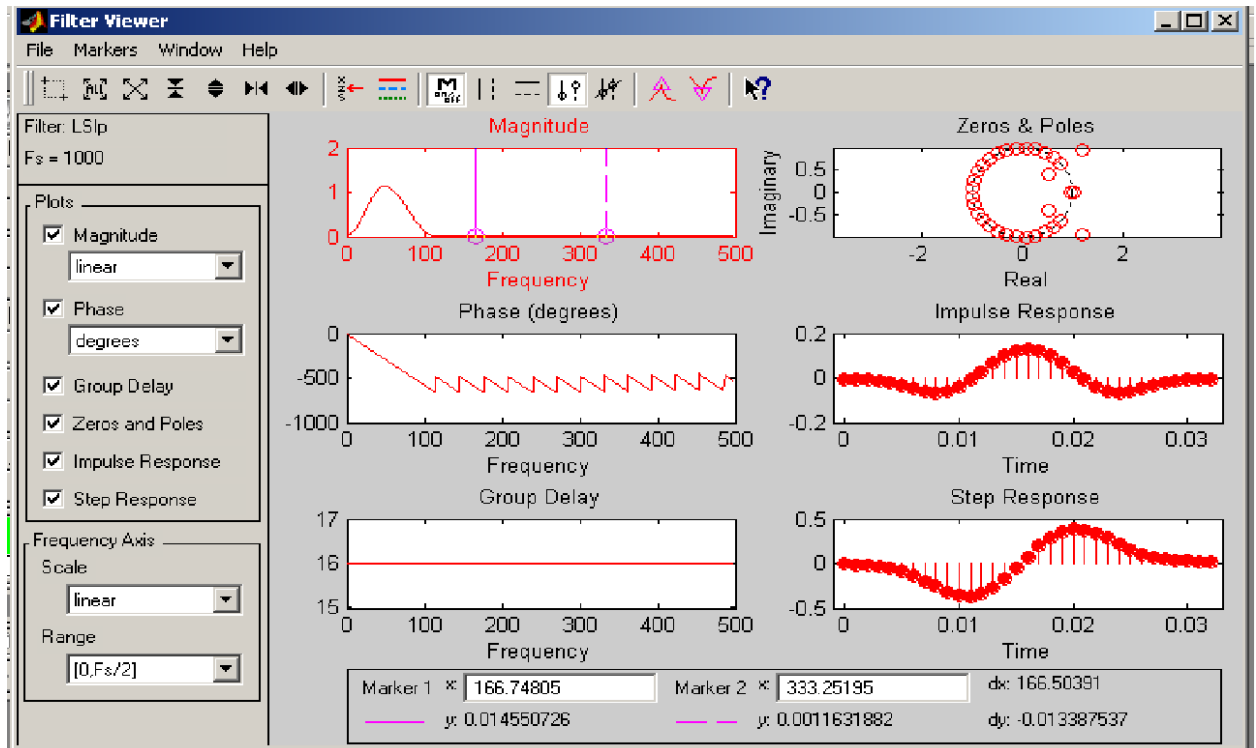


Рисунок 1.7 - Програма SPTool (перегляд характеристик фільтра)

## 1.8 Фільтрування сигналу

Для пропускання сигналу через фільтр у програмі SPTool, треба вибрати відповідні сигнал і фільтр у списках Signals та Filters головного вікна програми (малюнок 1.1) і натиснути кнопку Apply, розташовану під списком Filters. З'явиться вікно Apply Filter, зображене на рисунку 1.8.

Єдиним полем введення в цьому вікні є Output Signal, де потрібно вказати ім'я для нового сигналу. У розкритому списку Algorithm можна обрати функцію MATLAB для фільтрації - filter, filtfilt або fftfilt. Після виконання цих кроків слід натискати кнопку OK. Результатом буде обчислений вихідний сигнал, який з'явиться під вказаним ім'ям у списку Signals головного вікна програми.

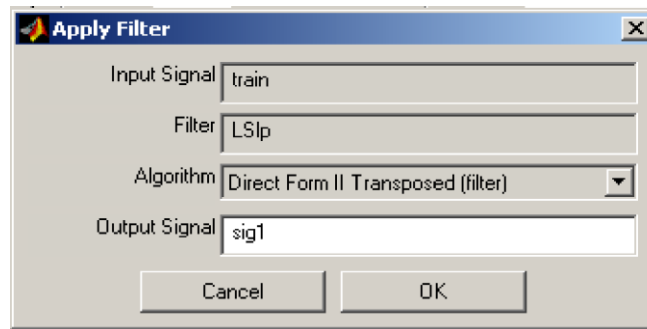


Рисунок 1.8 - Програма SPTool (вікно застосування фільтра)

Для одночасного перегляду графіків вхідного та вихідного сигналів в програмі SPTool, слід вибрати обидва сигнали у списку Signals головного вікна програми (для цього, як у звичайному випадку в Windows, тримаючи натисканою клавішу Ctrl, клацніть на обраних сигналах) і потім натиснути кнопку View. У вікні Signal Browser, яке відкриється, будуть відображені графіки обох вибраних сигналів (рисунок 1.9).

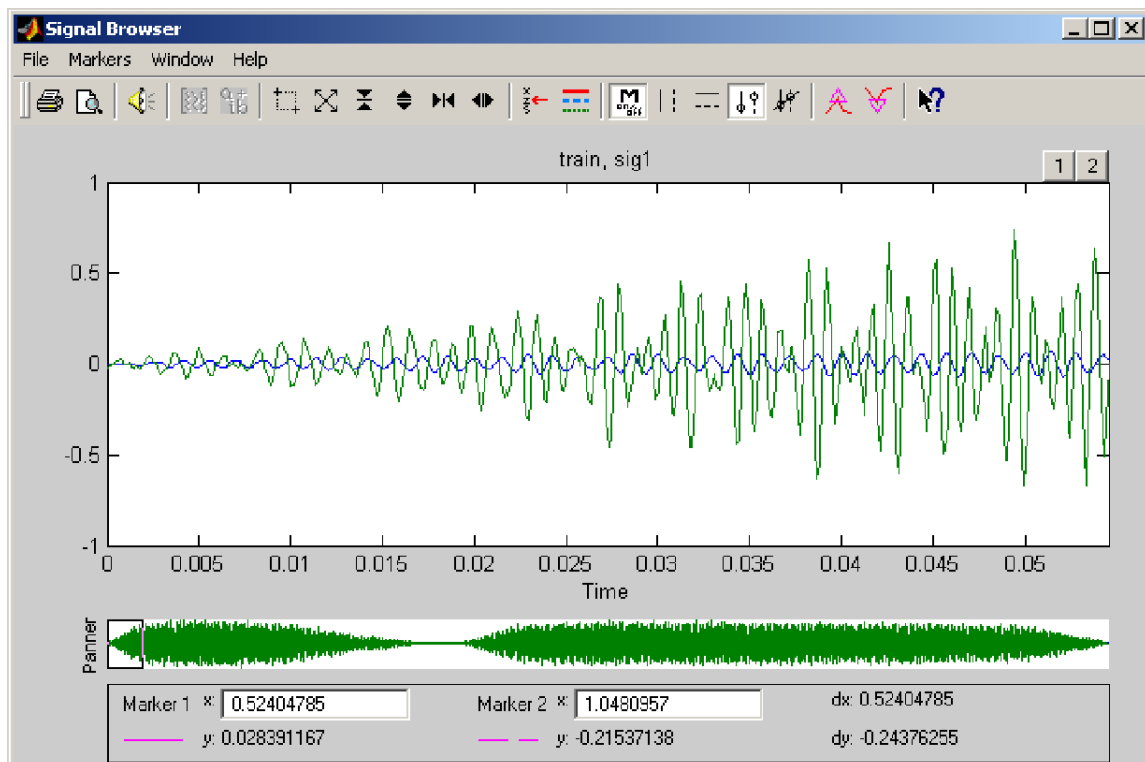


Рисунок 1.9 - Програма SPTool (одночасний перегляд графіків вхідного і вихідного сигналів фільтра)

Аналогічно можна переглядати кілька спектрів або характеристики декількох фільтрів одночасно.

### **1.9 Збереження результатів роботи**

Роботу з SPTool можна зберегти, використовуючи опції "Save Session" або "Save Session As" в меню "File" головного вікна програми. Файли сеансів матимуть розширення SPT. Завантажити збережений сеанс можна, скориставшись командою "Open Session" того ж меню "File".

Додатково, можна експортувати сигнали, фільтри та спектри у вигляді структур даних. Для цього використовується опція "Export" у меню "File" головного вікна програми SPTool. Після вибору цієї опції відкриється вікно "Export from SPTool", яке зображено на рисунку 1.10.

У списку, який знаходиться ліворуч у цьому вікні, слід вибрати об'єкти для експорту. Натискання кнопки "Export to Workspace" дозволить експортувати відповідні структури даних у робочу область пам'яті MATLAB, а для збереження їх у MAT-файл слід скористатися кнопкою "Export to Disk".

Інформацію про склад експортованих структур даних можна знайти у довідковій системі пакету Signal Processing.

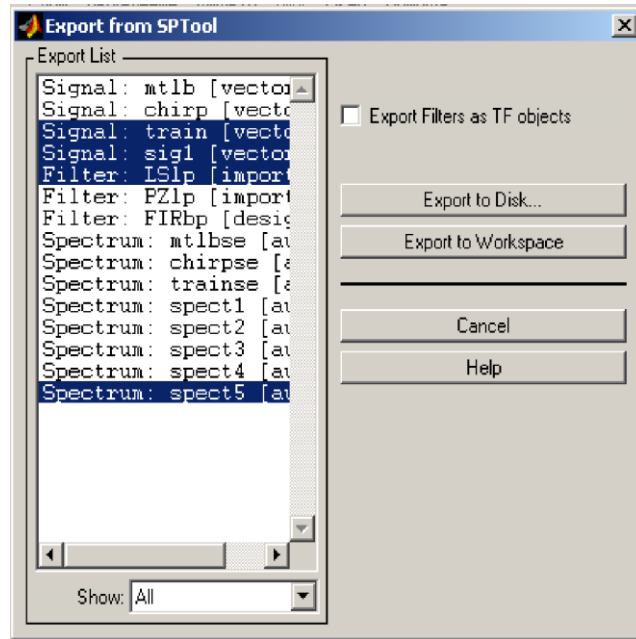


Рисунок 1.10 - Програма SPTool (вікно експорту даних)

## **2 ІНСТРУМЕНТ SIMULINK LTI-VIEWER: ВИЗНАЧЕННЯ І ПОБУДОВА ЧАСТОТНИХ ХАРАКТЕРИСТИК СИСТЕМИ**

### **2.1 Робота з Simulink LTI-Viewer**

Інструмент Simulink LTI-Viewer є складовою частиною Control System Toolbox і використовується для аналізу лінійних стаціонарних систем. З його допомогою можна легко будувати частотні характеристики системи, отримувати відгуки на одиничні ступінчасті та імпульсні впливи, а також визначати нулі та полюси системи. Нижче наведено короткий алгоритм роботи з Simulink LTI-Viewer:

1. Виконати команду "Tools\Linear Analysis..." у вікні Simulink-моделі. В результаті виконання цієї команди відкриється вікно Model\_Inputs\_and\_Outputs, як показано на рисунку 2.1, а також порожнє вікно Simulink LTI-Viewer.
2. Встановити блок "Input Point" на вході та блок "Output Point" на виході досліджуваної системи, як показано на рисунку 2.2.
3. У вікні LTI Viewer виконати команду "Simulink \ Get Linearized Model". Ця команда виконує лінеаризацію моделі та будує реакцію системи на одиничний ступінчастий вплив. Результат виконання цього пункту показаний на малюнку 2.3.

Якщо система має кілька входів та виходів, і для всіх них встановлені блоки "Input Point" і "Output Point", то на графіці буде відображено кілька вікон, які показують реакцію на кожному виході при впливі на кожен вхід.

4. Для отримання інших параметрів системи слід виконати команду "Edit\Plot Configuration..." у вікні LTI Viewer. В результаті виконання

цієї команди відкриється вікно "Plot Configuration", показане на рисунку 2.4.

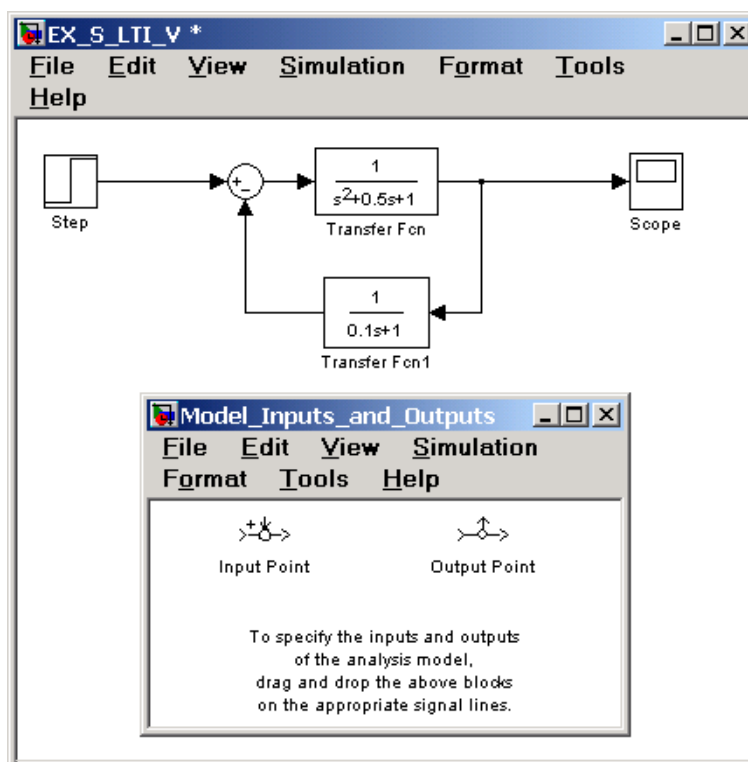


Рисунок 2.1 - Дослідна модель і вікно Model\_Inputs\_and\_Outputs інструмента Simulink LTI-Viewer

