



УДК 681.513;62.505;621.9.04

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ МОДЕЛЬНО-ОРІЄНТОВАНОГО ПРОЄКТУВАННЯ КОНТРОЛЕРУ РЕАЛЬНОГО ЧАСУ

INFORMATION TECHNOLOGY OF MODEL-BASED DESIGN REAL-TIME CONTROLLER

Zozulia V.A.¹, Osadchiy S.I.², Timoshenko A.S.³¹ Державний торговельно-економічний університет, м. Київ, Україна² Льотна академія Національного авіаційного університету, м. Кропивницький, Україна³ Льотна академія Національного авіаційного університету, м. Кропивницький, Україна¹к.т.н., доцент, ²д.т.н., проф., ³ст.викл.ORCID: ¹<https://orcid.org/0000-0003-3793-4686>, ²<https://orcid.org/0000-0002-1811-3594>, ³<https://orcid.org/0000-0003-2411-497X>E-mail: ¹irish38@ukr.net, ²srg2005@ukr.net, ³jaanti1980@gmail.com

Copyright © 2023 by author and the journal “Automation of technological and business – processes”.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0>

DOI: 10.15673/atbp.v%vi%i.2493

Анотація. Підвищення конкурентоспроможності вітчизняних систем автоматичного керування вимагає інтенсифікації проектно-конструкторської роботи на основі впровадження сучасних інформаційних технологій. Однією з необхідних умов зазначеного впровадження є використання інформаційної технології побудованою на методи модельно-орієнтованого проектування [1, 2] з використанням синергії апаратного забезпечення National Instruments (NI), програмних модулів Labview, Matlab та SolidWorks, для розробки контролера реального часу системи керування програмного руху робочої поверхні платформи Стюарта. В статті показано, що поєднання в інформаційну технологію трьох систем Solidworks, яка дозволяє твердотільне моделювання кінематики і динаміки на основі фундаментальних фізичних принципів механіки, Matlab, що має найкращі функціональні можливості при моделюванні динамічних систем, аналізі та синтезі систем керування та Labview, який має найкращі апаратно-програмні засоби проектування вбудованого контролера реального часу, значно розширює можливості технології модельно-орієнтованого проектування систем керування складних динамічних об'єктів. Наведено механізми імпорту, експорту та з'єднання створених моделей згідно етапів проектування. Поєднання використання програмних інструментів при проектуванні швидко приводить до створення прототипу системи керування, тестування та верифікації програмного забезпечення, при цьому гарантується повне відслідковування складу її параметрів та іншої конструкторської інформації на всіх етапах проектування. Створений таким чином контролер реального часу системи керування програмного руху робочої поверхні платформи Стюарта, буде з проектованою оптимальною структурою, що підвищить швидкодію та надійність.

Abstract: Increasing the competitiveness of domestic automatic control systems requires the intensification of design and construction works based on the introduction of modern information technologies. One of the necessary conditions for this implementation is the use of information technology based on the method of Model-Based design [1, 2] using the synergy of National Instruments (NI) hardware, software modules Labview, Matlab, and SolidWorks, for the development of a real-time controller of the software motion control system of the work surface of the Stewart platform. The article shows that the compound in the information technology of three systems, it's the Solidworks, which allows solid-state modeling of kinematics and dynamics based on the fundamental physical principles of mechanics, Matlab, which has the best functionality in modeling dynamic systems, analysis, and synthesis of control systems and Labview, which has the best hardware and software tools for designing an embedded real-time controller, significantly expands the capabilities of the technology of Model-Based design of control systems of complex dynamic objects. The mechanisms of import, export, and connection of the created models according to the design stages are given. The combination of the use of software tools during design quickly leads to the creation of a prototype of the control system, testing and verification of the software, while guaranteeing full tracking of the composition of its parameters and other design information at all



stages of design. The real-time controller of the software motion control system of the work surface of the Stewart platform created in this way will be designed with an optimal structure that will increase the speed and reliability.

Keywords: design information technology, model-oriented design, real-time controller, offline and real-time simulation SolidWorks, MatLab, LabView, Stewart

Ключові слова: інформаційна технологія проектування, модельно-орієнтоване проектування, контролер реального часу, оффлайн-моделювання та моделювання в реальному часі SolidWorks, MatLab, LabView, Стюарт

Вступ

Постійний розвиток технічних засобів САК та об'єднання об'єктів керування в великі комплекси показує, що використання тільки традиційного інструментарію при проектуванні САК в даний час є недостатнім з огляду на складний характер сучасних систем керування. Труднощі традиційного підходу до проектування систем керування полягають в тому, що на етапі складання вимог і специфікацій використовувани текстові документи недосконалі при ітеративному підході, фізичні прототипи на етапі проектування також недосконалі, складні і дорогі. На етапі реалізації при ручній розробці застосування різних програмних засобів і людські помилки роблять процес ненадійним, а на етапі перевірки традиційне тестування призводить до виявлення помилок лише в кінцевій стадії розробки [1].

Ці труднощі багато в чому долаються при використанні технології модельно-орієнтованого проектування. Модельно-орієнтоване проектування (МОП) - це математичний і візуальний метод вирішення завдань, пов'язаних з проектуванням систем керування, заснований на використанні наочної імітаційної моделі майбутнього виробу, яка є основним носієм інформації про його концепції, особливості конструкції і реалізації. Така модель використовується протягом усіх стадій процесу проектування: дослідження, конструювання, реалізація, тощо. На кожному етапі модель оновлюється і еволюціонує. При цьому гарантується повне відслідковування складу її параметрів та іншої конструкторської інформації на всіх етапах проектування [1, 2].

Принципи МОП істотно відрізняються від традиційної методології проектування. Замість створення складних програмних кодів розробники можуть застосовувати МОП для поліпшення характеристик моделі, використовуючи стандартні функціональні блоки з безперервним і дискретним часом. Побудовані таким чином моделі разом з використанням інструментів для моделювання можуть швидко привести до створення прототипу системи керування, тестування та верифікації програмного забезпечення. [4, 3].

