

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»

Приладобудівний факультет

(повна назва інституту/факультету)

Кафедра виробництва приладів

(повна назва кафедри)

«На правах рукопису»  
УДК 519.682:621.7

«До захисту допущено»

В.о. завідувача кафедри

\_\_\_\_\_ В.В. Шевченко  
(підпис) (ініціали, прізвище)

“ \_\_\_\_\_ ” \_\_\_\_\_ 2018 р.

## Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 151– Автоматизація та комп'ютерно-інтегровані технології  
(код і назва)

на тему: Імітаційне моделювання виробничих систем в приладобудуванні

Виконав : студент II курсу, групи ПБ-61м  
(шифр групи)

\_\_\_\_\_ Нестеренко Андрій Олегович \_\_\_\_\_

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник доцент, к.т.н., доцент Вислоух С.П. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант стартап-проекту к.е.н., доцент Бояринова К.О. \_\_\_\_\_  
(назва розділу) (науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Засвідчую, що у цій магістерській  
дисертації немає запозичень з праць інших  
авторів без відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

Київ – 2018 року

**Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»**

Факультет Приладобудівний  
(повна назва)

Кафедра Виробництва приладів  
(повна назва)

Рівень вищої освіти – другий (магістерський)

Спеціальність 151 – Автоматизація та комп'ютерно-інтегровані технології  
(код і назва)

ЗАТВЕРДЖУЮ

В.О. завідувача кафедри

Шевченко В.В.  
(підпис) (ініціали, прізвище)

«\_\_» \_\_\_\_\_ 2018 р.

**ЗАВДАННЯ  
на магістерську дисертацію студенту**

Нестеренку Андрію Олеговичу

(прізвище, ім'я, по батькові)

1. Тема дисертації «Імітаційне моделювання виробничих систем в приладобудуванні»

науковий керівник дисертації Вислоух Сергій Петрович, к.т.н., доцент,  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «23» березня 2018 р. № 1006-с

2. Термін подання студентом дисертації 10 травня 2018 р.

3. Об'єкт дослідження виробничі технологічні системи в приладобудуванні

4. Предмет дослідження Методи імітаційного моделювання виробничих систем

5. Перелік завдань, які потрібно розробити Виконати аналіз стану моделювання роботи виробничих систем та постановка задач дисертаційних досліджень. Розглянути основи побудови та моделювання роботи виробничих систем, вибрати ефективних метод моделювання технологічних систем в приладобудуванні. Розглянути питання використання системи AnyLogic для опису роботи і моделювання виробничих систем в приладобудуванні. Розробити методику практичного застосування системи AnyLogic на прикладах моделювання технологічного процесу виготовлення деталі та складання приладу. Розробити стартап-проект реалізації методики імітаційного моделювання виробничих систем.

6. Орієнтовний перелік ілюстративного матеріалу Стан питання моделювання та оптимізації роботи виробничих систем. Мета роботи, об'єкт та предмет дослідження. Наукова новизна та практична корисність. Методи моделювання виробничих систем.

Класифікація систем імітаційного моделювання. Особливості побудови та використання виробничих систем в приладобудуванні. Алгоритм організації імітаційного моделювання виробничих систем. Математичні моделі роботи виробничих систем. Порівняльний аналіз засобів імітаційного моделювання виробничих систем в приладобудуванні. Система AnyLogic та особливості її застосування. Методика моделювання складних систем засобами AnyLogic. Математична модель технологічного процесу виготовлення деталі та її дослідження. Математична модель процесу складання приладу та його дослідження. Методика практичного застосування системи AnyLogic для моделювання роботи виробничих систем в приладобудуванні. Стартап-проект реалізації імітаційного моделювання складних систем.

7. Орієнтовний перелік публікацій Статті та тези доповідей за темою магістерської дисертації (надано в Додатку в вигляді Форми 2б) \_\_\_\_\_

8. Консультанти розділів дисертації\*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Стартап-проект	Бояринова К.О., к.е.н., доцент		

9. Дата видачі завдання \_\_\_\_\_ 16.03.2018 р.

#### Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Виконати аналіз стану питання математичного моделювання та оптимізації роботи виробничих систем та поставити задачі дисертаційних досліджень	30.03.2018 р.	
2	Розробити основи побудови та моделювання роботи виробничих систем.	09.04.2018 р.	
3	Огляд системи імітаційного моделювання, методів і засобів моделювання складних систем та вибір ефективного методу для застосування	16.04.2018 р.	
4	Розробка методики імітаційного моделювання виробничих систем засобами <u>AnyLogic</u>	23.04.2018 р.	
5	Практичне застосування використання розробленої системи	30.04.2018 р.	

Студент \_\_\_\_\_

(підпис)

А.О.Нестеренко.

(ініціали, прізвище)

Науковий керівник дисертації \_\_\_\_\_

(підпис)

С.П. Вислоух

(ініціали, прізвище)

\_\_\_\_\_

## АНОТАЦІЯ

Магістерська дисертація складається зі вступу, п'яти розділів, висновків, списку використаної літератури та додатків у вигляді тексту програм, результату програм, технологічної документації, списку наукових праць та акту впровадження. Основний текст складається зі 122 сторінок, включає в себе 27 таблиць і 25 рисунків.

У вступі обґрунтована актуальність проблеми, сформовані мета і задачі роботи, наукова новизна та практична корисність, основні положення, що винесені на захист. Приведена коротка характеристика роботи.

У першому розділі дисертації представлений огляд стану моделювання роботи виробничих систем, розглянуто поняття виробничої системи та засоби математичного моделювання складних систем, а також проведена постановка задач дисертаційних досліджень.

У другому розділі наведено особливості побудови та використання виробничих систем, описані їхні методи формалізації та проведено вибір ефективного методу їх опису та моделювання.

У третьому розділі детально досліджено систему AnyLogic та особливості її застосування, розглянуто опис складних систем засобами системи AnyLogic. Описано динаміку моделювання роботи виробничої системи засобами AnyLogic. Описано переваги використання системи AnyLogic.

У четвертому розділі описано практичне використання розробленої системи. Описано етапи створення імітаційного моделювання роботи виробничої дільниці механічного цеху при виготовленні деталі та етап створення імітаційного моделювання роботи виробничої дільниці складального цеху при складанні хвильового редуктора, а також приведено результати комп'ютерного експерименту та рекомендацій щодо покращення.

У п'ятому розділі менеджменту стартап-проекту проведено опис проекту, технологічний аудит ідеї проекту, аналіз ринкових можливостей запуску стартап-проекту та розроблення ринкової стратегії проекту.

Ключові слова: приладобудування, AnyLogic, імітаційне моделювання, виробнича система, комп'ютерний експеримент, модель, складна система, система моделювання.

## ABSTRACT

The master's dissertation consists of an introduction, five sections, conclusions, list of used literature and appendices in the form of program text, program results, technological documentation, list of scientific works and an act of introduction. The main text consists of 122 pages, includes 27 tables and 25 drawings.

The introduction substantiates the relevance of the problem, the purpose and tasks of the work, the scientific novelty and practical utility, the basic provisions that have been made for protection. A brief description of the work is given.

The first section of the dissertation presents an overview of the state of the modeling of production systems, the concepts of the production system and the means of mathematical modeling of complex systems are considered, as well as the tasks of the dissertation research are presented.

The second section presents the peculiarities of the construction and use of production systems, describes their formalization methods, and chooses an effective method for their description and modeling.

The third section explores in detail the system AnyLogic and the peculiarities of its application, describes the description of complex systems by means of the system AnyLogic. The dynamics of the modeling of the production system by means of AnyLogic is described. The dynamics of the modeling of the production system by means of AnyLogic is described. The advantages of using AnyLogic system are described.

The fourth section describes the practical use of the developed system. The stages of creation of simulation modeling of the production section of the mechanical shop at the manufacture of the part and the stage of creation of simulation modeling of the production area of the assembly shop during the assembly of the wave reducer are described, as well as the results of the computer experiment and recommendations for improvement.

In the fifth section of the startup project management, a description of the project, a technological audit of the project idea, an analysis of market opportunities for launching a startup project and the development of a market strategy of the project has been conducted.