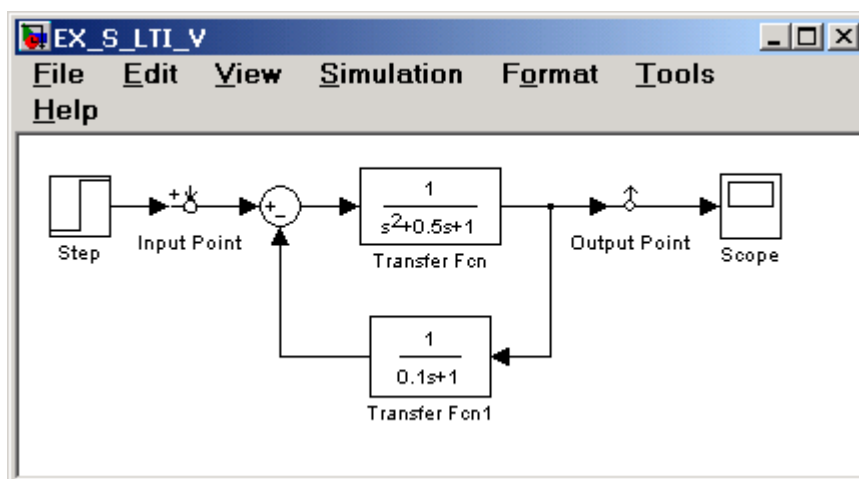


Рисунок 2.2 - Дослідна модель з установленими блоками Input Point і Output Point

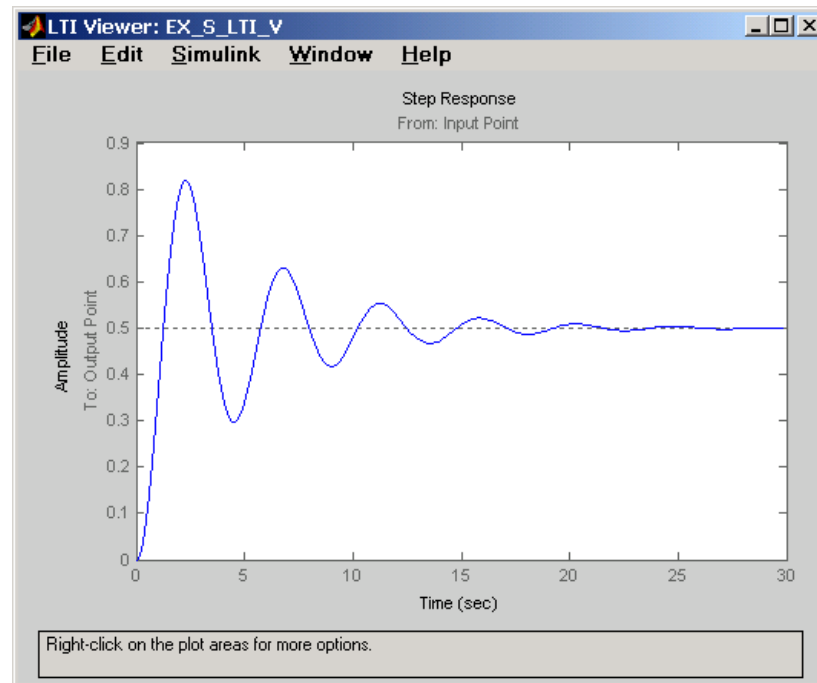


Рисунок 2.3 - Реакція системи на одиничну ступінчасту дію

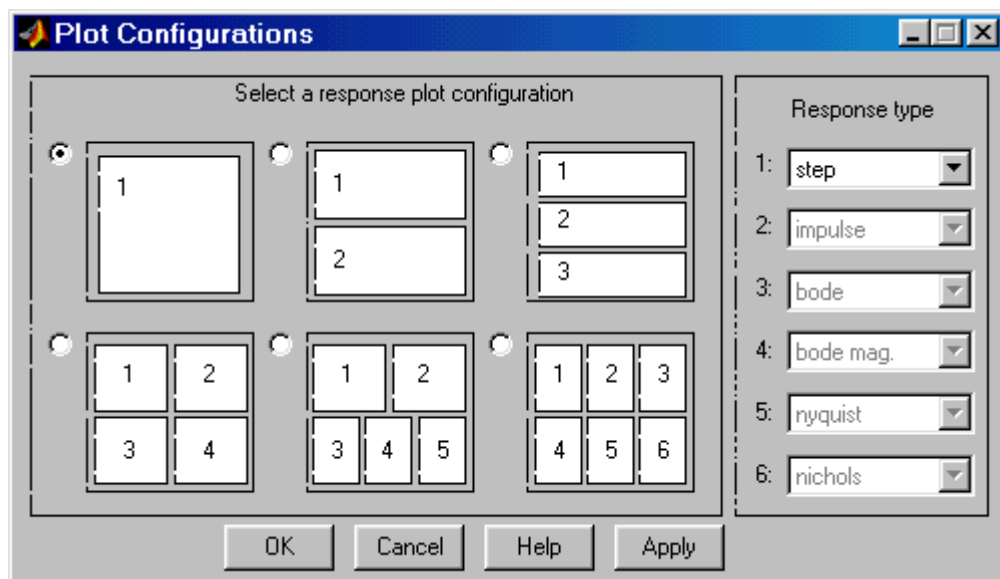


Рисунок 2.4 - Вікно Plot Configuration

У вікні Simulink LTI-Viewer доступна можливість вибрати число відображених графіків через панель "Select a response plot configuration" та обрати тип графіків, що відображаються, за допомогою панелі "Response type". Доступні для побудови такі графіки (діаграми):

- "step" – реакція на одиничний ступінчастий вплив;
- "impulse" - реакція на одиничний імпульсний вплив;
- "bode" - логарифмічні амплітудні та фазові частотні характеристики;
- "bode mag" - логарифмічна амплітудна частотна характеристика;
- "nyquist" - діаграма Найквіста;
- "nichols" - годограф Ніколса;
- "sigma" - сингулярні числа;
- "pole/zero" - нулі та полюси системи.

На рисунку 2.5 наведено приклад вікна Simulink LTI-Viewer із зображенням декількох різних характеристик досліджуваної системи.



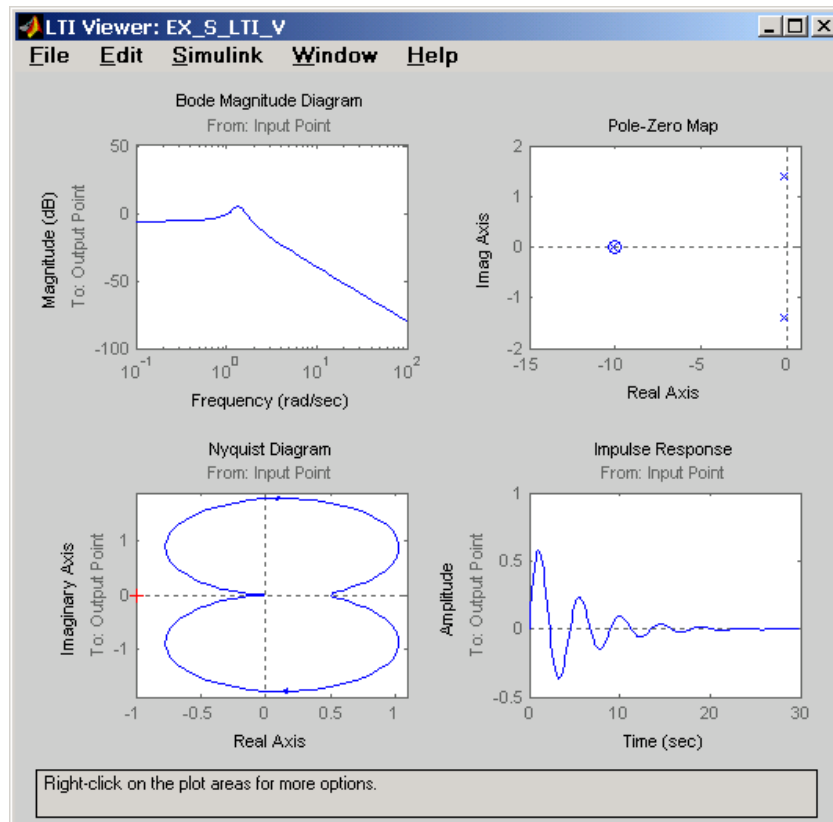


Рисунок 2.5 - Вікно Simulink LTI-Viewer з декількома графіками

Для налаштування зовнішнього вигляду графіків можна скористатися опцією Edit\Line Styles..., де можна встановити вид та кольори ліній, а також вигляд маркерів.

## 2.2 Налаштування Simulink LTI-Viewer

За допомогою команди Edit\Viewer Preferences... виконуються такі види налаштувань:

- 1) Встановлення одиниць виміру (вкладка Units). Вид вікна при налаштуванні одиниць виміру показаний на рисунку 2.6.

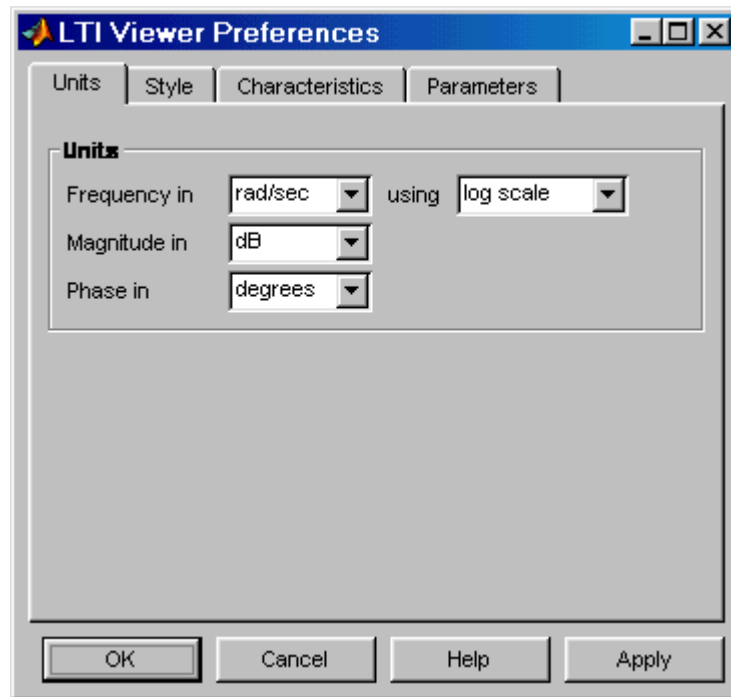


Рисунок 2.6 - Вкладка Units

На вкладці Units вікна можна налаштувати одиниці вимірювання для частоти (рад/с або Гц), рівня (dB або абсолютні одиниці) та фази (градуси або радіани). Також тут можна встановити вид шкали частоти (логарифмічний або лінійний).

Щодо стилів графіків, на вкладці Style можна налаштувати різноманітні аспекти. Зокрема, тут можна налаштувати шрифти вікна Simulink LTI-Viewer (панель Fonts), вибрати колір осей графіків (панель Colors), а також визначити, чи слід відображати лінії сітки на графіках (прапорець Show grids). Вигляд вкладки Style показаний на рисунку 2.7.

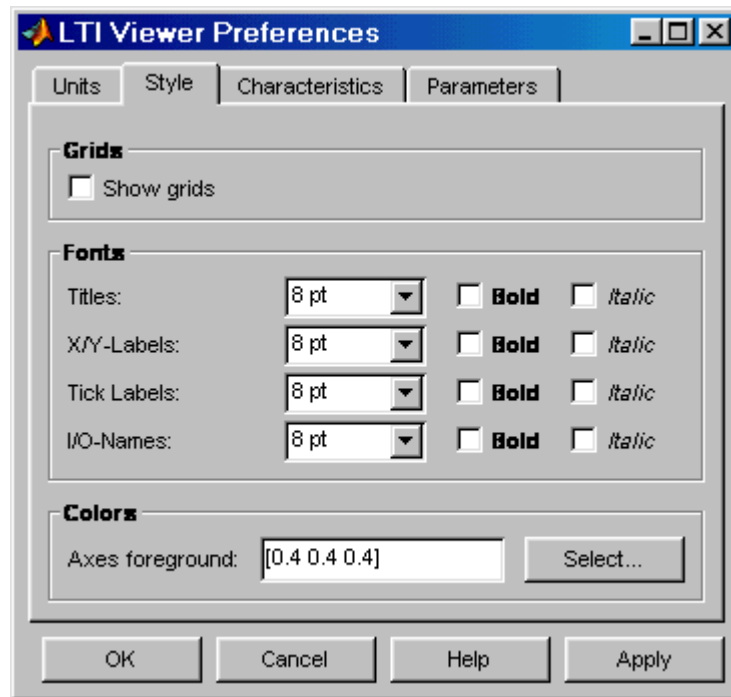


Рисунок 2.7 - Вкладка Style

На вкладці Characteristics можна налаштувати параметри розрахунку перехідного процесу. Тут можна встановити параметри за замовчуванням для обчислення часу наростання та часу перехідного процесу. Simulink LTI-Viewer за замовчуванням обчислює час перехідного процесу як час, коли перехідна функція входить у 2% зону і більше не виходить із неї (параметр Show setting time within). Також можна змінити параметри обчислення часу перехідного процесу (Show rise time from). На цій вкладці є також прапорець Unwrap phase, встановлення якого дозволяє уникнути відображення розривів у фазочастотній характеристиці, пов'язаних з областю визначення функції  $\arctg$ , що обчислює фазовий зсув. Вигляд вкладки Characteristics показано на рисунку 2.8.

Щодо вкладки Parameters, тут визначається часовий інтервал для розрахунку перехідного процесу (панель Time Vector), а також інтервал частот для розрахунку частотних характеристик (панель Frequency Vector). Зовнішній вигляд вкладки Parameters показано на рисунку 2.9.

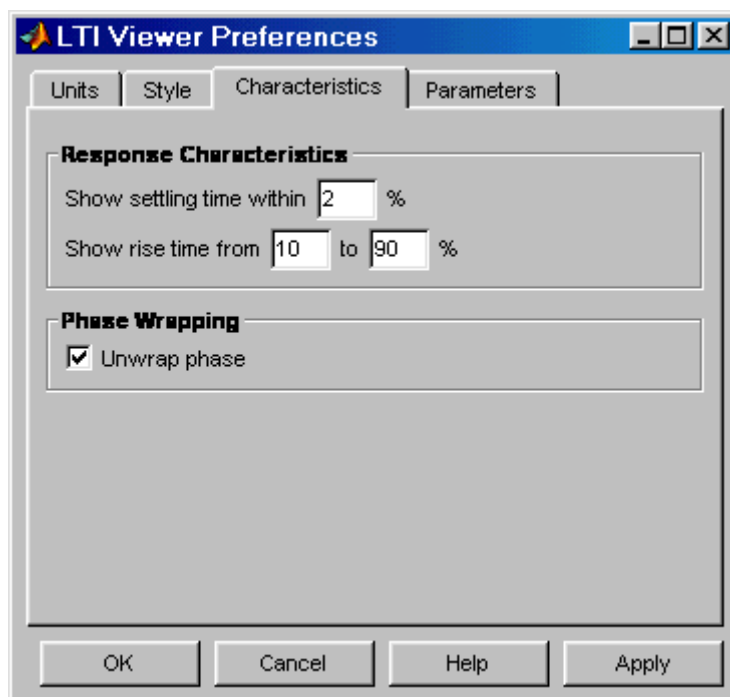


Рисунок 2.8 - Вкладка Characteristics

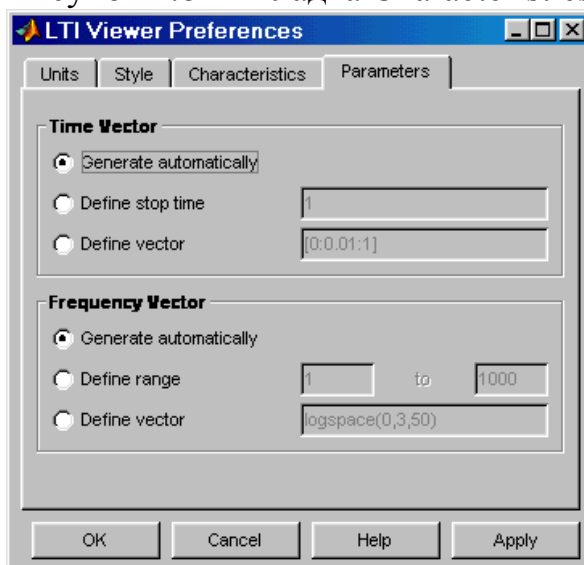


Рисунок 2.9 - Вкладка Parameters

Вектори часу та частоти можна обчислювати в автоматичному режимі (Generate automatically), ввести конкретне значення для часу закінчення розрахунку (Define stop time) або діапазон значень частоти (Define range), або задати безпосередньо вектор значень часу або частоти (Define vector). Вигляд вкладки Parameters показано на рисунку 2.9.

### 2.3 Експорт моделі

Команда File/Export дозволяє експортувати модель у вигляді матриць рівнянь простору стану в робочу область MATLAB чи файл. При виконанні цієї команди відкривається вікно діалогу (рисунок 2.10), де необхідно вибрати варіант експорту.

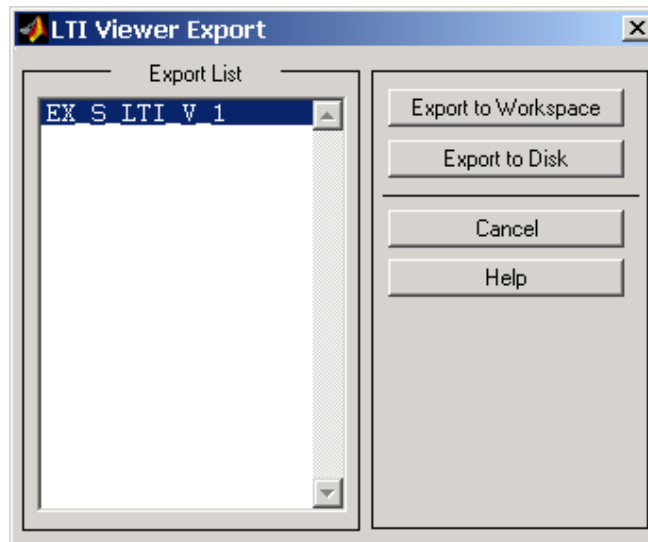


Рисунок 2.10 - Вікно діалогу при експорті моделі

При експорті моделі в робочу область MATLAB утворюється структура з ім'ям\_моделі\_1 (ім'я\_моделі\_2, ім'я\_моделі\_3 і так далі при подальших операціях експорту). Наприклад, для моделі з ім'ям EX\_S\_LTI\_V.mdl ім'я структури буде EX\_S\_LTI\_V\_1. Для перегляду значень матриць рівнянь простору стану потрібно ввести в командному рядку MATLAB ім'я структури та натискати клавішу Enter. Значення матриць будуть виведені в командному вікні.