МОП ефективно застосовується при проектуванні систем керування рухом в складних, багатовимірних об'єктах, в таких, як мехатронні системи. Одним із таких мехатронних об'єктів є платформа Стюарта, яка має шість однотипних кінематичних ланцюгів (штанг) (рис. 1) [5]. Програмно регулюючи довжину штанг l_i платформи Стюарта, можна керувати положенням вихідної ланки (робочою поверхнею), переміщати її в вертикальному і горизонтальному напрямках, повертати в трьох площинах. Платформа Стюарта, має шість ступенів вільності робочої поверхні: три поступальні ступені рухливості (зміщення характерної точки O вздовж осей нерухої системи координат, пов'язаної з нерухомим основою) і три обертальні ступені рухливості (поворот робочої поверхні щодо власних осей рухої системи координат, пов'язаної з робочою поверхнею).

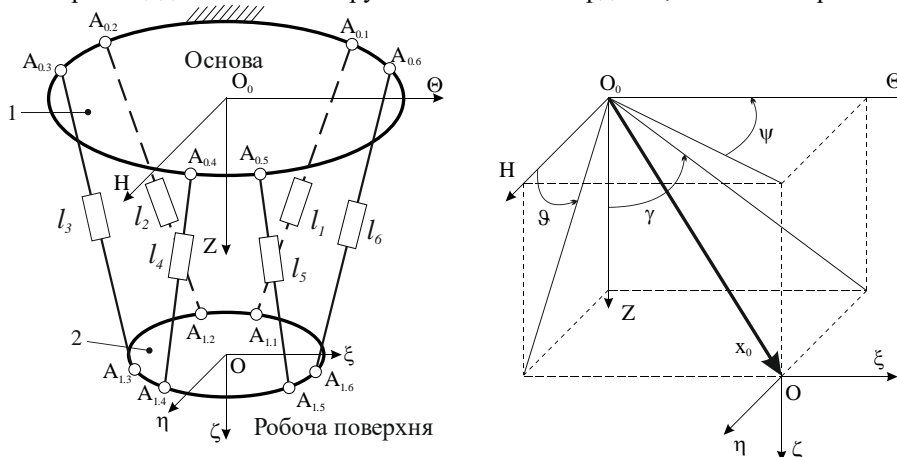


Рис. 1 Кінематична схема платформи Стюарта
Fig. 1 Kinematic scheme of the Stewart platform

1. Аналіз літературних даних і постановка проблеми

Існують публікації [1-4] спрямовані на теоретичне обґрунтування МОП, зокрема розкривають переваги такі, як скорочення часу виходу на ринок, зниження кількості помилок, які допускаються при проектуванні, програмуванні та тестуванні, що підвищує безпеку та надійність [6, 7], але не дають рекомендації до практичного застосування.

Багатьма авторами були досліджено використання МОП тільки з використанням одного виду прикладних засобів. Так в роботах [8-10], було розглянуто можливості модельно-орієнтованого проектування за допомогою програмних продуктів MathWorks, що дозволяють автоматизувати процес розробки, налагодження та верифікації



програмного забезпечення для систем керування.

Також існують публікації в яких показано застосування технології МОП для проектування системи керування платформи Стюарта. В роботі [11], було показано, що застосування технології МОП дозволило значно скоротити термін розробки складного механічного пристрою та його системи керування. А також, зазначено, що складність керування будь-яким паралельним механізмом полягає в тому, що при неузгодженій роботі його рухомих опор негативними результатами буде не тільки неузгодженість з необхідною траєкторією руху платформи, але і вихід з ладу всього механізму через заклинювання рухомих елементів. Так в роботі [12], показано переваги використання МОП в поєднанні Labview та Solidworks для проектування системи керування платформи Стюарта.

Аналіз досліджень та публікацій з питань застосування технології модельно-орієнтованого проектування системи керування, показує, що при наявності існуючого теоретичного обґрунтування цієї технології, переваг над традиційним підходом, особливо при проектуванні складних мехатронних об'єктів та застосування окремих прикладних засобів не дозволяють ефективно застосовувати таку технологію.

Тому, поєднання в інформаційну технологію трьох систем Solidworks, яка дозволяє твердотільне моделювання кінематики і динаміки на основі фундаментальних фізичних принципів механіки, Matlab, що має найкращі функціональні можливості при моделюванні динамічних систем, аналізі та синтезі систем керування та Labview, який має найкращі апаратно-програмні засоби проектування вбудованого контролера реального часу, значно розширює можливості технології модельно-орієнтованого проектування систем керування складних динамічних об'єктів.

2. Мета і матеріали дослідження

Метою дослідження є створення інформаційної технології, яка ґрунтується на основі методу модельно-орієнтованого проектування, з використанням пакетів прикладних програм SolidWorks, MatLab та LabView на прикладі розробки вбудованого контролера реального часу системи керування програмного руху робочої поверхні платформи Стюарта, що забезпечує збереження заданої точності виконання програмного руху у всьому просторі робочої зони.

Методологія МОП має наступні етапи проектування [3]:

1. Побудова моделі об'єкта. Побудова моделі може бути емпіричним і теоретичним. При емпіричному побудові моделі використовуються такі методи, як ідентифікація системи. До теоретичного моделювання належить фізичне моделювання, де модель створюється за допомогою з'єднуються блоків, що представляють собою фізичні елементи, з яких фактично складається модель.

2. Аналіз і побудова системи керування. Математична модель, сконструйована на кроці 1, використовується для визначення динамічних характеристик моделі об'єкта. На основі цих характеристик будується система керування.

3. Оффлайн-моделювання та моделювання в реальному часі. Час відгуку динамічної системи на вхідні дані, що змінюються в часі, досліджується за допомогою симуляції моделі у вигляді простої лінійної стаціонарної системи або нелінійної системи. Симуляція дозволяє негайно визначити параметри моделі, вимоги, що накладаються на неї, і помилки побудови до початку проектування. Моделювання в реальному часі може бути здійснено за допомогою автоматичної генерації коду системи керування, побудованої на кроці 2. Таким чином система керування може бути протестована в реальному часі на моделі об'єкта.

4. Реалізація вбудованого регулятора. Малоімовірно, що система керування буде працювати в реальній системі так само добре, як це було при моделюванні, тому ітераційний процес налагодження здійснюється на основі аналізу результатів на фактичному об'єкті і поновлення моделі регулятора.

На підставі вище сказаного було розроблено загальну структуру інформаційної технології МОП, що визначає взаємодію у процесі проектування, при цьому ефективно реалізуючи V-подібний цикл розробки (рис. 2).