Key words: instrument making, AnyLogic, simulation modeling, production system, computer experiment, model, complex system, modeling system.

## ЗМІСТ

СПИСОК УМОВНИХ ПОЗНАЧЕНЬ.....	8
ВСТУП.....	9
РОЗДІЛ 1. АНАЛІЗ СТАНУ МОДЕЛЮВАННЯ РОБОТИ ВИРОБНИЧИХ СИСТЕМ ТА ПОСТАНОВКА ЗАДАЧІ ДИСЕРТАЦІЙНИХ ДОСЛІДЖЕНЬ.....	13
1.1 Виробнича система як складна система та особливості її моделювання...13	
1.2 Засоби математичного моделювання складних систем.....20	
1.3 Стан математичного моделювання роботи виробничих систем в приладобудуванні.....30	
1.4 Постановка задачі дисертаційних досліджень.....33	
РОЗДІЛ 2. ОСНОВИ ПОБУДОВИ ТА МОДЕЛЮВАННЯ РОБОТИ ВИРОБНИЧИХ СИСТЕМ.....	35
2.1 Особливості побудови та використання виробничих систем.....35	
2.2 Методи формалізації та моделювання роботи виробничих систем.....43	
2.3 Вибір ефективного методу опису та моделювання виробничих систем...48	
Висновки до розділу.....50	
РОЗДІЛ 3. ВИКОРИСТАННЯ СИСТЕМИ ANYLOGIC ДЛЯ ОПИСУ РОБОТИ ТА МОДЕЛЮВАННЯ РОБОТИ ВИРОБНИЧИХ СИСТЕМ.....	52
3.1. Система AnyLogic та особливості її застосування.....52	
3.2. Опис складних систем засобами системи AnyLogic.....55	
3.3. Моделювання динаміки роботи виробничих систем за допомогою системи AnyLogic.....60	
Висновки до розділу.....68	
РОЗДІЛ 4. ПРАКТИЧНЕ ВИКОРИСТАННЯ РОЗРОБЛЕНОЇ МЕТОДИКИ ТА СИСТЕМИ ДЛЯ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ ВИРОБНИЧИХ ПРОЦЕСІВ В ПРИЛАДОБУДУВАННЯ.....	70

4.1. Імітаційне моделювання роботи виробничої дільниці механічного цеху при виготовленні деталі.....	70
4.2. Імітаційне моделювання роботи виробничої дільниці складального цеху при складанні хвильового редуктора.....	81
4.3. Методичні рекомендації зі створення імітаційної моделі виробничої системи та проведення відповідного комп'ютерного експерименту.....	91
Висновки до розділу.....	92
<b>РОЗДІЛ 5. РОЗРОБКА СТАРТАП-ПРОЕКТУ «ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ВИРОБНИЧОЇ СИСТЕМИ В ПРИЛАДОБУДУВАННІ».....</b>	
5.1. Опис ідеї проекту .....	94
5.2. Технологічний аудит ідеї проекту.....	96
5.3 Аналіз ринкових можливостей запуску стартап-проекту .....	96
5.4. Розроблення ринкової стратегії проекту .....	103
5.5. Розроблення маркетингової програми стартап-проекту .....	105
Висновки до розділу.....	108
ЗАГАЛЬНІ ВИСНОВКИ ДО МАГІСТЕРСЬКОЇ ДИСЕРТАЦІЇ.....	109
СПИСОК ЛІТЕРАТУРИ.....	112
ДОДАТКИ .....	122
ДОДАТОК А Креслення корпусу та технологічний процес його виготовлення	
ДОДОТОК Б. Креслення хвильового редуктора та технологічний процес його складання	
ДОДАТОК В. Інформаційна модель виготовлення корпусу та її програмний код	
ДОДАТОК Г. Інформаційна модель складання хвильового редуктора та її програмний код	

## СПИСОК УМОВНИХ ПОЗНАЧЕНЬ

ВС – виробнича система;

ГВС – гнучка виробнича система;

АТСС – автоматизована транспортно-складська система;

ЧПК – числове програмне керування;

ЕОМ – електронно обчислювана машина;

ГВМ – гнучкий виробничий модуль;

АСУ ТП – автоматизовані системи управління технологічними процесами;

ТП – технологічний процес;

ІМ – імітаційне моделювання;

ПЗ – програмне забезпечення;

БД – бази даних;

СМ – система моделювання;

САПР – система автоматизованого проектування;

ІМ – Імітаційне моделювання.