## З ЗНЯТТЯ ЧАСТОТНИХ ХАРАКТЕРИСТИК ОБ'ЄКТА РЕГУЛЮВАННЯ З ВИКОРИСТАННЯМ ВІРТУАЛЬНИХ ФАЗОМЕТРА ТА ВИМІРЮВАЧА ЗГАСАННЯ

### 3.1 Опис автоматичного пікового вимірювача згасання

Це програмний засіб у середовищі SIMULINK, призначений для отримання амплітудно-частотних характеристик невідомого динамічного об'єкта, що розглядається як "чорний ящик" (рис. 3.1).

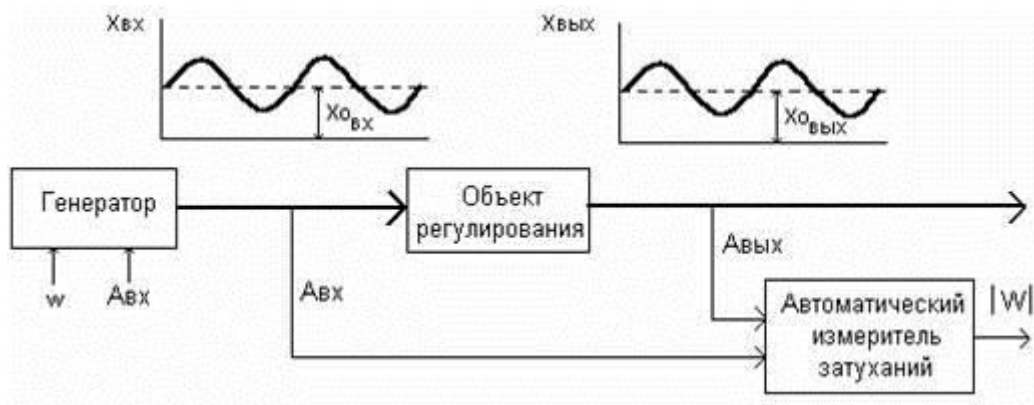


Рисунок 3.1 - Схема віртуального вимірювача згасання

Принцип дії полягає в тому, що при використанні синусоїдального випробувального сигналу достатньо визначити та запам'ятати значення сигналу в точках  $X_{\min}$  та  $X_{\max}$  для входу та виходу (рисунок 3.2).

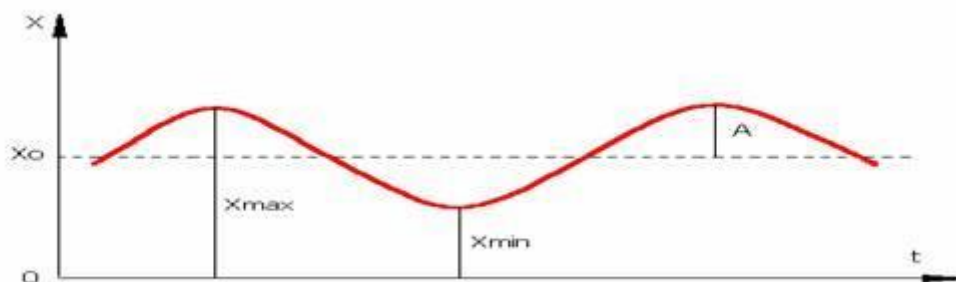


Рисунок 3.2 - Принцип дії вимірювача

Незалежно від частото отримуємо значення амплітуди:

$$A = 0.5 * (X_{\max} - X_{\min}) .$$

Модуль передаточної функції (згасання):

$$|W| = A_{\text{ВЫХ}} / A_{\text{ВХ}} .$$

Прилад також виконує додаткові обчислення постійних складових досліджуваного і вихідного сигналів (зміщення  $X_{0\text{ВХ}}$  и  $X_{0\text{ВЫХ}}$ ):

$$X_0 = 0.5 * (X_{\max} + X_{\min}) .$$

Пристрій було розроблено на базі лінійних, логічних а також перемикальних елементів SIMULINK (рисунки 3.3, 3.4).

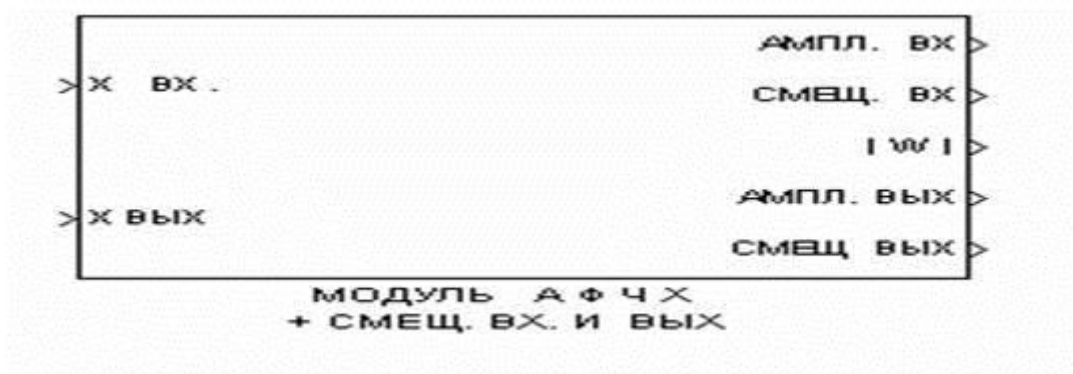


Рисунок 3.3 - Схема пристрою

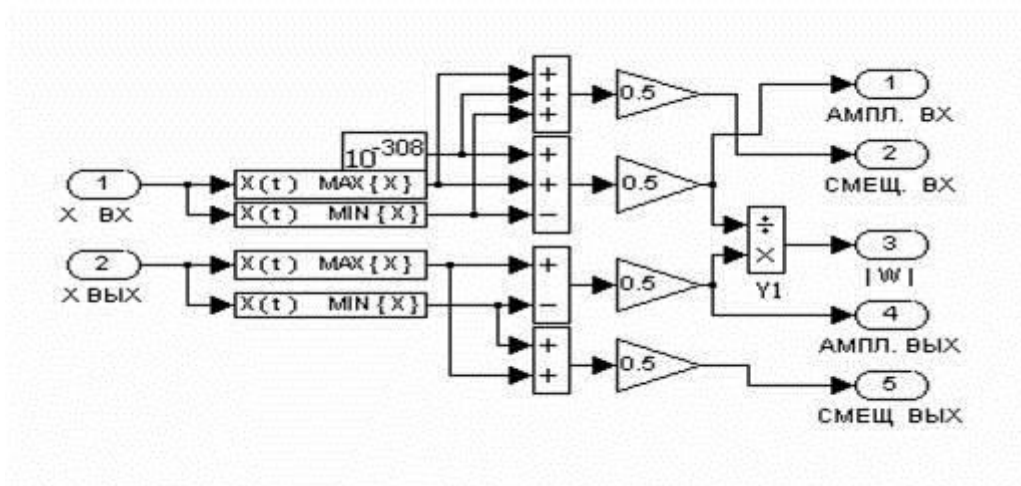


Рисунок 3.4 - Схема пристрою

Важливо відзначити, що при обчисленні згасання в початковий момент у елементі  $Y1$  може виникнути ситуація "ділення на нуль". Для уникнення цього, до вхідного сигналу  $X(t)$  додається нескінченно мале число  $10^{-308}$ .

Крім того, слід відмітити відсутність простих засобів в наборі елементів SIMULINK для пошуку мінімуму або максимуму скалярної функції. З цієї

причини їх довелося додатково створити, використовуючи наявні стандартні елементи.

Схему знаходження мінімуму можна побачити на рисунках 3.5 і 3.6.

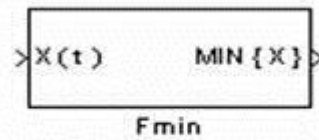


Рисунок 3.5 - Схема для знаходження мінімуму

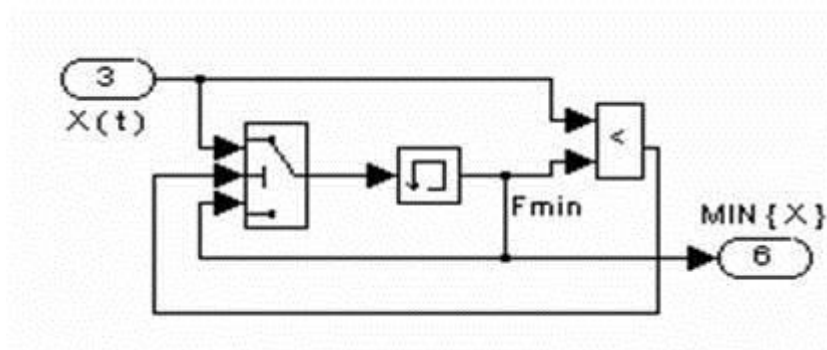


Рисунок 3.6 - Схема для заходження мінімуму

Операція схеми (рисунок 3.6) виглядає наступним чином: вхідний сигнал  $X(t)$  надходить до блоку аналізатора та порівнюється з його попереднім значенням. У початковий момент часу  $X(t)$  порівнюється з числом, яке встановлено як початкові умови блоку пам'яті. У схемі для пошуку мінімуму рекомендується встановлювати як початкові умови блоку пам'яті число  $10^{307}$  (або  $+\text{inf}$ ), а при знаходженні максимуму (рисунок 3.9) - число мінус  $10^{307}$  (або  $-\text{inf}$ ).

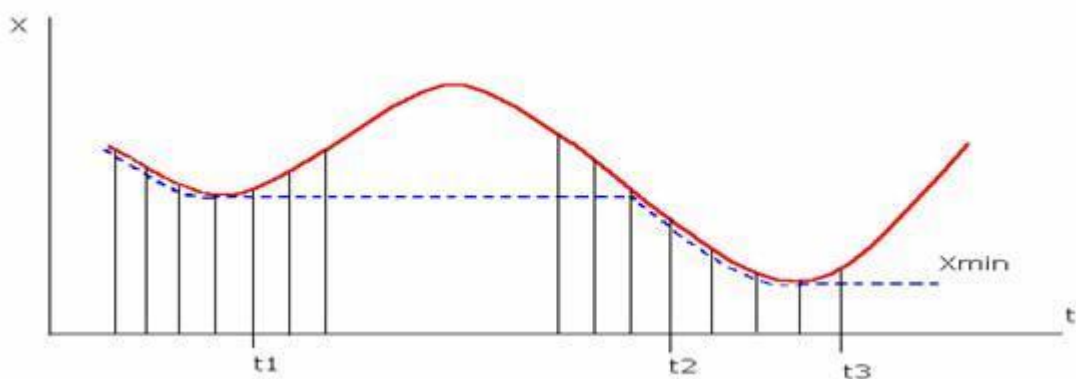


Рисунок 3.7 - Приклад роботи пристрою



У випадку, коли виконується нерівність (тобто поточний сигнал  $X_i$  менший за попередній  $X_{i-1}$ ), це свідчить про виявлення ще меншого значення (див. рисунок 3.7), і на середній вхід перемикача (див. рисунок 3.7) подається одиниця. Він переходить у верхнє положення, і пошук продовжується.

На момент часу  $t_1$  (див. рисунок 3.7), коли поточне значення  $X_i$  стає більшим за попереднє  $X_{i-1}$  (тобто уникнено локального мінімуму), нерівність не виконується, на вхід перемикача подається нуль, і він переходить у нижнє положення. При цьому блок пам'яті замикається сам на себе і запам'ятовує останнє знайдене мінімальне значення. Далі наступне поточне значення  $X(t)$  порівнюється зі знайденим локальним мінімумом.

На момент часу  $t_2$  поточне значення стає меншим за раніше знайдений мінімум, нерівність знову виконується, і перемикач переходить у верхнє положення. Пошук триває, поки не буде виявлено наступний мінімум у момент часу  $t_3$ .

Пристрій для пошуку максимуму працює аналогічно (див. рисунки 3.8, 3.9):

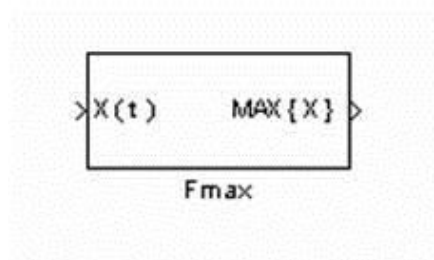


Рисунок 3.8 - Пристрій для пошуку максимуму

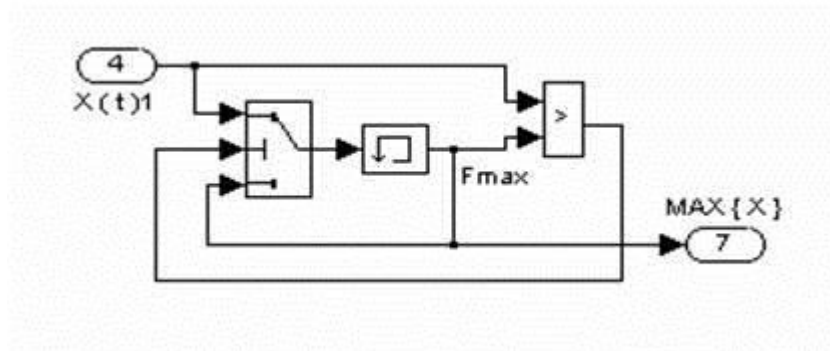


Рисунок 3.9 - Пристрій для пошуку максимуму

### 3.2 Робота з приладом

Прилад обладнаний двома входами (див. рисунок 3.3):

- Перший вхід призначений для вхідного сигналу.
- Другий вхід призначений для сигналу на виході об'єкта.

Час вимірювання повинен бути обраний так, щоб перехідні процеси в об'єкті завершилися і утворилися стаціонарні коливання.

Прилад має п'ять виходів:

1. Амплітуда вхідного (випробувального) сигналу.
2. Зміщення на вході  $X_{0вх}$ .
3. Модуль амплітудно-частотної характеристики (АФЧХ), представлений символом  $|W|$ .
4. Амплітуда вихідного сигналу.
5. Зміщення на виході  $X_{0вих}$ .

Значення зсувів будуть використовуватися як вихідні дані для описаного нижче приладу.

### 3.3 Опис фазочастотоміра гармонійних коливань

Це віртуальний пристрій, розроблений для вимірювання фазо-частотних характеристик будь-яких динамічних об'єктів, які розглядаються як "чорна скринька".

Принцип роботи приладу полягає в тому, щоб визначити фазу, достатньо виміряти інтервали часу  $L$  і  $D$  (див. графіки на рисунку 3.11).

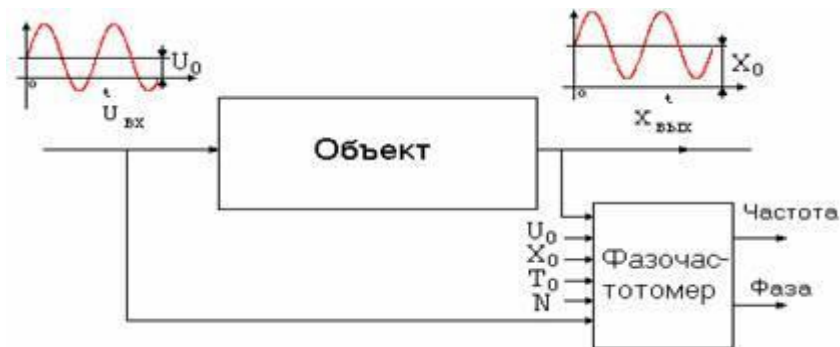


Рисунок 3.10 - Схема приладу

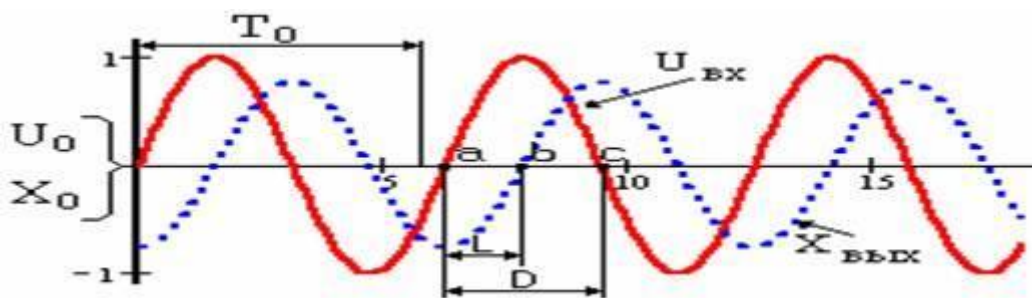


Рисунок 3.11 - Залежність фази від часу

### 3.4 Етапи вимірювання

Спочатку зчитується інтервал часу з вхідного сигналу, за яким встановлюються стаціонарні коливання (інтервал  $T_0$  на малюнку 3.11). Потім вимірюється тривалість  $D$  одного напівколивання вхідного сигналу та зсув  $L$  вихідних коливань в порівнянні з вхідними (див. малюнок 3.11).