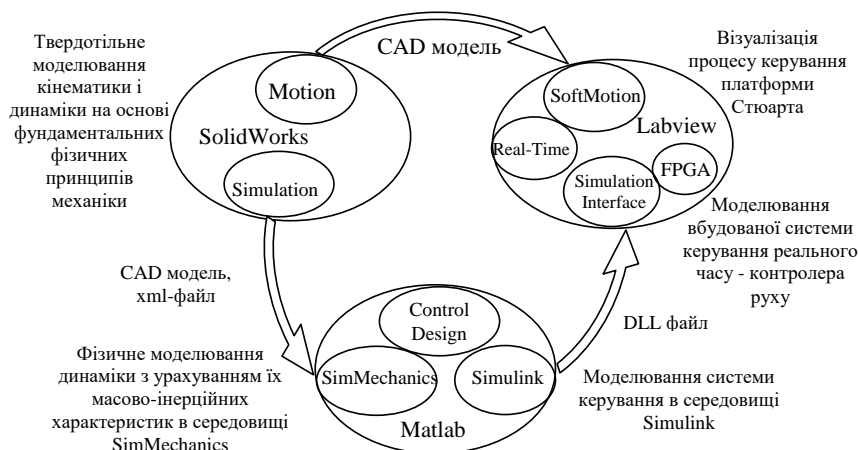


Рис. 2 Структурна схема інформаційної технології розробки вбудованої систем реального часу за допомогою методу модельно-орієнтованого проектування

Fig. 2 Structural diagram of information technology development of embedded real-time systems using the



3. Результати досліджень

Згідно *першого етапу* технології МОП буда підготовлена розрахункова модель платформи Стюарта в середовищі SolidWorks (рис. 3). У тривимірній моделі реалізовані основні деталі та вузли з урахуванням матеріалів та технологій виробництва. Базова частина моделі формується в базовому пакеті SolidWorks Standard (рис. 3а). Для SolidWorks Motion, було спрощено тривимірну модель збереженням всіх необхідних з'єднань, які в середовищі SolidWorks формуються як сполучення (рис. 3б).

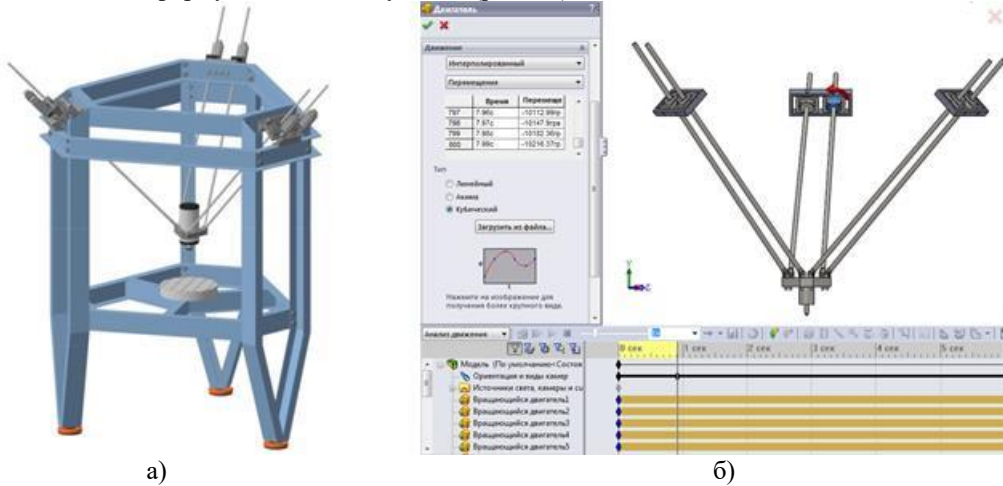


Рис. 3 Платформа Стюарта, модель в середовищі SolidWorks а) та в SolidWorks Motion б)
Fig. 3 Stewart's platform, a model in the SolidWorks environment а) and in SolidWorks Motion б)

Верхня та нижня платформи, рухома та нерухома частини виконавчих механізмів є ланками у платформі Стюарта. Ці зв'язки визначаються їх значеннями тензора маси та інерції з посиланнями на системи координат тіла у їхніх центрах тяжкості. Ці системи координат тіла мають бути прив'язані до інерційної системи координат. Центри тяжкості ланок також визначаються з посиланням на їх системи координат тіла. З'єднання, що з'єднують ланки, представлені визначенням їх осей руху (поступальний та/або обертальний). Вони можуть приводитися в дію динамічно як сила/крутний момент, а також кінематично (положення/швидкість/ прискорення).

Для перевірки кінематичних залежностей було проведено моделювання руху робочої поверхні платформи Стюарта за допомогою вбудованого інструменту пакету Solidworks Motion, який забезпечує моделювання фізичної динаміки механічних систем. Для перевірки кінематичних залежностей по заданій довільно обраній траєкторії руху робочої поверхні були розраховані координати приводів, які були записані у файл і завантажені до віртуальних приводів Solidworks Motion. Після запуску розрахунку робоча поверхня віртуальної моделі платформи Стюарта виконала задані переміщення, зберігаючи задану орієнтацію, що свідчить про правильність розроблених кінематичних залежностей. Також, відсутність неузгодженості роботі рухомих штанг платформи Стюарта та негативних наслідків не тільки неузгодженості з необхідною траєкторією руху платформи, але і виходу з ладу всього механізму через заклинювання рухомих елементів.

Для виконання *другого етапу* технології МОП необхідно створену модель платформи Стюарта в SolidWorks передати до Matlab, згідно рисунку 2. Процес імпорту моделі SolidWorks до Matlab/SimMechanics складається з експорту 3D-моделі SolidWorks до xml-файлу та stl-файлу, імпорту xml-файлу та stl-файлу до середовища SimMechanics (рис.4).