## ВСТУП

**Актуальність роботи.** Однією з умов успішного функціонування сучасного підприємства є використання виробничих систем (ВС). Основне їхнє завдання полягає в максимальному зниженні тимчасових затримок на різних етапах виробництва і в здатності підлаштуватися під мінливі внутрішні і зовнішні умови, що впливають на виробництво. Для проектування, аналізу та дослідження ВС використовується віртуальне моделювання. Пов'язано це з тим, що проведення експериментів на реальних об'єктах вимагає великих фінансових і трудових витрат.

Імітаційне моделювання дає можливість описувати поведінку ВС за допомогою динамічних моделей. Однак збільшення розміру досліджуваної системи підвищує складність її моделювання, загальне розуміння внутрішніх взаємодій різних процесів погіршується, внесення змін вимагає великих тимчасових витрат. При цьому повторне використання моделі або її частини, у разі модернізації виробництва або аналізу його поточного стану, є трудомістким завданням. Рішенням цих проблем є можливість представлення складних систем із застосуванням декомпозиції, поділяючи систему на окремі процеси в вигляді набору бібліотечних компонентів, і автоматизованої підтримка актуальних даних про модельовані об'єкти.

Крім ефективної побудови моделі, необхідна організація її взаємодії з реальними об'єктами для оперативного отримання інформації про зміни різних параметрів, що дозволить заздалегідь спрогнозувати можливі наслідки застосування тих або інших дій.

У зв'язку з вищесказаним актуальною є проблема складності аналізу та використання ВС.

**Об'єкт дослідження** – виробничі системи в приладобудуванні.

**Предмет дослідження** – методи імітаційного моделювання виробничих систем.

**Мета роботи** – підвищення ефективності аналізу та використання ВС на основі розробки методик і алгоритмів імітаційного моделювання засобами системи AnyLogic.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- проаналізувати особливості побудови та використання виробничих систем;
- розглянути стан математичного моделювання роботи виробничих систем в приладобудуванні;
- розглянути методи формалізації та моделювання роботи виробничих систем;
- провести вибір ефективного методу опису та моделювання виробничих систем;
- надати особливості моделювання засобами AnyLogic та особливості її застосування;
- провести опис складних систем засобами системи AnyLogic;
- виконати імітаційне моделювання роботи виробничої ділянки механічного цеху при виготовленні деталі;
- виконати імітаційного моделювання роботи виробничої ділянки складального цеху при складанні хвильового редуктору;
- надати методичні рекомендації, щодо розробки методики створення імітаційних математичних моделей роботи ВС;
- виконати впровадження методики імітаційного моделювання виробничих систем в приладобудуванні.

**Методи дослідження** – поставлені завдання у дисертаційній роботі вирішувалися з використанням методів технології приладобудування, системного аналізу, теорії масового обслуговування, а також методів імітаційного моделювання і програмування складних систем.

**Наукова новизна** роботи полягає в наступному:

- створено методичку імітаційного моделювання складних виробничих систем;
- створено математичну модель технологічного процесу виготовлення деталі «корпус»;
- створено математичну модель технологічного процесу складання хвильового редуктора.

**Практична корисність** дисертаційної роботи є такою:

- розроблено алгоритм методу імітаційного моделювання з використанням системи AnyLogic;
- визначено оптимальне завантаження роботи обладнання при виготовленні деталі «корпус» шляхом проведення комп'ютерного експерименту;
- визначено оптимальне завантаження роботи обладнання при складанні хвильового редуктора шляхом проведення комп'ютерного експерименту;
- надано методичні рекомендації з використання системи AnyLogic при моделюванні виробничих систем в приладобудуванні.