Після отримання значень  $L$  та  $D$  обчислюється фаза, а саме:  $\Phi = (L/2D) \cdot 2\pi$ . (Знання постійних зсувів вхідних  $U_0$  і вихідних  $X_0$  коливань

необхідне для обчислення фази, оскільки на практиці коливання відбуваються щодо не нульового рівня).

Після отримання значень  $L$  та  $D$  прилад обчислює частоту коливань:  
 $f_{\text{рад}} = \pi/D$  або  $f_{\text{герц}} = 1/(2*D)$ .

### 3.5 Влаштування та функціонування фазочастотоміра

Вид приладу представлений на рисунку 3.12 і 3.13. У нього є 6 входів та 5 виходів.



Рисунок 3.12 - Фазочастотомір

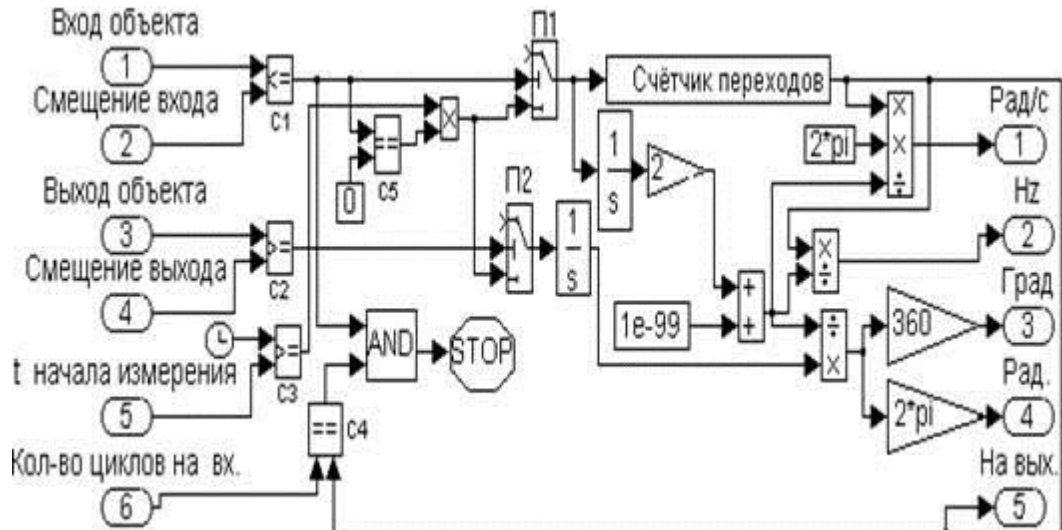


Рисунок 3.13 - Фазочастотомір

Фазочастотомір використовує логічні елементи для порівняння постійного зміщення ( $U_0$ ) щодо самого сигналу (рисунок 13, елемент C1). Якщо умова виконується, то "крива вниз", якщо ні - "крива вгору" (на виході 1 або 0 відповідно).

При вимірюванні величини  $D$  прилад розглядає позитивну напівхвилю (але тільки після інтервалу  $T_0$ ).

Блок СЗ використовується спільно з елементом Clock для порівняння поточного часу моделювання зі значенням  $T_0$  і далі для видачі сигналу на початок вимірювання відрізків  $L$  і  $D$ .

На точність вимірювання  $L$  і  $D$  впливає як час  $T_0$  (закінчення перехідних процесів), а й нульові значення постійних складових  $U_0$  і/або  $X_0$ .

Для цього, значення сигнал порівнюється з величиною зміщення  $U_0$  (або  $X_0$ ) і, як тільки  $U_{вх}$  виявиться в точці "а", тобто. на рівні  $U_0$  (або в точці "b", тобто на рівні  $X_0$  для  $X_{вх}$ ), відразу починається обчислення інтервалів  $L$  і  $D$ . Для вхідного сигналу вимірюється лише тривалість  $D$  напівколивання, оскільки сигнал симетричний.

Власне вимірювання інтервалів  $L$  та  $D$  здійснюється на інтеграторах.

Константа  $1e^{-99}$  дозволяє уникати ситуацій "ділення на нуль" у початковий період  $[0-T_0]$ .

Важливим елементом приладу є блок під назвою Лічильник переходів (рис. 3.14).



Рисунок 3.14 - Лічильник переходів

Він використовується для обчислення кількості перетинів вхідним сигналом рівня  $U_0$  за час вимірювання, використовуючи стандартні елементи пам'яті та визначення перетинів (рис. 3.14).

Блок Hit Crossing є датчиком, який виявляє перетин вхідним сигналом встановленого рівня. Вхідний сигнал є прямокутними коливаннями в

діапазоні  $[0 - 1]$ . Один повний період може мати один "підйом" та один "спуск". Встановивши значення "підйому" у блоку Hit Crossing, отримуємо сигнал про початок нового періоду коливань. Кількість переходів запам'ятовується блоком Memory, який зберігає попереднє значення.

Прилад розрахований на те, щоб підраховувати кількість переходів  $N$  за час вимірювання  $L$  і  $D$ , і здійснює сумування цих значень. Це корисно для усереднення результатів вимірювань за кілька періодів, особливо при наявності сигналів спотворень і шумів.

Організація завершення процесу вимірювання використовує логічний елемент  $C4$ , який порівнює встановлену кількість циклів  $N$  з кількістю вже виявлених переходів "знизу вгору". Сигнал із виходу  $C4$  може використовуватися для зупинки обчислень, але потрібно уникати припинення вимірювань на останньому циклі, коли позитивне напівколивання щойно почалося і виміри ще не завершилися. Для цього використовується логічний елемент  $AND$ , на другий вхід якого подається сигнал з виходу елемента  $C1$ , що дозволяє "затримати" зупинку вимірювань до тих пір, поки вхідний сигнал не перейде в негативну область.

Нагадаємо, що схема на рисунку 3.13 обмежує вимірювання фази до  $0-180^\circ$ . Деякі модифікації можуть бути внесені для розширення діапазону до  $360^\circ$ .

## 4 МОДЕЛЬ СИСТЕМИ, РЕАЛІЗОВАНА У ПІДСИСТЕМІ SIMULINK

Побудова моделі в Simulink включає переміщення відповідних блоків з бібліотек Simulink у вікно створеної моделі та їх з'єднання за допомогою функціональних зв'язків. У контексті опису задачі про два баки за стандартами UML можна виділити підсистему System, що є складовим блоком. Ця підсистема містить у собі функціональну схему з діаграмою Stateflow (представленою блоком Controller, який є екземпляром стандартного блоку Chart і описує поведінку контролера), а також складовий блок Tank\_System\_Block, які з'єднані відповідними функціональними зв'язками (див. рисунок 4.2).

У блоці System також присутній годинник Clock, який подає системний час з Simulink в діаграму Stateflow. Це виконано для синхронізації внутрішнього системного часу у Stateflow і Simulink, оскільки блоки Stateflow мають переходи, які ініціюються закінченням деяких тимчасових інтервалів.

На рисунку 4.2 зображено зв'язки між блоками Clock і Controller, а також чотири блоки Hit Cross, які приєднані до відповідних зв'язків блоку Controller і виходу h2 блоку Tank\_System\_Block. Використання блоків Hit Cross необхідне для правильного виконання переходів у діаграмі Stateflow, що включена у модель Simulink. У цій діаграмі відбувається безперервне інтегрування.

Блок System не має входів і має два виходи - h1 і h2. Ці виходи об'єднуються у вектор (h1, h2) за допомогою стандартного блоку Mux, а потім цей вектор подається на вивід блоку Scope для візуалізації результатів.

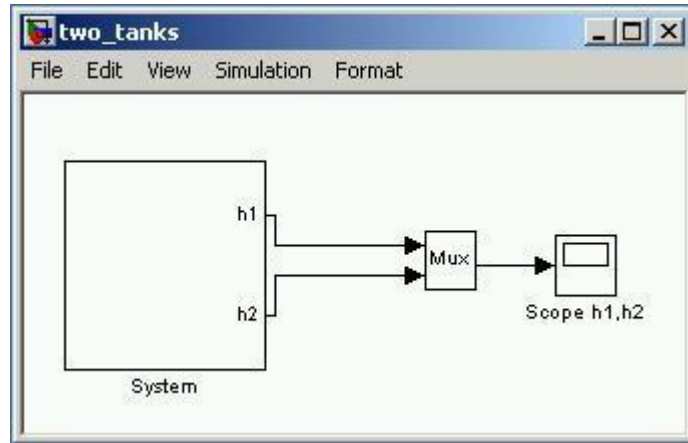


Рисунок 4.1 - Підсистема System

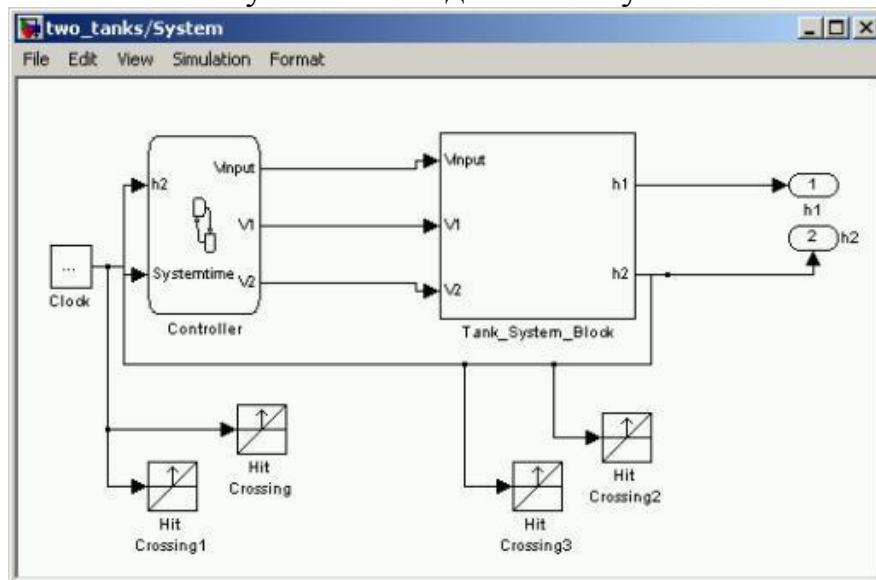


Рисунок 4.2 - Блок Tank System Block

У свою чергу, блок Tank\_System\_Block включає в себе функціональну схему, яка складається з блоку Tank\_System, який має три входи, з'єднані з відповідними виходами блоку Chart, на які подаються сигнали контролера, а також два виходи - h1 і h2. Крім того, в цьому блоку присутні пара екземплярів стандартних блоків Const, Relational Operator та Stop Simulation.

Елемент Const використовується для встановлення постійних значень, які використовуються в процесі моделювання. Relational Operator використовується для виконання порівнянь, наприклад, для перевірки на переповнення баків. Stop Simulation відповідає за зупинку симуляції при виконанні певних умов.



Зв'язки між цими блоками реалізовані за допомогою функціональних зв'язків, які визначають потік даних та контролюють взаємодію між різними частинами моделі.

Блок Tank\_System\_Block має три входи, на які подаються сигнали від контролера, і два виходи - h1 та h2, які відображають висоту рідини в баках.

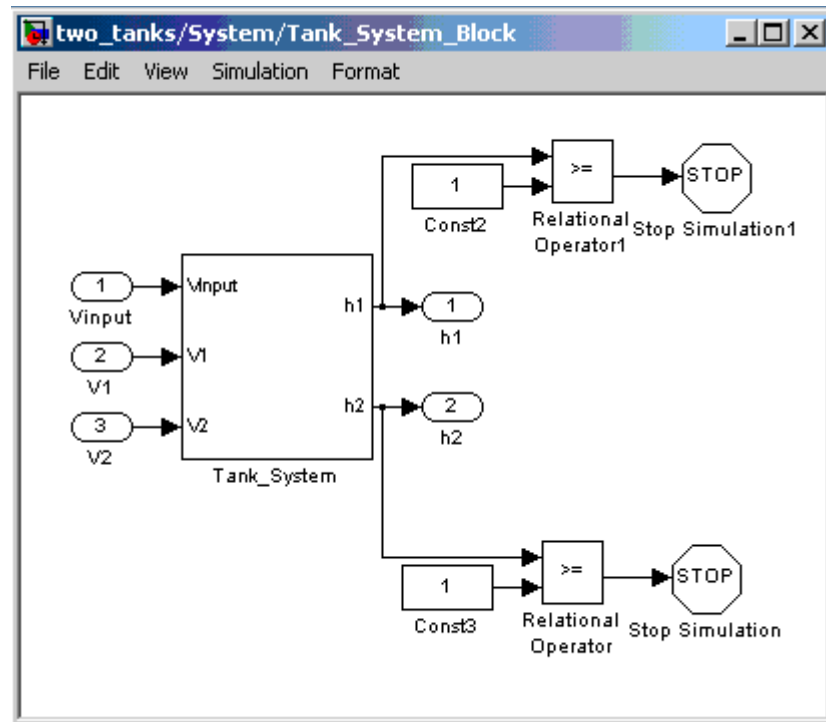


Рисунок 4.3 - Функціональна схема Tank System Block

Складовий блок Tank\_System включає в себе функціональну схему, яка складається із складових блоків Two\_Tanks, Vin\_Control, K1\_Control та K2\_Control, що взаємодіють між собою через функціональні зв'язки (рисунок 4.4). Подібно до попередніх блоків, Tank\_System має однакову кількість входів і виходів, із тими самими значеннями.

Блок Two\_Tanks, імітуючи систему з двома баками, взаємодіє з блоками управління (Vin\_Control, K1\_Control та K2\_Control) для визначення висоти рідини в баках. Величини Vin\_Control, K1\_Control і K2\_Control відображають керовані параметри системи, такі як вхідний об'єм рідини і

коефіцієнти контролю. Функціональні зв'язки між цими блоками визначають логіку взаємодії та потік даних в системі.

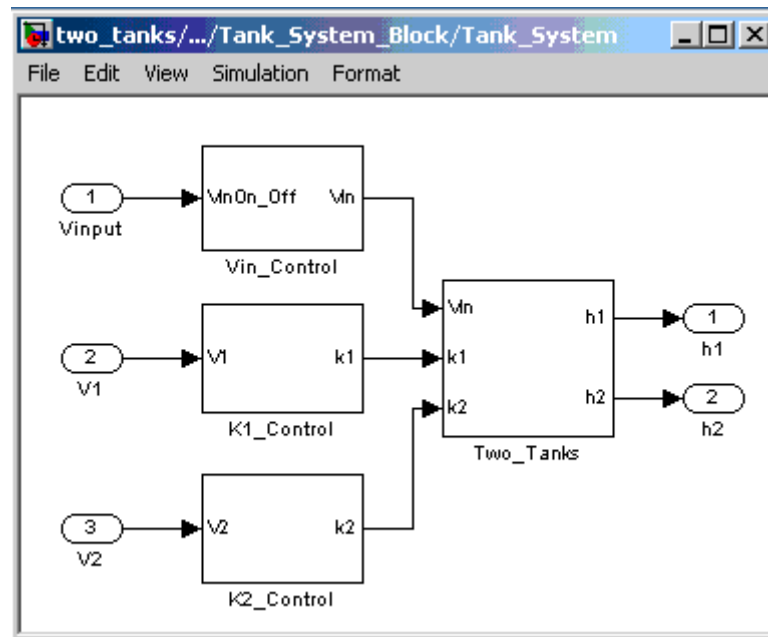


Рисунок 4.4 - Функціональна схема Tank System Block

Складовий блок `Vin_Control` включає функціональну схему, що складається зі стандартного блоку `Switch` та двох екземплярів стандартного блоку `Const`. Останні містять значення, які відповідають швидкості вхідного потоку води в систему двох баків при відкритому і закритому вхідному крані відповідно. Перемикання між цими значеннями відбувається в блоку `Switch` залежно від вхідного сигналу від контролера, і відповідне значення `Vin` подається на вихід. Блок `Vin_Control` має один вхід та один вихід (рисунок 4.5).