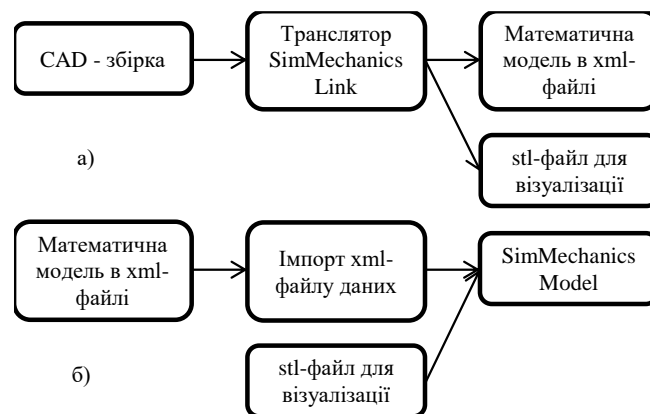


Рис. 4 Процес експорту а) та імпорту б) моделі SolidWorks до Matlab/SimMechanics
Fig. 4 The process of exporting а) and importing б) a SolidWorks model to Matlab/SimMechanics



XML-файл, це текстовий файл, в якому деталі вихідної збірки механізму інтерпретуються як тіла, а сполучення між ними як ступеня вільності. STL – файл, створюваний у процесі трансляції, містять опис геометрії деталей.

При цьому всі механічні параметри твердотільної моделі, що визначають її динаміку (маса, інерційні характеристики, координати центрів мас, параметри з'єднань тощо) передаються динамічній моделі. SimMechanics автоматично формує систему диференціальних рівнянь руху механізму (вона є внутрішнім об'єктом, прихованим від користувача).

При складанні математичної моделі будь-якого механізму природне прагнення врахувати якомога більшу кількість факторів, що впливають на його динаміку. Розумне, обгрунтоване спрощення моделі – ключове завдання моделювання. Важливо виявити основні, найбільш суттєві сторони об'єкта та абстрагуватися від несуттєвих.

В зв'язку відсутності в даному дослідженні, необхідності представлення більш детальної розробки моделі в MATLAB/Simulink, тому у моделі платформи Стюарта використовувалися такі припущення: сферичні шарніри вважалися ідеальними, без урахування люфту, сухого тертя та пружних деформацій у штангах.

Після доопрацювання моделі в SimMechanics для забезпечення можливості її зв'язку з об'єктами Simulink, результат імпорту моделі платформи Стюарта у середовищі SimMechanics представлений на рисунку 5.

Механічна частина маніпулятора складається з нерухомої бази, рухомої платформи (*TopPlate*) та шести послідовних кінематичних ланцюгів (*Leg1, Leg2, ..., Leg6*), що з'єднують платформу та базу. Кожна послідовний ланцюг паралельного механізму, складається з двох тіл - шатуна і шарніра, з'єднаних карданним з'єднанням. Для імітації нерухомої бази системи, яка насправді представлена шістьма поворотними виконавчими двигунами, в структурно-динамічному описі використовуються елементи фіксації з нерухомою основою. Кожен такий елемент жорстко з'єднаний із шарніром відповідного кінематичного ланцюга за допомогою поворотного з'єднання з двома ступенем вільності. Аналогічно з рухомою платформою, яка представлена одним однорідним елементом, за допомогою призматичного шарніра з'єднані шість шатунів (рис. 5).

Для моделювання руху платформи використовується елемент *Joint Actuator*, що входить у модель кожного окремого послідовного кінематичного ланцюга *Leg1, Leg2, ..., Leg6* паралельного механізму (рис. 5), на вхід якого подається керуючий сигнал, що є функціональною залежністю зміни обертаючого моменту в часі. Для моделі платформи вхідними змінними будуть шість сил (*Force*), що розвиваються електроприводом штанг, в результаті дії яких рухома платформа механізму переходить у відповідне положення та орієнтацію, при цьому вихідними величинами будуть переміщення та швидкості переміщення штанг.

За допомогою блоку *Joint Sensor* вимірюється швидкість та положення штанги. Усі шість штанг впливають на рухома платформу – блок *Top Plate*. Таким чином, ми отримали модель паралельного маніпулятора, яка оформляється у вигляді окремої моделі SimMechanics.

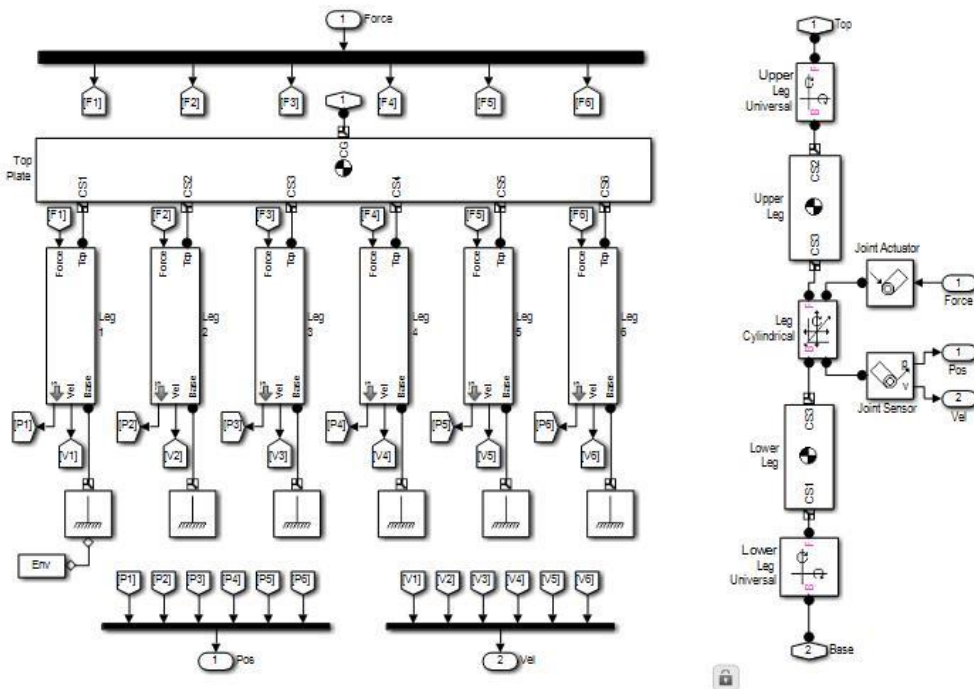


Рис. 5. SimMechanics-модель платформи Стюарта та підсистеми штанги
Fig. 5 SimMechanics model of the Stewart platform and subsystems of the rod

Імітаційна модель платформи Стюарта у комплексі візуального моделювання MATLAB/Simulink складається з наступних блоків: пристроєм узгодження сигналів керування (*Leg Trajectory*); дискретно-позиційної системи керування (*H_inf*); сервоприводів (*Servo*); механічної системи платформи Стюарта (*Plant*), з якої знімаються дані



про положення (*position*) та швидкість (*velocity*) зміни довжин штанг, а також блоку підрахунку та демонстрації результатів моделювання Scope (рис. 6). При натисканні на блок «*Body Position Sensor*» з'являється вікно для вибору параметрів перегляду: візуалізація рухів платформи Стюарта, зміна координат центру рухомої платформи та кутів орієнтації, переміщення штанг, швидкостей переміщень, прискорень, помилок обчислення і тощо. Зміна вибраних параметрів виконується у функції часу. Для перегляду результатів обчислень та демонстрації результатів моделювання на екрані дисплея слід натиснути на відповідну кнопку блоку «*Scope*» (рис. 6).