**Практична апробація.** Основні положення та результати роботи доповідалися та обговорювалися на конференціях з публікацією тез доповідей: VII міжнародна науково-практична конференція (КЗЯТПС – 2017), м. Чернігів, 24–27 квітня 2017 р.; VI Міжнародна науково-технічна конференція «Датчики, прилади та системи – 2017», м. Черкаси, 2017 р.; XIII Міжнародна науково-практична конференція студентів, аспірантів та молодих вчених «Ефективність інженерних рішень у приладобудуванні», м. Київ, НТУУ «КПІ», 12 квітня 2017р.; VII Міжнародна науково-практична конференція молодих вчених та студентів «Технічне регулювання, метрологія та інформаційні технології», м. Одеса, 10–11 жовтня 2017 р.; X Міжнародна науково-технічна конференція “Приборостроения-2017”,

м. Мінськ, 1–2 листопада 2017 р.; Міжнародна науково-технічна конференція «Материалы, оборудование и ресурсосберегающие технологии» м. Могилів, 27–28 квітня 2017 р.; XX Міжнародна молодіжна науково-практична конференція «Людина і космос», м. Дніпро, НЦАО, 11-13 квітня 2018р.; VIII Міжнародна науково-практична конференція «Комплексне забезпечення якості технологічних процесів та систем – 2018», м. Чернігів, 10–12 травня 2018 р.

За темою магістерської дисертації опубліковано наукову статтю у науковому журналі «Молодий вчений» м. Херсон 2018р., випуск №3(55), а також опубліковано статтю у збірнику XIII Міжнародної науково-практичної конференції студентів, аспірантів та молодих вчених «Погляд у майбутнє приладобудування», м. Київ, НТУУ «КПІ», 15-16 травня 2018р.

## РОЗДІЛ 1

### АНАЛІЗ СТАНУ МОДЕЛЮВАННЯ РОБОТИ ВИРОБНИЧИХ СИСТЕМ ТА ПОСТАНОВКА ЗАДАЧІ ДИСЕРТАЦІЙНИХ ДОСЛІДЖЕНЬ

1.1. Виробнича система як складна система та особливості її моделювання

Промислова революція кінця XVIII — початку XIX ст. відкрила нову епоху в розвитку теорії та практики організації виробництва. Паровий двигун як джерело енергії для машин дозволив організувати машинне фабричне виробництво. Виникла необхідність в узгоджених діях усіх його ланок за допомогою встановлення й суворого виконання певних норм та пропорцій між усіма сторонами виробництва. Потреби в жорсткій регламентації спонукали визначення норм виробітку й оплати праці, максимальної швидкості роботи устаткування, оптимальних обсягів виготовлення продукції на основі вдосконалення організації виробництва й праці [1].

Короткий огляд стадій (етапів) розвитку організації та управління виробництвом, розглянуто авторами у своїх творах [1-4], починаючи з часів А. Сміта.

Забезпечення необхідних умов життєдіяльності людей, вимагає постачання багатьох продуктів вжитку та користування. Розвиток та сучасний стан життєвого показника завдячують саме виробничій діяльності, яка направлена на забезпечення необхідних потреб людини.

За визначенням В.Г. Василькова, «виробництво»— це цілеспрямована діяльність зі створення будь-якого корисного продукту (товари, предмети, речі, послуги, інформація, знання тощо). Виробництво можна охарактеризувати як систему, де здійснюється цілеспрямований процес перетворення вхідних елементів (сировини, матеріалів) у корисну продукцію. У творі [2], автор розглядає теоретичні та практичні засади організації промислового виробництва, а також описує особливості формування

виробничих систем, організаційні аспекти виробничих і трудових процесів, їх просторово-часової єдності, методи виробництва, роль і значення його обслуговування.

Вивчення різних факторів та чинників пов'язаних між собою, які гарантують успішне виконання виробництва та його функцій, вимагають розглядати їх з позиції системного підходу, як виробничу систему.