Складовий блок `Tank_System` містить у собі функціональну схему, яка складається із складових блоків `Two_Tanks`, `Vin_Control`, `K1_Control` та `K2_Control`, з'єднаних між собою функціональними зв'язками (рисунок 4.4).

Блок Tank\_System має таку ж кількість входів і виходів із тими самими значеннями, як і попередній блок.

У свою чергу, складовий блок Tank\_System\_Block містить у собі функціональну схему, що складається зі складового блоку Tank\_System (має три входи, з'єднані з відповідними виходами блоку Chart, на які подаються сигнали контролера, і два виходи - h1 і h2) та пара екземплярів стандартних блоків Const, Relational Operator та Stop Simulation (які виконують перевірку на переповнення баків), з'єднаних відповідними функціональними зв'язками (рисунок 4.3). Блок Tank\_System\_Block має три входи (на які подаються сигнали від контролера) та два виходи (h1 та h2).

Побудова моделі Simulink зводиться до переміщення за допомогою миші необхідних блоків з бібліотек Simulink у вікно створюваної моделі і з'єднання цих блоків між собою за допомогою функціональних зв'язків. Бібліотеки Simulink містять багато різноманітних функціональних блоків, які відображаються на екрані у вигляді піктограм. Використовуючи опис задачі про два баки в термінах UML, можна виділити в даному прикладі підсистему System (рисунок 4.1), що є складовим блоком, який містить у собі функціональну схему з діаграмою Stateflow (представлену блоком Controller, що є екземпляром стандартного блоку Chart і описує поведінку контролера ) та складовий блок Tank\_System\_Block, з'єднані відповідними функціональними зв'язками (рисунок 4.2). Так само в блоці System присутні годинник Clock,

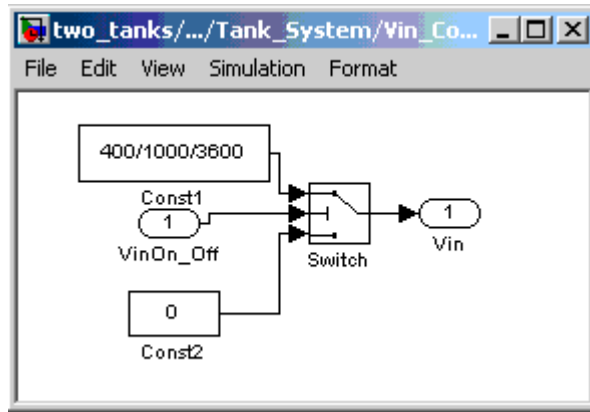


Рисунок 4.5 - Блок Vin Control

Складові блоки  $K1\_Control$  і  $K2\_Control$  включають ідентичні функціональні схеми, відмінні лише за значенням коефіцієнтів у рівнянні, представленому в стандартному блоку  $fcp$ . Функціональна схема складається з двох екземплярів стандартного блоку  $Switch$  (один необхідний для перемикання між положеннями кранів  $V1/V2$  (відкрито/закрито), а інший для відстеження ситуації, коли параметр  $p$  досягає значення 80 і відповідного перемикання між значеннями функції  $K(p)$ ), трьох екземплярів стандартного блоку  $Const$  (які містять значення 1 і -1, що відповідають положенню крана  $V1/V2$  (відкрито/закрито) та значення 0 для функції  $K(p)$  у ситуації, коли  $p \geq 80$ ), стандартного блоку  $Integrator$  (інтегрує значення від перемикача в межах від 0 до 80 з початковим значенням інтегрованої величини 80) і стандартного блоку  $f(u)$ , в якому обчислюється значення функції  $K1(p)/K2(p)$ . Перемикання в блоці  $Switch$  відбувається в залежності від значення, що надходить на вхід блоку  $K1\_Control/K2\_Control$  від контролера. Відповідне значення  $k1/k2$  подається на вихід. Блоки  $K1\_Control/K2\_Control$  мають по одному входу та виходу (на рисунку 4.6 представлений блок  $K1\_Control$ , блок  $K2\_Control$  ідентичний):

У функціональних схемах складових блоків  $K1\_Control$  і  $K2\_Control$  використовуються однакові структури, відрізняючись лише значеннями коефіцієнтів у рівнянні, що представлено у стандартному блоку  $fcp$ . Кожна

функціональна схема включає два екземпляри стандартного блоку Switch. Перший відповідає за перемикання між положеннями кранів V1/V2 (відкрито/закрито), а другий відстежує ситуацію, коли параметр  $p$  досягає значення 80, і відбувається відповідне перемикання між значеннями функції  $K(p)$ ). Також включено три екземпляри стандартного блоку Const. Перші два містять значення 1 і -1, які відображають положення крана V1/V2 (відкрито/закрито), а третій містить значення 0 для функції  $K(p)$  у випадку, коли  $p \geq 80$ .

Додатково, використовується стандартний блок Integrator, який інтегрує значення від перемикача в межах від 0 до 80 з початковим значенням інтегрованої величини 80. У функціональному блоці  $f(u)$  обчислюється значення функції  $K1(p)/K2(p)$ . Перемикання в блоку Switch відбувається в залежності від значення, що надходить на вхід блоку K1\_Control/K2\_Control від контролера. Відповідне значення  $k1/k2$  подається на вихід. Блоки K1\_Control/K2\_Control мають по одному входу та виходу. На рисунку 4.6 наведено блок K1\_Control, але блок K2\_Control аналогічний.

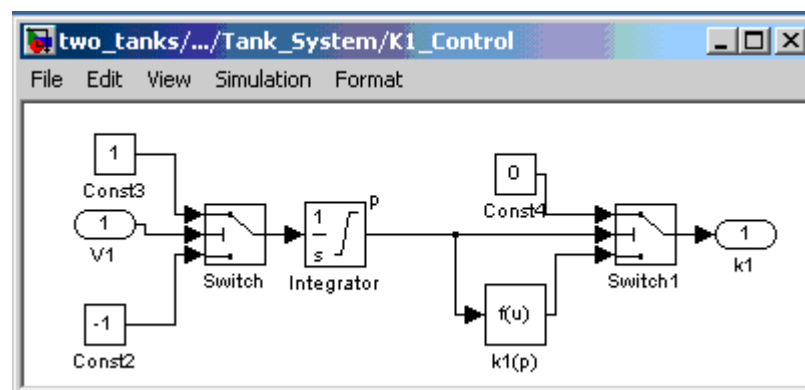


Рисунок 4.6 - Блок K1 Control

Складовий блок Two\_Tanks включає функціональну схему, що складається з інших складових блоків, таких як Vout\_calc, V12\_calc1 і V12\_calc2 (відповідальні за розрахунок Vout і V12), стандартного блоку Switch, двох екземплярів стандартного блоку Mux (які об'єднують в собі

виходи блоку Vin\_Control), V12 і Vout), двох екземплярів стандартного блоку fcn (які виконують обчислення похідних h1 і h2), та двох екземплярів стандартного блоку Integrator, які виводять значення h1 і h2.

У блока Two\_Tanks реалізовано перемикання в блоці Switch в залежності від значення h2, що подається на вхід блоку Switch від інтегратора. Відповідно до цього значення підключається один із блоків, які обчислюють V12. Блок Two\_Tanks має три входи, на які подаються значення Vin, k1 та k2, і два виходи - h1 та h2 (див. рисунок 4.7).

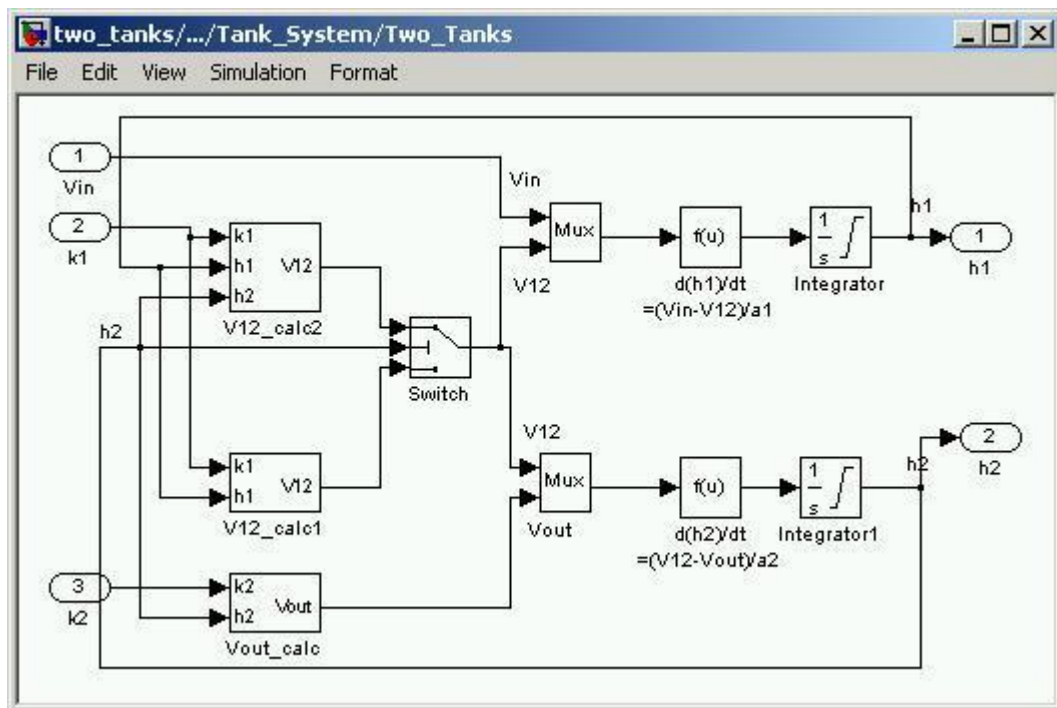


Рисунок 4.7 - Блок Two Tanks

Складовий блок Vout\_calc включає функціональну схему, що складається із стандартного блоку Mux. Цей блок об'єднує в один вектор значення k2 і h2, які подаються на вхід блоку Vout\_calc. Сформований вектор подається на вхід стандартного блоку fcn, де відбувається обчислення вихідної величини Vout. Блок Vout\_calc має два входи та один вихід (див. рисунок 4.8).

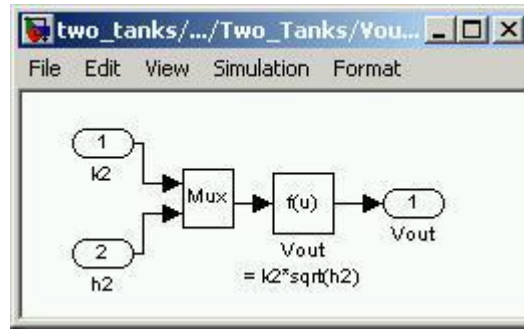


Рисунок 4.8 - Блок Vout calc

Складовий блок V12\_calc1 включає функціональну схему, яка складається із стандартного блоку Mux. Цей блок об'єднує в один вектор значення k1 і h1, які подаються на вхід блоку V12\_calc1. Сформований вектор подається на вхід стандартного блоку fcn, де відбувається обчислення вихідної величини V12. Блок V12\_calc1 має два входи та один вихід (див. рисунок 4.9).

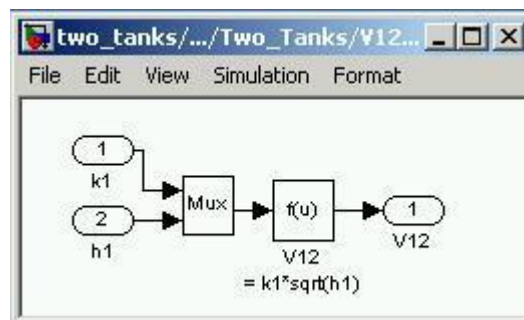


Рисунок 4.9 - Блок V12 calc1

Складовий блок V12\_calc2 включає функціональну схему, яка складається зі стандартного блоку Mux. Цей блок об'єднує в один вектор значення k1, h1 і h2, які подаються на вхід блоку V12\_calc2. Сформований вектор подається на вхід стандартного блоку fcn, де відбувається обчислення вихідної величини V12. Блок V12\_calc2 має три входи та один вихід (див. рисунок 4.10)

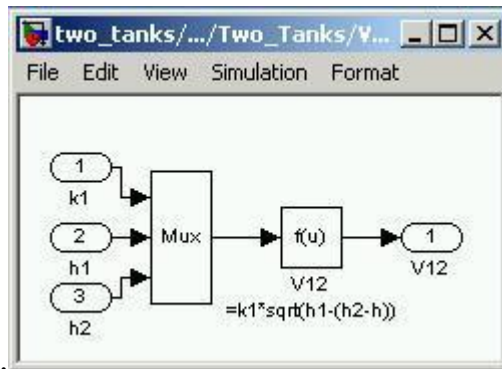


Рисунок 4.10 - Блок V12 calc2

Діаграма Stateflow, представлена екземпляром стандартного блоку Chart, практично ідентична відповідній діаграмі станів UML (див. рис. 4.11). У цій діаграмі замість внутрішнього часу  $t$  використовується системний час Simulink, який подається на вхід блоку у вигляді вхідної змінної Systemtime.

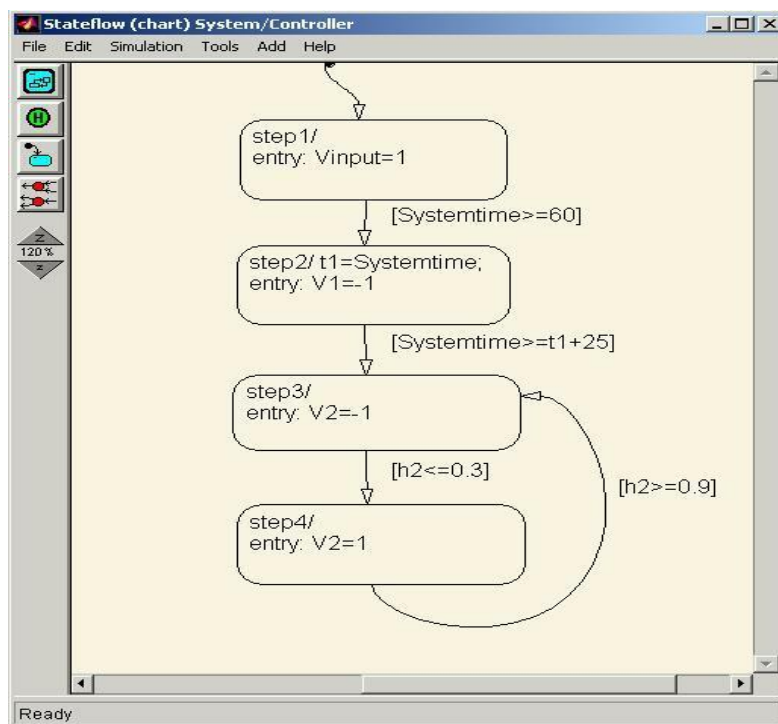


Рисунок 4.11 - Діаграма Stateflow



## 5 ОСНОВНІ ПРИЙОМИ ПІДГОТОВКИ ТА РЕДАКТУВАННЯ МОДЕЛІ

### 5.1. Додавання текстових написів

Для збагачення візуального сприйняття моделі рекомендується використовувати текстові елементи. Додавання текстового напису в модель можна виконати, вказавши місце розташування напису та використовуючи подвійний клацок лівою кнопкою миші. Після цього з'явиться прямокутна рамка разом із курсором для введення тексту.

Також можливе змінення підписів блоків моделі за допомогою аналогічного процесу. На малюнку 5.1 демонструється додавання текстового напису та зміна підпису в блоку передавальної функції. Важливо врахувати, що в даній версії програми Simulink 4 не підтримується використання кирилических шрифтів, і їх застосування може викликати проблеми, такі як некоректне відображення, обрізання тексту чи навіть неможливість відкриття моделі після збереження. Тому важливо утримуватися від використання російської мови для текстових елементів у даній версії Simulink.