Це модель, так звана з «прямої динамікою», в якій сили є вхідними сигналами для системи, що моделює зусилля виконавчого механізму, а положення та швидкість виконавчих механізмів та рух платформи є вихідними сигналами. Ця модель використовується при реалізації системи керування, оскільки положення та швидкості виконавчих механізмів будуть сигналами зворотного зв'язку, а керуючими сигналами будуть зусилля на виконавчих механізмів. При моделюванні отримане необхідне значення довжини штанги порівнюється з поточним і подається на систему керування, де вираховуються корегувальні значення довжини штанг та подається на приводи.

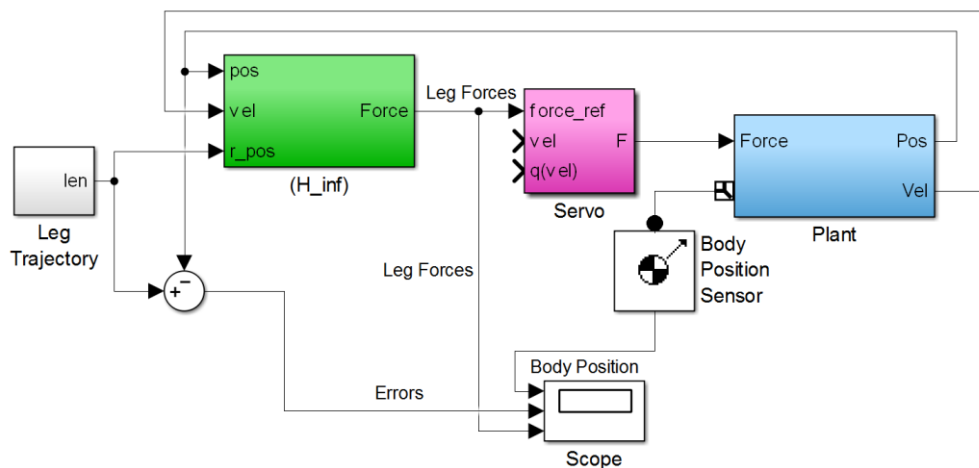


Рис. 6. Simulink-модель платформи Стюарта з системою керування
Fig. 6 Simulink model of Stewart platform with control system

У блоці «*Leg Trajectory*» на вхід подається завдання, одержуване у вигляді координат та кутів центру рухомої платформи, ξ_{pos} , η_{pos} , ζ_{pos} , ψ_{ang} , ϑ_{ang} , γ_{ang} . Де ξ_{pos} , η_{pos} , ζ_{pos} , поступальні переміщення вздовж осей Θ , H , Z , ψ_{ang} , ϑ_{ang} , γ_{ang} кути поворотів щодо цих осей (кути Ейлера) (рис. 1). В даному випадку приймаємо $\psi_{ang} = 0^\circ$, $\vartheta_{ang} = 0^\circ$, $\gamma_{ang} = 0^\circ$. Далі, проводиться перерахунок завдання, одержуваного у вигляді координат та кутів центру рухомої платформи штанг платформи Стюарта $l_1, l_2, l_3, l_4, l_5, l_6$.

Довжини штанг $l_1, l_2, l_3, l_4, l_5, l_6$ у вигляді напруги подаються на двигуни і на виході отримуємо наступні змінені значення довжин штанг $l_1, l_2, l_3, l_4, l_5, l_6$. При моделюванні отримане необхідне значення довжини штанги порівнюється з поточним і подається на систему керування, яка виробляє значення сили, що подається на штанги. Далі, змінені значення довжин штанг $l_1, l_2, l_3, l_4, l_5, l_6$ перераховуються у значення координат та кутів центру рухомої платформи $\varphi_{ang}, \psi_{ang}, \chi_{ang}$. При цьому виникають помилки позиціонування між заданими координатами та кутами центру рухомої платформи, які отриманими під час моделювання.

У сучасних промислових мехатронних системах в якості виконавчих двигунів використовуються, головним чином, синхронні та асинхронні двигуни змінного струму, динаміка яких описується складними істотно нелінійними рівняннями. У той же час сучасні блоки керування (перетворювачі) містять математичні моделі двигунів, на базі яких в реальному масштабі часу здійснюється лінеаризація їх нелінійних характеристик і зазначені вище двигуни з метою аналітично з успіхом описуються лінійними залежностями аналогічно двигунам постійного струму. Важливим етапом при побудові позиційного керування є синтез приводу за заданими показниками якості.

У зв'язку з цим при створенні моделі необхідно налаштувати регулятори сервоприводу з використанням пакета Simulink.

Типовий перетворювач в сервопривод містить регулятори швидкості, струму, датчик струму. Регулятори швидкості і струму є ПІД – регуляторами із змінною структурою, а зворотний зв'язок за швидкістю організується шляхом підрахунку числа імпульсів з датчика кута повороту ротора в одиницю часу.

Динамічні властивості системи багато в чому визначаються якістю контуру швидкості. Тому необхідно і достатньо здійснити параметричний синтез регулятора швидкості. Для цього можна скористатися процедурою синтезу Nonlinear Control Design Blockset (NCD-Blockset), розробленою програмістами фірми Math і наявною в бібліотеці MatLab. Інструментальний пакет NCD-Blockset надає графічний інтерфейс для налаштування параметрів динамічних об'єктів, що забезпечують бажану якість перехідних процесів.