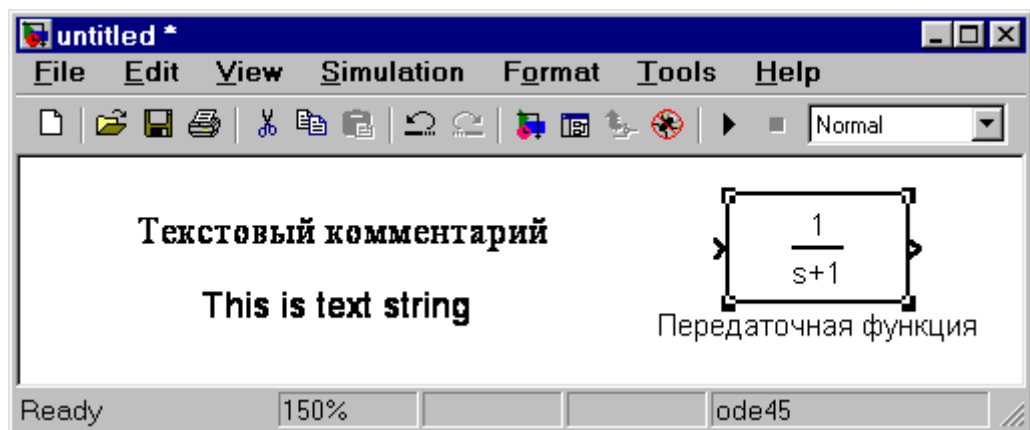


Рисунок 5.1 - Текстовий напис і зміна напису в Transfer Function

## 5.2 Виділення об'єкта

Для виконання будь-якої операції з елементом моделі, таким як блок, з'єднувальна лінія або текстовий напис, спочатку його потрібно виділити.

Процес виділення об'єктів найлегший за допомогою миші. Достатньо встановити курсор миші на об'єкті і клацнути лівою кнопкою миші. Вибраний об'єкт буде позначений маркерами на його кутах (див. рис. 5.1). Крім того, можна виділити кілька об'єктів одночасно. Для цього треба встановити курсор миші біля групи об'єктів, натиснути ліву кнопку миші та, утримуючи її, рухати мишу. З'явиться пунктирна рамка, розмір якої буде змінюватися під час переміщення миші. Всі об'єкти, які опиняться всередині цієї рамки, будуть виділені. Також можна виділити всі об'єкти за допомогою команди "Edit/Select All". Після виділення об'єкта його можна скопіювати або перемістити в буфер обміну, витягнути з буфера, а також видалити, використовуючи стандартні методи роботи в програмах для Windows.

## 5.3 Копіювання та переміщення об'єктів у буфер проміжного зберігання

Для копіювання об'єкта в буфер необхідно попередньо виділити, а потім виконати команду Edit/Сору або скористатися інструментом на панелі інструментів.

Для вирізання об'єкта в буфер його необхідно попередньо виділити, а потім виконати команду Edit/Cut або скористатися інструментом на панелі інструментів. При виконанні даних операцій слід мати на увазі, що об'єкти розміщуються у власному буфері MATLAB і недоступні з інших програм. Використання команди Edit/Сору model to Clipboard дозволяє помістити

графічне зображення моделі в буфер Windows і, відповідно, робить його доступним для інших програм.

Копіювання можна виконати і так: натиснути праву клавішу миші, і не відпускаючи її, перемістити об'єкт. При цьому буде створено копію об'єкта, яку можна перемістити в потрібне місце.

#### **5.4 Вставлення об'єктів із буфера проміжного зберігання**

Щоб вставити об'єкт з буфера, спочатку вкажіть місце вставки, клацнувши лівою кнопкою миші у відповідному місці. Після цього використовуйте команду Edit/Paste або скористайтеся відповідним інструментом на панелі інструментів.

#### **5.5 Видалення об'єктів**

Для вилучення об'єкта спочатку виділіть його, а потім використовуйте команду Edit/Clear або клавішу Delete на клавіатурі. Важливо враховувати, що команда Clear видаляє об'єкт, не зберігаючи його в буфері обміну. Проте цю операцію можна скасувати за допомогою команди File/Undo.

#### **5.6 З'єднання блоків**

Для створення з'єднання між блоками встановіть курсор на вихідний порт одного з блоків. Курсор трансформується у великий хрест із тонкими лініями (рисунки 5.2). Утримуючи ліву кнопку миші, перенесіть курсор до вхідного порту іншого блоку. Курсор при цьому стане хрестом із тонкими здвоєними лініями (рисунки 5.3). Після створення з'єднання відпустіть ліву кнопку миші.

Як підтвердження успішного з'єднання, ви побачите жирну стрілку біля вхідного порту блоку. Для виділення лінії використовуйте одинарне клацання лівою кнопкою миші. Чорні маркери в вузлах сполучної лінії вказують, що лінія виділена.

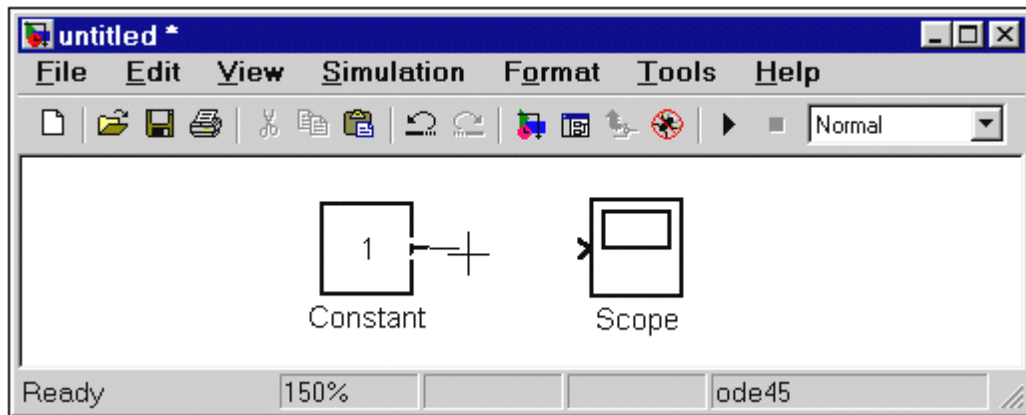


Рисунок 5.2 – Початок створення з'єднання

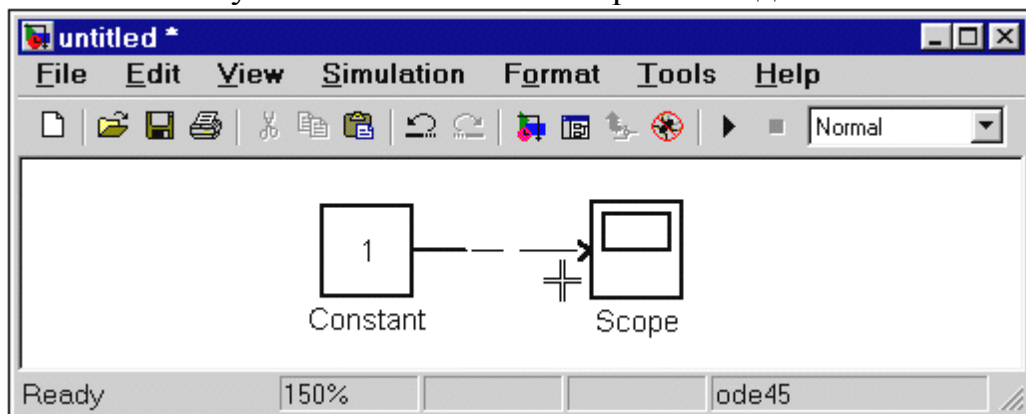


Рисунок 5.3 - Завершення створення з'єднання

Побудова петлі лінії з'єднання виконується за допомогою переміщення блоку. Спочатку виділіть лінію з'єднання, а потім перемістіть потрібну її частину. Процес зображений на рисунку 5.4.

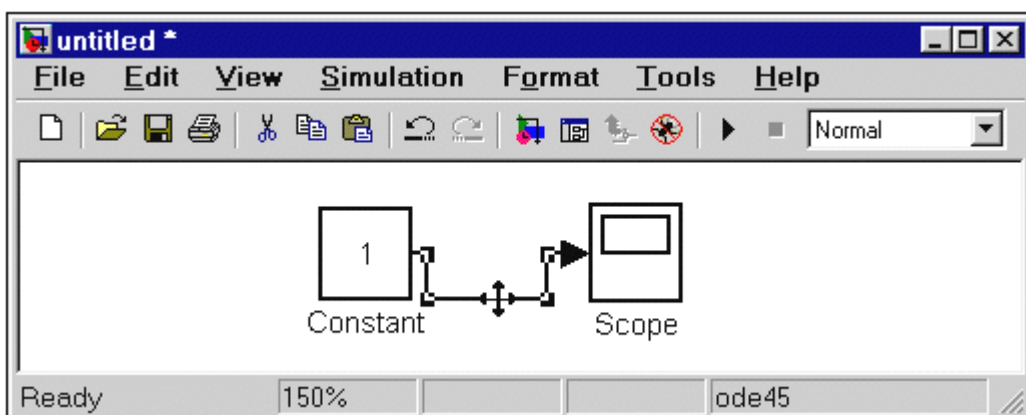


Рисунок 5.4 - Створення петлі в з'єднувальній лінії

### 5.7 Зміна розмірів блоків

Для зміни розміру блоку спочатку його слід виділити, а потім встановити курсор миші на один із маркерів в його кутах. Після того, як курсор перетвориться на двосторонню стрілку, натисніть ліву клавішу миші і розтягніть (або стисніть) зображення блоку. Процес зображений на рисунку 5.5. Важливо відзначити, що розміри написів блоку залишаються незмінними.

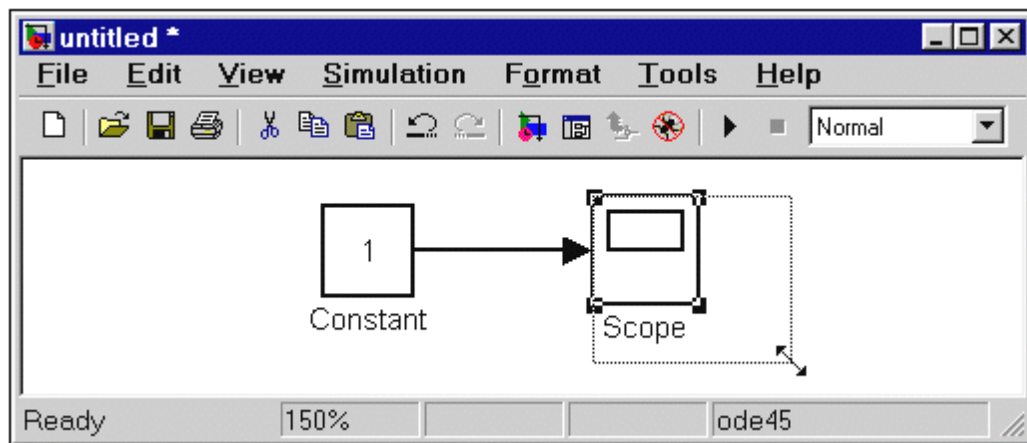


Рисунок 5.5 - Зміна розміру блока

### 5.8 Переміщення блоків

Будь-який блок моделі можна перемістити, виділивши його та пересунувши, утримуючи ліву клавішу миші. Якщо до входів і виходів блоку підведені сполучні лінії, то вони не розриваються, лише скорочуються або збільшуються в довжині. У з'єднання можна вставити блок, що має один вхід і один вихід, розташувавши його у потрібному місці сполучної лінії.

### 5.9 Використання команд Undo та Redo

Під час ознайомлення з програмою користувач може випадково виконати дії, які здаються неповерненними (наприклад, видалення важливої частини моделі чи копіювання). У таких випадках можна скористатися командою "Undo" – для скасування останньої виконаної операції. Цю команду можна викликати як за допомогою кнопки на панелі інструментів вікна моделі, так і через меню "Edit". Щоб відновити скасовану операцію, використовується команда "Redo" (або відповідний інструмент).

### 5.10 Форматування об'єктів

У розділі "Format" меню (а також у контекстному меню, яке можна викликати, натиславши праву кнопку миші на об'єкті), доступний набір команд для форматування блоків. Команди форматування розділені на декілька груп:

#### 1. Зміна відображення написів:

- **Font** - форматування шрифту написів та текстових блоків.
- **Text alignment** - вирівнювання тексту у текстових написах.
- **Flip name** - переміщення підпису блоку.
- **Show/Hide name** - відображення або приховування підпису.

#### 2. Зміна кольорів відображення блоків:

- **Foreground color** - вибір кольору ліній для виділених блоків.
- **Background color** - вибір кольору фону для виділених блоків.
- **Screen color** - вибір кольору фону для вікна моделі.

#### 3. Зміна положення блоку та його вигляду:

- **Flip block** - дзеркальне відображення вертикальної осі симетрії.
- **Rotate block** - поворот блоку на 90 градусів за годинниковою стрілкою.
- **Show drop shadow** - показ тіні від блоку.

- **Show port labels** - показ міток портів.

#### 4. Інші установки:

- **Library link display** - показ зв'язків із бібліотеками.
- **Sample time colors** - вибір кольору індикації часу.
- **Wide nonscalar lines** - збільшення/зменшення ширини нескалярних ліній.
- **Signal dimensions** - показ розмірності сигналів.
- **Port data types** - показ даних про тип портів.
- **Storage class** - клас пам'яті (параметр, який встановлюється під час роботи Real-Time Workshop).
- **Execution order** - виведення порядкового номера блоку у послідовності виконання.

### 5.11 Обмін даними між середовищем Matlab і S-моделлю

Робочий простір середовища MATLAB доступний для використання в S-моделі. Це означає, що якщо у вікнах настроювання блоків S-моделі використовуються ідентифікатори замість числових значень для параметрів, а відповідні значення цих ідентифікаторів вже визначені у робочому просторі, то ці значення автоматично передаються відповідним блокам S-моделі. Таким чином, для зручного змінювання параметрів блоків S-моделі (наприклад, у діалоговому режимі) можна виконати наступне:

1. У вікнах настроювання блоків S-моделі вказати ідентифікатори (імена) замість чисел для параметрів.
2. Організувати, наприклад, програмно у середовищі MATLAB присвоєння числових значень цим ідентифікаторам та, за потреби, їх змінювання у діалоговому режимі.

3. Після присвоєння числових значень усім ідентифікаторам (наприклад, через виклик відповідного скрипту MATLAB), запустити моделювання S-моделі.

Деякі інструменти обміну даними вже були описані раніше, такі як блоки From Workspace і To Workspace стандартної бібліотеки Simulink. Перший призначений для включення сигналів, попередньо обчислених та записаних у робочому просторі MATLAB (наприклад, під час обчислень у середовищі MATLAB), у процес моделювання S-моделі. Другий дозволяє записати результати моделювання S-моделі у робочий простір середовища MATLAB.

Для запису отриманих під час моделювання значень у робочий простір, можна використовувати блок To Workspace. Підключивши вхід цього блоку до сигналу, який потрібно записати, і вказавши в полі Variable name (Ім'я змінної) ім'я, під яким цей процес слід зберегти у робочому просторі системи MATLAB. При цьому відповідні моменти модельного часу не будуть записані.

Щоб використовувати дані, записані у робочому просторі, у S-моделі, вставте блок From Workspace у схему S-моделі, з'єднайте його вихід з одним із входів інших блоків, розкрийте вікно налаштування блоку (рис. 5.6) і в полі введення Data вкажіть вектор, що складається з двох імен – імені масиву значень аргументів (моментів часу, у які визначено цей процес) і імені масиву значень процесу при вказаних значеннях аргументу, наприклад, [T, D]. У цьому випадку значення масиву T робочого простору будуть використовуватися в S-моделі як значення модельного часу, а значення вихідного сигналу блоку при моделюванні у моменти часу, що відповідає записаним у масиві T, буде дорівнювати значенням, записаним у масиві D. Якщо реальні значення моментів часу при моделюванні не збігаються з записаними у масиві T, буде виконана лінійна інтерполяція значень масиву



D, що відповідають попередньому і наступному значенням моментів часу у масиві T.