У вікні налаштування блоку Joint Actuator встановлюємо вид збудження Generalized forces (узагальнені сили) – Apply torque (момент), у вікні налаштування Joint Sensor виконуємо вибір тих характеристик відносного руху частин механізму, які необхідно зняти. Для виведення на екран знятих характеристик використовуємо блок Scope (осцилограф спостереження тимчасових залежностей).

Для завдання руху до блоків Joint Actuator додаємо комп'ютерну модель приводу, наведену на рисунку 7.

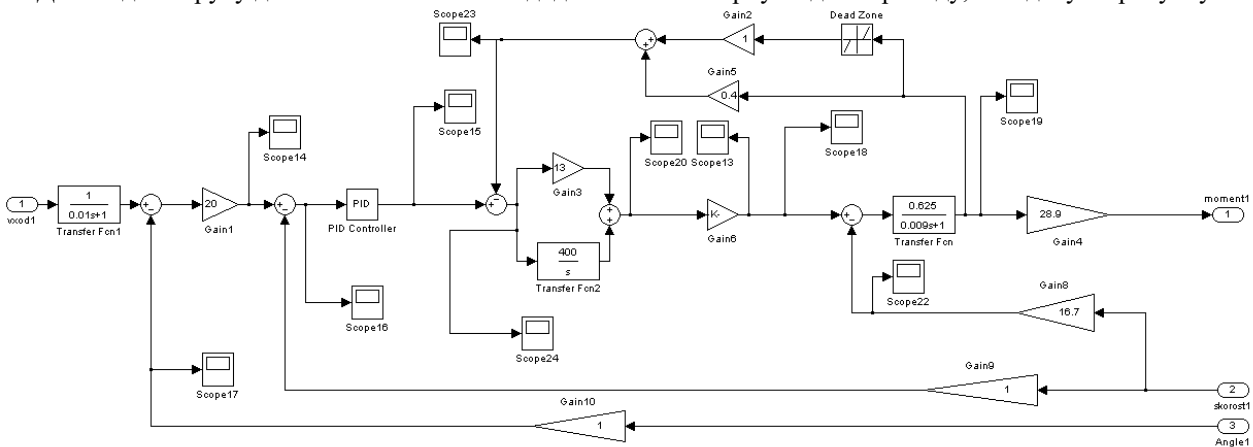


Рис. 7. Комп'ютерна модель приводу
Fig. 7 Computer model of the drive

Привід має контур регулювання положення, швидкості та струму. При моделюванні використані параметри реальних двигунів та контур струму. Параметри регулятора швидкості налаштовувалися за моделлю приводу для середніх значень навантаження за допомогою блоку NCD, що здійснює автоматизований пошук коефіцієнтів ПІД регулятора за заданою кривою перехідного процесу кореневими методами.

За допомогою всіх складових перевіреної моделі платформи Стюарт можна перейти до синтезу її системи керування (відповідно до *другого етапу* технології МОП), яка враховуватиме динамічні характеристики механізму. Необхідно отримати лінійну модель платформи Стюарта, щоб можна було використовувати потужніші методи проектування лінійних елементів керування. Для цього необхідно знайти точку рівноваги для нашої нелінійної моделі, витягуючи рівноважні сили для кожної опори, які утримують платформу Стюарта в нерухомому стані. Після вибору цього параметра моделюємо модель і використовуємо вихідні дані блоків датчиків для отримання сили, яка буде додана до блоків виконавчих механізмів для кожної опори, приводячи в стан спокою платформу.

Тепер можна згенерувати модель системи керування у просторі стану LTI на основі моделі SimMechanics для використання як вхідні дані для Control Systems Toolbox. Команда лінеаризації повертає систему LTI з тією самою кількістю станів, як і кількість станів у дереві механічної системи. Блок «*H_inf*» є регулятором, який показан на рисунку 6, його розгорнута структура показана на рисунку 8. За допомогою датчиків зворотного зв'язку та регулятора здійснюється компенсація похибки між заданим та реальним положенням платформи. Використовуємо Control System Toolbox, щоб синтезувати багато параметричний контролер для платформи Стюарта.

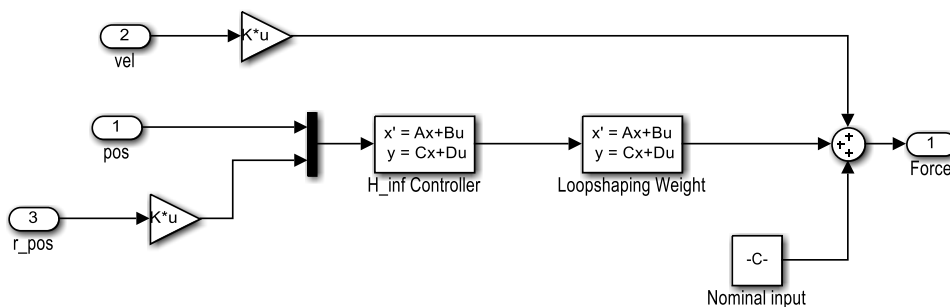


Рис. 8 Модель робастного регулятора
Fig. 8 Robust controller model

Для моделювання системи керування в реальному часі, згідно *третього етапу* технології МОП (рис. 2), необхідно імпортувати її модель в LabVIEW. Імпортувати Simulink модель або MATLAB код в LabVIEW можливо використовуючі Simulation Interface Toolkit (SIT) або Model Interface Toolkit.

Набір засобів LabVIEW SIT забезпечує пряму інтеграцію пакетів MatLab/Simulink та LabVIEW, таким чином, що дозволяє розробляти, прототипувати та тестувати системи керування з використанням моделей, розроблених у середовищі моделювання Simulink. Перетворивши модель Simulink на DLL інструментарій Simulation Interface



Toolkit автоматично генерує код LabVIEW для взаємодії з моделлю Simulink, в результаті чого виходить гнучкий і простий у використанні інтерфейс користувача.

Для імпортування Simulink моделі або MATLAB коду необхідно створити новий проект у LabVIEW, та підключити проект із SolidWorks для імпортування моделі двигунів. Далі був налаштований проект за допомогою функції Properties, щоб прописати додаткові дані щодо проекту та у вкладці Scan Engine у My Computer для запуску скануючого механізму під час компіляції. Завдання цього механізму у тому, щоб опитувати весь час устаткування під час роботи щодо помилки. Для роботи LabVIEW без помилок кроку розрахунку необхідно налаштувати модель, імпортовану з SolidWorks, змінивши крок розрахунку на 0,0001.