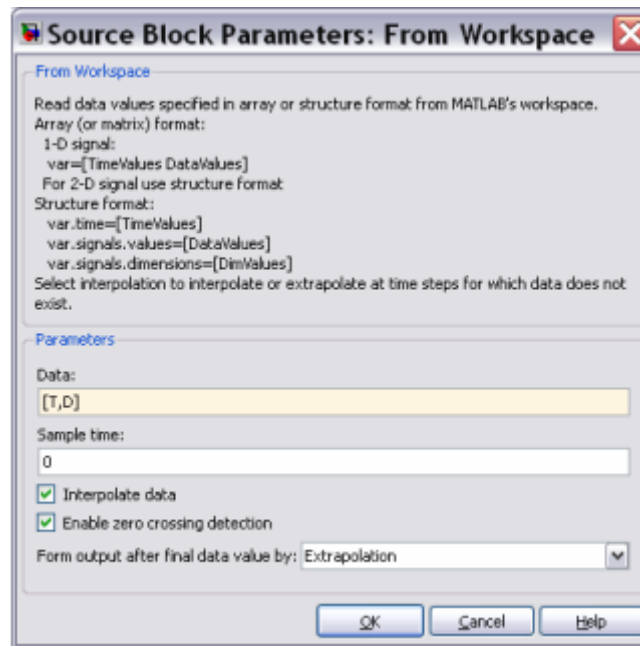


Рис. 5.6. Вікно налаштувань блоку From Workspace

Але існує більш прямий спосіб виконання вищезазначених операцій без використання зазначених блоків. Для того щоб підключити визначений у MATLAB процес до S-моделі як вхідний, можна скористатися механізмом включення портів входу і виходу, виконавши наступні дії:

1. Вставте блок вхідного порту In у блок-схему S-моделі та з'єднайте його з одним з блоків S-моделі.
2. У вікні S-моделі викличте команду Simulation > Configuration Parameters, щоб відкрити вікно Configuration Parameters, в якому виберіть команду Data Import/Export (Імпорт-експорт даних).
3. В області Load from workspace (Завантажити з робочого простору) встановіть прапорець Input (Вхід) і в полі праворуч введіть ім'я, яке складається з імені вектора значень аргумента і імені вектора значень вхідного сигналу при цих значеннях аргумента, наприклад: [t, u].

4. Встановіть значення цих векторів у MATLAB, наприклад, так:  $t = (0 : 0.1:1)'$ ;  $u = [\sin(t), \cos(t), 4*\exp(t)]$ .
5. Запустіть S-модель для моделювання.

Щоб вивести деякі сигнали, які формуються в S-моделі, у робочий простір MATLAB, виконайте наступні дії:

1. Вставте блоки вихідного порту Out у блок-схему моделі та під'єднайте до них необхідні вихідні величини інших блоків.
2. У вікні S-моделі викличте команду Simulation > Configuration Parameters, щоб відкрити вікно Configuration Parameters (рис. 5.7), в якому виберіть команду Data Import/Export (Імпорт-експорт даних) .
3. В області Save to workspace (Зберегти у робочому просторі) встановіть прапорці Time і Output, а в поле праворуч введіть імена, під якими будуть записані значення часу і величин, що подаються на вихідні порти, у робочий простір. За замовчуванням ці імена є tout (для модельного часу) і yout (для даних з вихідних портів).

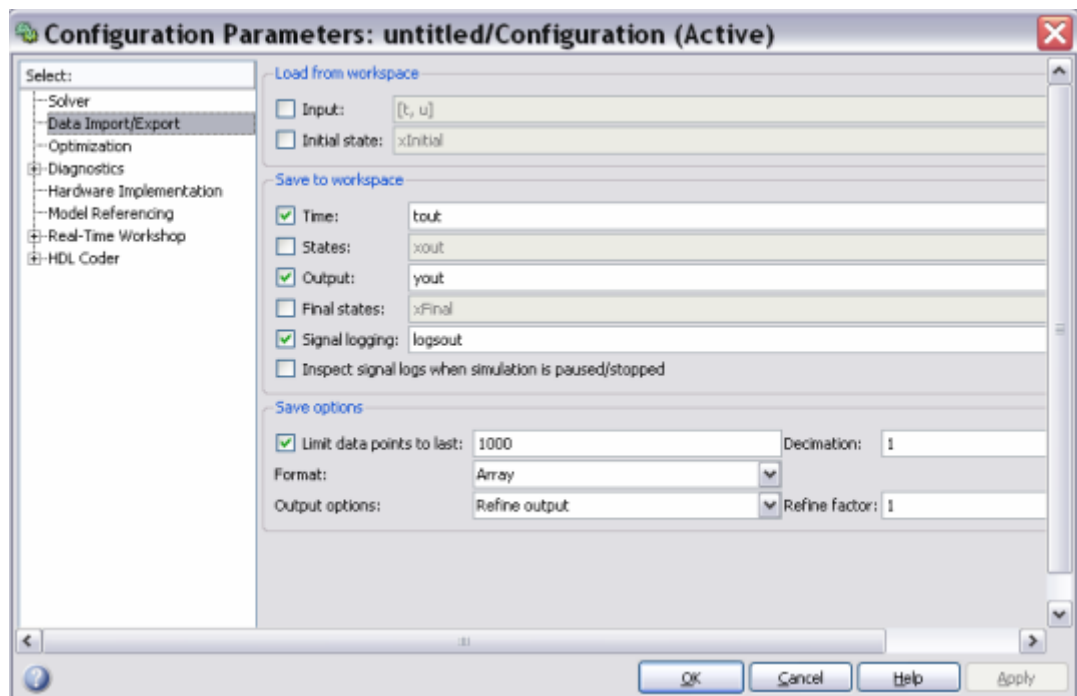


Рис. 5.7. Вкладення Data Import/ Export вікна Configuration Parameters

У цьому випадку значення модельного часу будуть записані у робочий простір у масив під ім'ям tout, а відповідні значення вихідних сигналів при цих значеннях часу - у стовпці матриці yout. Кожен стовпець матриці yout відповідає відповідному вихідному порту (наприклад, перший стовпець - процес, що подається на перший вихідний порт Out1, другий стовпець - процес, поданий на другий порт Out2, і так далі).

Після встановлення прапорця "Initial state" (Початковий стан) в області "Load from workspace" (Завантажити з робочого простору), можна ввести в S-модель початкові значення змінних стану системи. Встановивши прапорець "States" (Змінні стану) в області "Save to workspace" (Зберегти у робочому просторі), можна записати поточні значення змінних стану системи у робочий простір під ім'ям xout (або під іншим ім'ям, якщо його записати у поле праворуч від прапорця "State"). Нарешті, можна записати й кінцеві значення змінних стану у вектор xFinal, якщо встановити прапорець "Final state" (Кінцевий стан).

## 5.12 Запуск процесу моделювання S-моделі з середовища Matlab

Розглянемо засоби, які дозволяють запускати процес моделювання утворених S-моделей із програми Matlab. S-модель запускається на виконання, якщо у програмі Matlab (або у командному вікні Matlab) викликати процедуру sim:  $[t,x,y1,y2,\dots,yn] = \text{sim}(\text{model}, \text{timespan}, \text{options}, \text{ut})$ . Тут model – символічний рядок, що містить ім'я MDL-файлу, в якому записана відповідна S-модель; timespan – вектор, що складається з двох елементів – значень початкового і кінцевого моментів часу моделювання; options – вектор значень параметрів інтегрування, який встановлюється процедурою `simset options = simset('Властивість1',Значення1,'Властивість2',Значення2...)`; Процедура sim повертає наступні значення: t – вектор вихідних значень моментів модельного часу; x – масив (вектор) змінних стану системи; y1 –

першій стовпець матриці вихідних змінних системи (які надаються до вихідних портів) і т. д.

Змінювати параметри розв'язувача і процесу інтегрування у Matlab можна за допомогою функції `simset`, як це показано вище. У такий спосіб можна задати значення наступних властивостей розв'язувача:

- 'Solver' – назва розв'язувача; значення (вказується між двома апострофами) може бути однією з наступних: `ode45`, `ode23`, `ode1b`, `ode15s`, `ode23s` – для інтегрування з автоматично змінюваним кроком; `ode5`, `ode4`, `ode3`, `ode2`, `ode1` – для інтегрування з фіксованим кроком;

- 'RelTol' – відносна припустима похибка; значення може бути додатним скаляром; за замовчуванням встановлюється  $1e-3$ ;

- 'AbsTol' – абсолютна припустима похибка; значення може бути додатним скаляром; за замовчуванням встановлюється  $1e-6$ ;

- 'FixedStep' – фіксований крок (додатний скаляр);

- 'MaxOrder' – максимальний порядок методу (застосовується лише для методу `ode15s`); може бути одним з цілих чисел 1, 2, 3, 4; за замовчуванням дорівнює 5;

- 'MaxRows' – максимальна кількість рядків у вихідному векторі; невід'ємне ціле;

- 'InitialState' – вектор початкових значень змінних стану; - 'FinalStateName' – ім'я вектора, в який буде записуватися кінцеве значення вектора змінних стану моделі;

- 'OutputVariables' – вихідні змінні системи; за замовчуванням має значення {txy}; можливі варіанти tx, ty, xy, t, x, y; усі вони неявно вказують, які саме вихідні змінні не будуть виводитися.

### 5.13 Створення S-блоків з використанням програм Matlab

В системі Matlab передбачено засіб перетворення процедур, написаних мовами високого рівня, в блоки S-моделі за допомогою S-функцій. S-функція представляє собою програму, написану користувачем на мові Matlab або C, яка візуально відображається у вигляді блока в Simulink. Застосування S-функцій дозволяє вирішити ряд завдань, зокрема:

1. Створення нових блоків, які розширюють бібліотеку Simulink.
2. Використання опису модельованої системи у вигляді системи математичних рівнянь.
3. Включення раніше створених програм, написаних на мові Matlab або C, у S-модель.

Код S-функції має структуровану форму. У випадку, коли S-функція створюється на основі коду на мові Matlab, її структура визначається у файлі SfinTMPL.m, розташованому у папці TOOLBOX\SIMULINK\BLOCKS. Заголовок S-функції виглядає наступним чином: `function [sys, x0, str, ts] = <Ім'я S-функції>(t, x, u, flag{,<Параметри>})`

Стандартні аргументи S-функції включають поточне значення часу моделювання (**t**), поточне значення вектора змінних стану (**x**), поточне значення вектора вхідних величин (**u**), цілочислову змінну **flag**, що вказує на етап виконання S-функції, та **<Параметри>** - додаткові ідентифікатори.

У результаті виконання S-функції присвоюються значення таким змінним:

- **sys** – системна змінна, залежність вмісту якої визначається значенням **flag**.

- $\mathbf{x0}$  – вектор початкових значень змінних стану.
- $\mathbf{str}$  – символна змінна стану.
- $\mathbf{ts}$  – матриця, що містить інформацію про часову дискретизацію.

Головна процедура S-функції містить звернення до внутрішніх процедур відповідно до значення **flag**. Це включає ініціалізацію блока, обчислення правих частин диференціальних рівнянь, оновлення значень змінних стану, обчислення значень вихідних величин, визначення моменту часу для перетинання заданого рівня та завершення роботи блока.

Таким чином, використання S-функцій уможлиблює моделювання різних видів систем, включаючи як алгебраїчні, так і динамічні (неперервні або дискретні) ланки.

Щоб утворити S-блок на основі використання S-функції, виконайте наступні дії.

1. Напишіть текст S-функції, наприклад у виді M-файлу, користуючись файлом-шаблоном SfunTMPL.m.

2. Перетягніть стандартний блок S-функції (рис. 5.8) з поділу UserDefined Functions бібліотеки SIMULINK у вікно блок-схеми, в який буде створюватися новий S-блок.

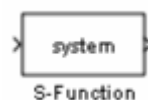


Рис. 5.8. Заготовка S- блоку S- функції

3. Двічі клацніть на зображенні блоку S-function. Це призведе до виникнення на екрані вікна його налаштування (рис. 5.9). Вікно містить поля введення S-function name (Ім'я S-функції), в яке вводиться ім'я файлу з написаним текстом S-функції, і S-function parameters (Параметри S-функції),

в яке вводяться ймення або значення параметрів блоку, вказаних у розділі М-файла, що містить написаний текст S-функції.

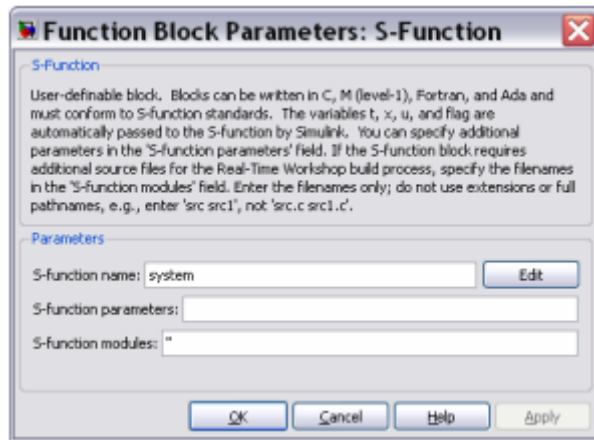


Рис. 5.9. Вікно налаштування блоку S-function

4. Введіть у вказані поля ім'я М-файла, у якому записаний текст S-функції, і список значень параметрів. Якщо, наприклад, у перше поле ввести ім'я S\_KA, а у друге – рядок J, Ug0, UgSk0

5. Клацніть мішкою на кнопці ОК. Якщо система виявить М-файл з вказаним ім'ям у папках які досяжні Matlab, вікно, що подане на рис. 5.10, зникне, а на зображенні блока у вікні блок-схеми виникне введене ім'я S-функції (точніше, написаного М-файла) (рис. 5.11).

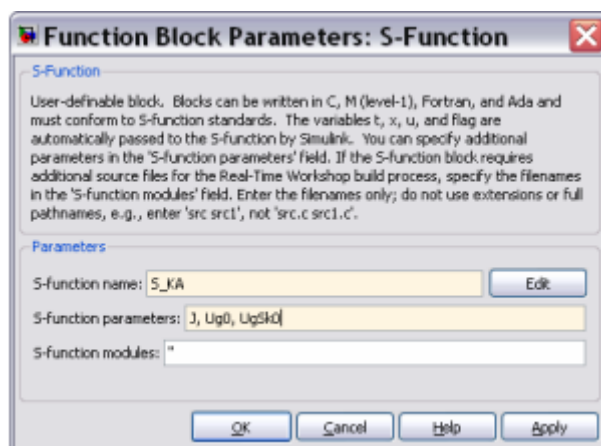


Рис. 5.10. Вікно S-function після введення даних

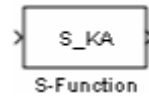


Рис. 5.11. Блок S-function після введення даних

S-блок, заснований на S-функції, що міститься у М-файлі **S\_KA.m**, тепер може бути використаний як звичайний S-блок у блок-схемі S-моделі. Призначенням цього блоку є обробка векторного сигналу **u** на вході. Результатом роботи блоку є векторний сигнал **y**, який формується за допомогою S-функції у внутрішній процедурі **mdlOutputs**.

Такий S-блок може бути використаний для моделювання різноманітних систем, включаючи ті, які описуються диференціальними або різницевими рівняннями. Він приймає векторний вхід **u** і генерує векторний вихід **y** відповідно до внутрішньої логіки S-функції, яка реалізована у файлі **S\_KA.m**.

## **6 ВСТАНОВЛЕННЯ ПАРАМЕТРІВ РОЗРАХУНКУ І ЙОГО ВИКОНАННЯ**

Перед виконанням розрахунків необхідно задати параметри розрахунку. Встановлення параметрів розрахунку виконується на панелі керування меню Simulation/Parameters. Вигляд панелі керування наведено на рисунку 6.1.