Наступним кроком було налаштування осей (Axis) для схеми керування рухом LabVIEW SolidWorks. Дали були співставлені вісі для кожного віртуального двигуна в SoftMotion Axis, що знаходяться в проекті LabVIEW. Також було додано координатну площину, щоб керувати відразу кількома вісями. Тут необхідно включити привод при переході в активний режим, який включає двигун при взаємодії для корекції при передачі даних з LabVIEW в SolidWorks.

При створенні системи керування необхідно додати VI до вже існуючого проекту, в одному вікні якого пишеться програмний код за допомогою візуальної мови програмування G для LabVIEW з додаванням компонентів, діаграм та об'єктів, пов'язаних між собою, на панель, а в іншому вікні додаються елементи системи реального керування двигунами. В цьому ж VI необхідно імпортувати Simulink модель або MATLAB код системи керування платформи Стюарта.

Для керування механічної частини платформи Стюарта, пошуку оптимальної траєкторії і взаємодії з програмою-моделлю (в середовищі SolidWorks) використовується програма, створена в середовищі LabVIEW з інструментарієм NI-SoftMotion, Real Time та FPGA в якості контролера для керування положенням робочої поверхні платформи Стюарта. Це дозволяє розподілити функції та обчислювальні ресурси за різними ЕОМ зі збереженням гнучкості і продуктивності системи в цілому. Модуль SoftMotion забезпечує режим реального часу (RT) та синхронізацію FPGA VI, щоб компенсувати дрейф та джитер між циклом об'єкта RT і цільовим циклом FPGA. Використовуючи модуль NI LabVIEW FPGA, можна створити Target Program апаратного забезпечення низького рівня. Після того, як Target Program була скомпільована, доступ до нього можна отримати за допомогою програми - host в LabVIEW для Windows або LabVIEW в реальному часі за допомогою драйвера NI-RIO.

Для нашого проекту ми вибрали реконфігуроване шасі на базі ПЛІС (FPGA) NI PCI-7833R. Це багатофункціональний RIO серії R з Virtex-II 3M Gate FPGA. R серія це багатофункціональних реконфігурованих пристроїв I/O (RIO), що настроюються, забезпечують повну гнучкість синхронізації та активації системи у програмах для збору даних (DAQ).

Після виконаних маніпуляцій з налаштування в проекті LabVIEW, готовий проект із підключеною збіркою SolidWorks з готовими для неї аналогами осей, які використовуються у створенні інтерфейсу та системи керування. На рисунку 9 та 10 показано програмна реалізація системи керування програмного руху робочої поверхні платформи Стюарта: робочий проект LabVIEW Project Explorer (рис. 9), з доданими складовими з SolidWorks, осями для руху двигунів, контролером для керування драйверами сервоприводів, підсумкова блок-діаграма для керування всіма сервоприводами з контролерами системи, що передбачає під собою графічну мову програмування (рис. 10), та лицьова панель керування (рис. 9).

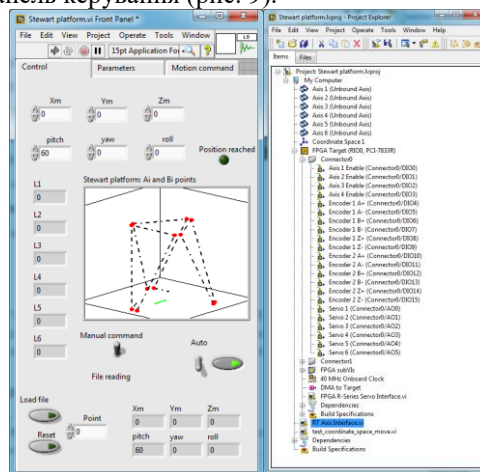


Рис. 9 Лицьова панель та робочий проект LabVIEW Project Explorer
Fig. 9 Front panel and working project LabVIEW Project Explorer

Тепер виконавши всі дії можна виконати оффлайн моделювання та моделювання в реальному часі, згідно *третього етапу* технології МОП. Під час такого моделювання виконується верифікація одержаних результатів шляхом комп'ютерного моделювання в інструментальних середовищах SolidWorks і National Instruments LabVIEW. Проводиться порівняння реалізації траєкторій руху робочої поверхні платформи Стюарта, яка була генерована в SolidWorks без системи керування і з генерованою в LabVIEW із системою керування.

Згідно з *четвертим етапом* технології МОП необхідно виконати реалізацію вбудованого контролера системи



управління. Для цього необхідно здійснити компіляцію проекту LabVIEW.

Модуль LabVIEW FPGA використовує стандартизований у промисловості компілятор Xilinx ISE, розроблений фірмою Xilinx. Насамперед, графічний програмний код LabVIEW FPGA транлюється на набір текстових файлів мовою опису апаратури VHDL.

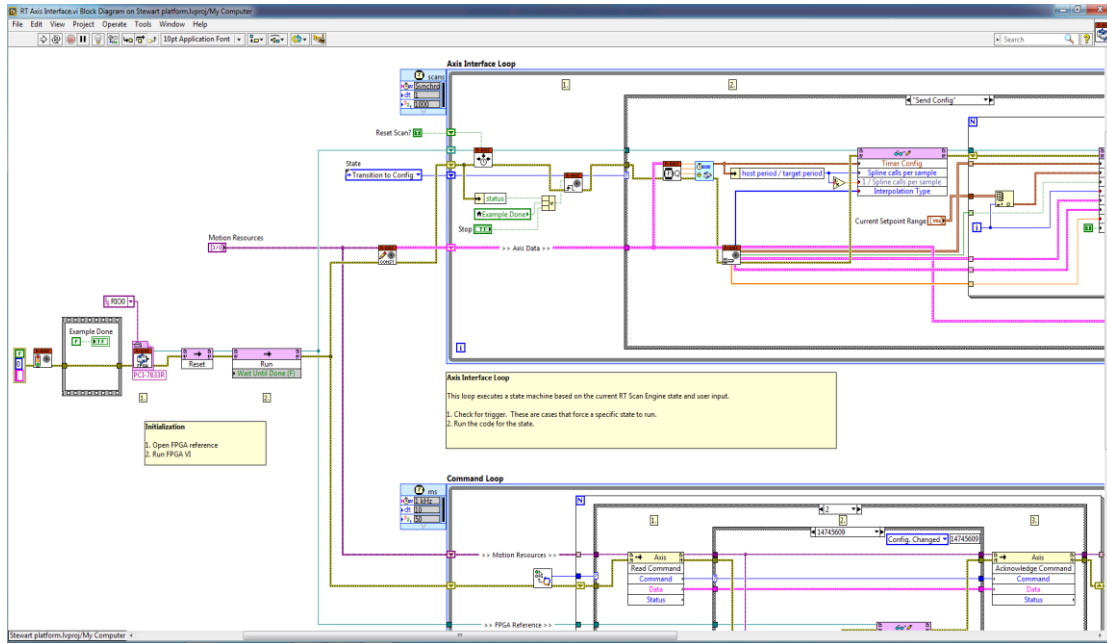


Рис. 10 Блок-діаграма керування сервоприводами і контролерами системи
Fig. 10 Block diagram of control of servo drives and controllers of the system

Компілятор Xilinx ISE виконує оптимізацію VHDL коду та синтез апаратної реалізації проекту. При виконанні зазначених операцій також враховуються практичні обмеження схеми швидкодії, що гарантує ефективність використання ресурсів мікросхеми FPGA. В результаті виходить високоефективна кремнієва реалізація вихідного коду LabVIEW, яка забезпечує паралельну обробку з продуктивністю і надійністю спеціалізованих апаратних засобів. Відсутність операційної системи на кристалі FPGA гарантує максимально можливу швидкодію, надійність за високого ступеня детермінізму.

Важливо відзначити, що розробнику при цьому немає необхідності знати досить складну спеціалізовану текстову або графічну мову проектування схем, особливості налагодження та тестування проектів FPGA, будь-яку іншу класичну мову програмування для вирішення свого прикладного завдання, витратити час на інтеграцію результатів розробки у різних середовищах в єдиний проект.

Висновки. В результаті проведених досліджень запропонована інформаційна технологія на основі методу модельно-орієнтованого проектування, що дозволяє розробити вбудований контролер реального часу керування рухом робочої поверхні платформи Стюарта, з використанням інтелектуальних технологій, підвищити ефективність проектних роботи та поліпшити показники якості з проектованої системи керування.

Поєднання в інформаційну технологію трьох систем Solidworks, яка дозволяє твердотільне моделювання кінематики і динаміки на основі фундаментальних фізичних принципів механіки, Matlab, що має найкращі функціональні можливості при моделюванні динамічних систем, аналізі та синтезі систем керування та Labview, який має найкращі апаратно-програмні засоби проектування вбудованого контролер реального часу, значно розширює функціональні можливості при моделюванні динамічних систем, їх аналізі та синтезі, та значно скорочує час проектування систем керування реального часу.

Перелік використаних джерел

- [1]. Ryssel U., Ploennigs J., Kabitzsch K., Folie M. Generative Design of Hardware-in-the-Loop Models // APGES'07. Salzburg, Austria, 2007. P. 4–11.
- [2]. Зозуля В.А. Класифікація завдань і принципів управління механізмом паралельної кінематичної структури для вирішення різних завдань. / В.А. Зозуля, С.І. Осадчий, Ю. Беляєв, Р. Pawłowski // Міжнародний науково-виробничий журнал «Автоматизація технологічних і бізнес-процесів». – 2018. – Т.10 №2. – С. 18 – 29
- [3]. Model Based Design and HIL Simulations. Kelemenová T., Kelemen M., Miková Ľ. // American Journal of Mechanical Engineering, 2013, Vol. 1, No. 7, P. 276-281
- [4]. Dias, B. M. D.; Laganá, A. A. M.; Justo, J. F.; Yoshika, L. R.; Santos, M. M. D.; Gu, Z. H. (2018). Model-Based Development of an Engine Control Module for a Spark Ignition Engine. IEEE Access 6: 53638-53649.
- [5]. Модельно-ориентированное проектирование программного обеспечения для встраиваемых систем в среде Matlab/Simulink. Топораш Г.К., Мазур А.В, Ковальчук Д. А., Пушкин А.А. Автоматизація технологічних і бізнес-процесів № 17/2014 ст. 26-29



[6]. W. Banaś, A. Sękala, A. Gwiazda, K. Foit, G. Kost, Stewart platform simulation using the LabView environment, Archives of Materials Science and Engineering 75/2 (2015) 82-88.

References

- [1]. Ryssel U., Ploennigs J., Kabitzsch K., Folie M. Generative Design of Hardware-in-the-Loop Models. APGES'07. Salzburg, Austria, 2007. pp. 4–11.
- [2]. Zozulya V.A., Osadchiy S.I., Belyaev Yu., Pawłowski P. (2018). Klasifikatsiya zavdan' i printsipiv upravlinnya mekhanizmom paralel'noï kinematichnoï strukturi dlya virishennya riznikh zavdan' [Classification of tasks and principles of controlling the mechanism of parallel kinematic structure]. Mizhnarodniy naukovo-virobnichiy zhurnal «Avtomatizatsiya tekhnologichnikh i biznes-protseviv». T.10 №2. pp. 18 – 29. (in Ukrainian)
- [3]. Kelemenová T., Kelemen M., Miková Ľ. (2013) Model Based Design and HIL Simulations. American Journal of Mechanical Engineering. Vol. 1, No. 7, pp. 276-281
- [4]. Dias, B. M. D.; Laganá, A. A. M.; Justo, J. F.; Yoshika, L. R.; Santos, M. M. D.; Gu, Z. H. (2018). Model-Based Development of an Engine Control Module for a Spark Ignition Engine. IEEE Access 6: 53638-53649.
- [5]. Toporash G.K., Mazur A.V, Koval'chuk D. A., Pushkin A.A. (2014) Model'no-oriyentirovannoye proyektirovaniye programmnoho obespecheniya dlya vstraivayemykh sistem v srede Matlab/Simulink [Model-Based Software Design for Embedded Systems in Matlab/Simulink]. Avtomatizatsiya tekhnologichnikh í biznes-protseviv № 17, pp. 26-29. (in Ukrainian)
- [6]. W. Banaś, A. Sękala, A. Gwiazda, K. Foit, G. Kost, (2015) Stewart platform simulation using the LabView environment, Archives of Materials Science and Engineering 75, vol. 2 82-88.