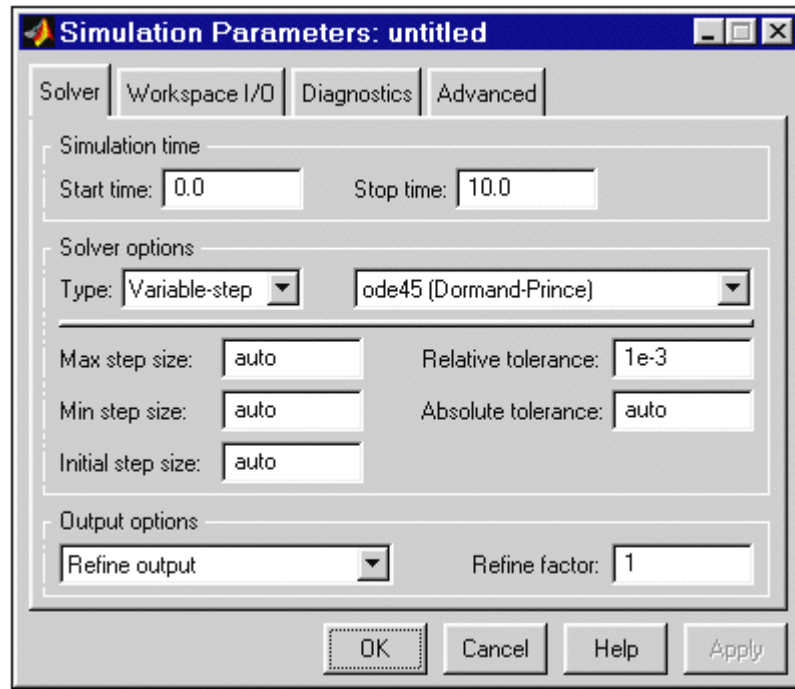


Рисунок 6.1 - Панель управління

Вікно конфігурації параметрів розрахунку складається з чотирьох вкладок:

- Solver (Розрахунок): Налаштування параметрів обчислення моделі.
- Workspace I/O (Введення/виведення даних у робочу область): Встановлення параметрів обміну даними з робочою областю MATLAB.
- Diagnostics (Діагностика): Вибір параметрів діагностичного режиму.
- Advanced (Додатково): Налаштування додаткових параметрів. Установка параметрів розрахунку моделі здійснюється через елементи управління, розташовані на вкладці Solver. Ці елементи поділяються на три групи (див. рисунок 6.1): Simulation time (Час моделювання або, іншими словами, інтервал обчислення), Solver options (Параметри розрахунку) та Output options (Параметри виведення).

### 6.1 Встановлення параметрів розрахунку моделі

### 6.1.1 Інтервал моделювання або час розрахунку

Час розрахунку визначається шляхом встановлення початкового (Start time) і кінцевого (Stop time) значень для інтервалу обчислення. Зазвичай, початковий час встановлюється на рівень нуля. Величина кінцевого часу визначається користувачем, враховуючи особливості задачі, що вирішується.

### 6.1.2 Параметри розрахунку

При виборі параметрів розрахунку необхідно вказати спосіб моделювання та метод розрахунку нового стану системи. Параметр "Type" має два варіанти – "Fixed-step" (фіксований крок) або "Variable-step" (змінний крок). Зазвичай "Variable-step" використовується для моделювання безперервних систем, тоді як "Fixed-step" – для дискретних.

Список методів розрахунку нового стану системи включає кілька варіантів. Перший варіант (Discrete) використовується для розрахунку дискретних систем. Інші методи використовуються для обчислення безперервних систем. Ці методи різняться для змінного (Variable-step) і фіксованого (Fixed-step) кроку часу, проте, в основному, це процедури розв'язання систем диференціальних рівнянь. Детальний опис кожного методу розрахунку станів системи подається в довідковій системі MATLAB.

Нижче розташовані два розкриваючі списки "Type", область вмісту яких змінюється в залежності від обраного способу зміни модельного часу. При виборі "Fixed-step" у цій області з'являється текстове поле "Fixed-step size" (величина фіксованого кроку), що дозволяє встановлювати величину кроку моделювання (рисунок 6.2). Розмір кроку моделювання за замовчуванням встановлюється системою автоматично. Потрібну величину кроку можна ввести замість значення "auto" у вигляді числа або виразу для обчислення

(той же підхід застосовується до всіх параметрів, що встановлюються системою автоматично).

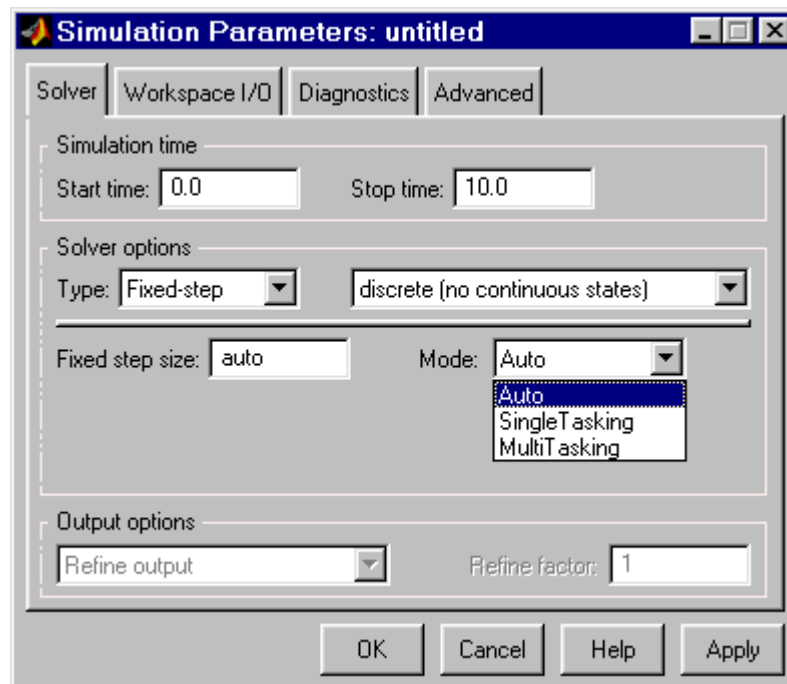


Рисунок 6.2 - Вкладка Solver при виборі фіксованого кроку розрахунку

При виборі режиму Fixed-step, також необхідно визначити режим розрахунку (Mode), який може бути обраний із трьох варіантів:

- **MultiTasking (Многозадачний):** Використовується, якщо в моделі присутні паралельно працюючі підсистеми, і результат роботи моделі залежить від тимчасових параметрів цих підсистем. Цей режим дозволяє виявити невідповідність швидкості та дискретності сигналів, що пересилаються блоками один одному.
- **SingleTasking (Однозадачний):** Використовується для моделей, де недостатньо строга синхронізація роботи окремих компонентів не впливає на кінцевий результат моделювання.
- **Auto (Автоматичний вибір режиму):** Дозволяє Simulink автоматично встановлювати режим MultiTasking для моделей з різними швидкостями передачі сигналів і режим SingleTasking для моделей, що містять блоки, які працюють з однаковою швидкістю.

При виборі Variable-step з'являються поля для введення трьох параметрів:

- Max step size (Максимальний крок розрахунку): Зазвичай встановлюється автоматично (auto), але може бути явно задано користувачем. Значення за замовчуванням визначається як  $(\text{StopTime} - \text{StartTime}) / 50$ , проте це може призводити до ламаних (не плавних) ліній на графіках. У таких випадках користувач повинен вказати величину максимального кроку розрахунку.
- Min step size (Мінімальний крок розрахунку).
- Initial step size (Початкове значення кроку моделювання).

При моделюванні безперервних систем з використанням змінного кроку, необхідно вказати точність обчислень: відносну (Relative tolerance) та абсолютну (Absolute tolerance). За замовчуванням, ці значення дорівнюють, відповідно,  $10^{-3}$  та auto.

У розділі Output options (Параметри виводу), який розташований у нижній частині вкладки Solver, встановлюються параметри виведення вихідних сигналів моделі. Для цього параметра доступні три варіанти:

- Refine output (Скоригований висновок): Дозволяє змінювати дискретність реєстрації модельного часу та сигналів у робочій області MATLAB за допомогою блоку "To Workspace". Встановлення величини дискретності виконується через Refine factor. За замовчуванням, Refine factor дорівнює 1, що означає реєстрацію для кожного значення модельного часу.
- Produce additional output (Додатковий висновок): Забезпечує додаткову реєстрацію параметрів моделі у задані моменти часу. Значення вводяться у полі "Output times" у вигляді списку, укладеного в квадратні дужки.

- Produce specified output only (Формувати лише заданий висновок):  
Встановлює виведення параметрів моделі лише у вказані моменти часу, які визначаються в полі "Output times" (Моменти часу виведення).

## 6.2 Встановлення параметрів обміну робочою областю

Елементи управління введенням та виведенням проміжних даних та результатів моделювання в робочу область MATLAB знаходяться на вкладці "Workspace I/O" (див. рисунок 6.3).

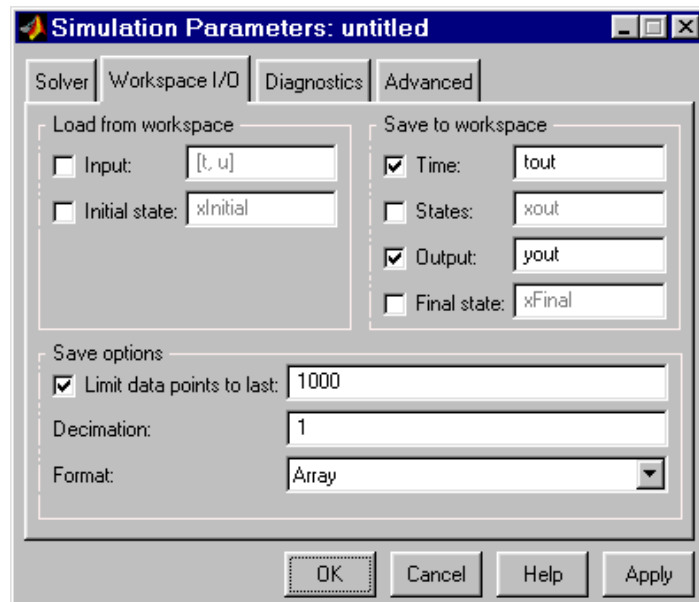


Рисунок 6.3 - Вкладка Workspace I/O діалогового вікна установки параметрів моделювання

Елементи вкладки розділені на три поля:

### 1. Load from workspace (Завантажити із робочої області):

- Прапорець "Input" дозволяє ввести формат даних, які будуть зчитуватися з робочої області MATLAB.
- Встановлення прапорця "Initial State" дозволяє ввести ім'я змінної, що містить параметри початкового стану моделі.

### 2. Save to workspace (Записати в робочу область):

- Дозволяє встановити режим виведення значень сигналів у робочу область MATLAB та вказати їх імена.

### 3. Save options (Параметри запису):

- Вказує кількість рядків під час передачі змінних до робочої області.
- Параметр "Decimation" визначає крок запису змінних у робочу область.
- Параметр "Format" задає формат даних, що передаються в робочу область.

## 6.3 Встановлення параметрів діагностування моделі:

Вкладка "Diagnostics" (див. Рисунок 6.4) дозволяє змінювати перелік діагностичних повідомлень, що виводяться Simulink у командному вікні MATLAB, а також встановлювати додаткові параметри діагностики моделі.

Повідомлення про помилки або проблемні ситуації, виявлені Simulink під час моделювання та які вимагають втручання розробника, виводяться у командному вікні MATLAB. Розробник може вказати вид реакції на кожну з них, використовуючи групу перемикачів у полі Action (вони стають доступними, якщо у списку вибрано одну з подій):

- None - ігнорувати,
- Warning - видати попередження та продовжити моделювання,
- Error - видати повідомлення про помилку та зупинити сеанс моделювання.

Вибраний вид реакції відображається у списку поруч із найменуванням події.

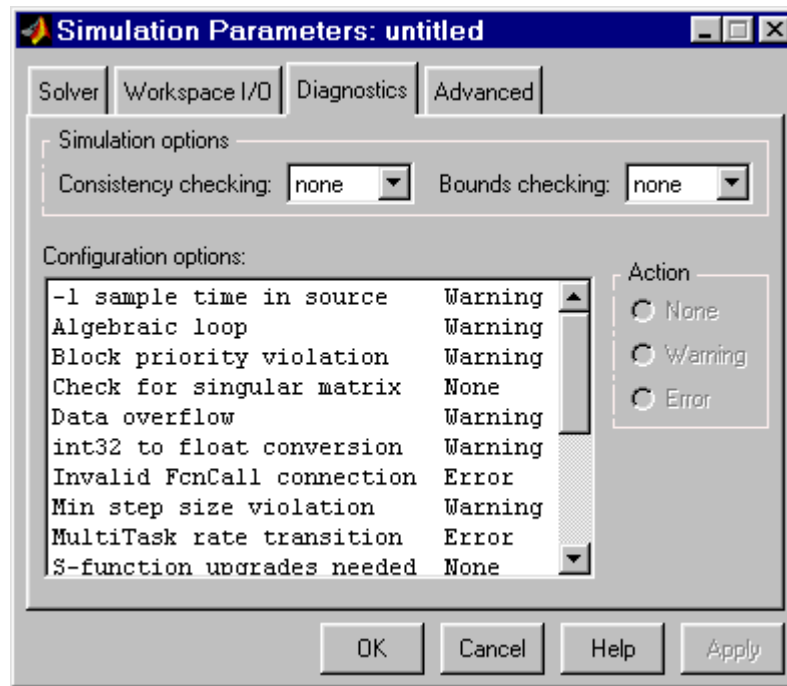


Рисунок 6.4 - Вкладка Diagnostics вікна установки параметрів моделювання

## 6.4 Виконання розрахунку

Запуск розрахунків виконується шляхом вибору пункту меню Simulation/Start або використання відповідного інструмента на панелі інструментів. Процес розрахунку може бути припинений достроково через вибір пункту меню Simulation/Stop або використання відповідного інструмента. Розрахунок також можна призупинити (Simulation/Pause) і потім відновити (Simulation/Continue).

## ВИСНОВКИ

Вивчення особливостей системи MATLAB вказує на її унікальність як застосовуваної системи для програмування та моделювання. Ця система дозволяє проводити розрахунки в різних галузях науки та техніки з високою ефективністю. Особливо успішно вона використовується для математичного моделювання механічних систем, зокрема в динаміці, гідродинаміці, аеродинаміці, акустиці, енергетиці та інших областях, кожна з яких має свої конкретні завдання.

Система MATLAB надає спеціальні інструменти для реалізації широкого спектру завдань, зокрема у сферах електротехніки та радіотехніки. Ці інструменти включають операції з комплексними числами, роботу з матрицями, векторами та поліномами, обробку даних, аналіз сигналів та цифрову фільтрацію. Також система забезпечує засоби для обробки зображень та реалізації нейронних мереж. Це розширює можливості системи та дозволяє створювати високоефективні віртуальні прилади для вирішення складних завдань у галузі механіки.



**ПЕРЕЛІК ПОСИЛАНЬ**

1. Лазарев Ю. Ф. Початки програмування у середовищі MatLAB: Навч. посібник. – К.: "Корнійчук", 1999. – 160 с.
2. Лазарев Ю. Ф. MatLAB 5.x. – К.: Издат. группа BHV, 2000. – 384 с.
3. Лазарев Ю. Ф. Моделирование процессов и систем в MATLAB. Учебный курс. – СПб.: Питер; Киев: Издат. группа BHV, 2005. – 512 с.
4. Лазарев Ю. Ф. Моделювання на ЕОМ. Навч. посібник. – К.: Корнійчук, 2007.-290 с.
5. Мартынов Г. Г., Иванов А. П. MATLAB 5.x, вычисления, визуализация, программирование. - М.: "Кудиц-образ", 2000. - 332 с.
6. Гульяев А. Визуальное моделирование в среде MATLAB: учебный курс. - СПб. : "Питер", 2000. - 430 с.
7. Краснопрошина А. А., Репникова Н. Б., Ильченко А. А. Современный анализ систем управления с применением MATLAB, Simulink, Control System: Учебное пособие. - К.: "Корнійчук", 1999. - 144 с.
8. Мартынов Г. Г., Иванов А. П. MATLAB 5.x, вычисления, визуализация, программирование. - М.: "Кудиц-образ", 2000. - 332 с.
9. Медведев В. С., Потемкин В. Г. Control System Toolbox. MatLAB 5 для студентов. - М.: ДИАЛОГ-МИФИ, 1999. - 287 с.
10. Потемкин В. Г. Система MatLAB: Справ. пособие. - М.: ДИАЛОГ-МИФИ, 1997. - 350 с.
12. Потемкин В. Г., Рудаков П. И. MatLAB 5 для студентов. - 2-е изд., испр. и дополн. - М.: ДИАЛОГ-МИФИ, 1999. - 448 с.
13. Потемкин В. Г. Система инженерных и научных расчетов MatLAB 5.x: - В 2-х т.Том 1. - М.: ДИАЛОГ-МИФИ, 1999. - 366 с.
14. Потемкин В. Г. Система инженерных и научных расчетов MatLAB 5.x: - В 2-х т. Том 2. - М.: ДИАЛОГ-МИФИ, 1999. - 304 с.
15. Рудаков П. И., Сафонов В. И. Обработка сигналов и изображений. MATLAB 5x. - М.: "ДИАЛОГ-МИФИ", 2000. - 413 с.