Існує багато визначень, пояснюючих що ж таке виробнича система (ВС). Градов А.П. у своїй роботі [5], визначає виробничу систему, як частину виробничого процесу, що відокремилася в результаті суспільного розподілу праці, яка здатна самостійно або взаємодіючи з іншими аналогічними системами задовольняти ті чи інші потреби і запити споживачів за допомогою вироблених цією системою товарів (виробів, послуг).

За визначенням Васильєва С.В. [6], виробнича система – цілеспрямований процес, завдяки якому відбувається перетворення окремих елементів системи в корисну продукцію.

Синиця Л.М. [7] описує виробничу систему, як особливий клас систем, що включає працівників, знаряддя, предмети праці та інші елементи, необхідні для функціонування системи, в процесі якого створюються продукція і послуги.

В свою чергу, Тальнов Ю.Н. та Нетішина С.І. описують виробничу систему, як складну систему взаємопов'язаних елементів виробничого процесу, що утворюють єдине ціле і функціонують з метою виробництва промислової продукції або надання послуг [8].

Кононова В.Ю. [9] приводить наступне визначення: виробнича система підприємства являє собою спосіб організації виробничих процесів, таких як управління закупівлями, організація робочих операцій, управління матеріальними потоками на виробництві, обслуговування устаткування, управління якістю і т. д. Виробнича система охоплює всі стадії виробничого і збутової діяльності підприємства, оскільки від її ефективності залежать

продуктивність роботи підприємства, якість продукції і, в кінцевому рахунку, конкурентоспроможність виробництва.

Володимир Речкалов у своїй статті [10] описує виробничу систему, як сукупність методів, процедур і планів, що включають в себе всі функції, які необхідні для переробки інформації та сировини на вході в готові товари або послуги на виході.

Всі наведені визначення авторів описують виробничу систему в загальній характеристиці, як систему взаємопов'язаних елементів.

Різні системи, залежно від тих чи інших умов, класифікують за окремими ознаками. Системи можуть бути природні або штучні, відкриті чи закриті. Системи в яких сукупність різних об'єктів мають речову форму та взаємодіють між собою називають фізичними, а символічне зображення процесів чи зв'язків, що виконуються у фізичних моделях, визначаючи їх поведінку, називають абстрактними системами.

Систему називають простою, якщо вона має обмежену кількість елементів, що забезпечують функціонування системи, або складною, якщо вона складається з великої кількості елементів, які вирішують складні задачі для її функціонування. Особливістю складної системи є її можливий розподіл на частини, елементи, підсистеми та інші складові.

Також, у джерелі [11] автор описує поняття централізованої та децентралізованої системи, поняття динамічних та статичних систем, а також систем зі зворотнім зв'язком. Автором надані визначення стабільних систем та систем, що складаються з комплексу взаємозалежних об'єктів. Як приклад, зображена складна система як вся діяльність підприємства, яка утворена з менш складних систем (рис.1.1).

Складовою частиною системи являються люди, а також різні об'єкти. На найменшому рівні така система може досліджуватися як група механізмів, які обслуговуються людиною, тобто працівником. Кожний робітник що

обслуговує механізм, являє собою, разом з ним два взаємопов'язаних та взаємозалежних елементи, які утворюють систему «людина — машина».

Елемент виробничої системи – складова частина системи, яка не розкладається на дрібніші складові. Найменшими елементами виробничої системи є робочі місця, що оснащені інструментами та пристосуваннями.

Васильков В.Г. описує в своїй роботі [2], що інтеграція первинних систем «людина – машина» створює виробничу дільницю – складну систему, яка охоплює основних і допоміжних робітників, основне і допоміжне устаткування, функціональні підсистеми зі складним комплексом взаємозв'язків, взаємовідносин та інтересів, що й зумовлює її складну структуру та організацію, а також дає пояснення щодо систем вищого рівня: «До систем вищого рівня належать цехи, підприємства, галузі. При цьому кожна ланка системи, підсистеми будь-якого рівня відбиває найістотніші риси системи вищого рівня, частиною якого вони є» [2].

Структура ВС являє собою постійний зв'язок між сукупністю елементів, які виконують необхідні функції і зберігають систему цілісною під час різноманітних впливів (зовнішніх або внутрішніх).

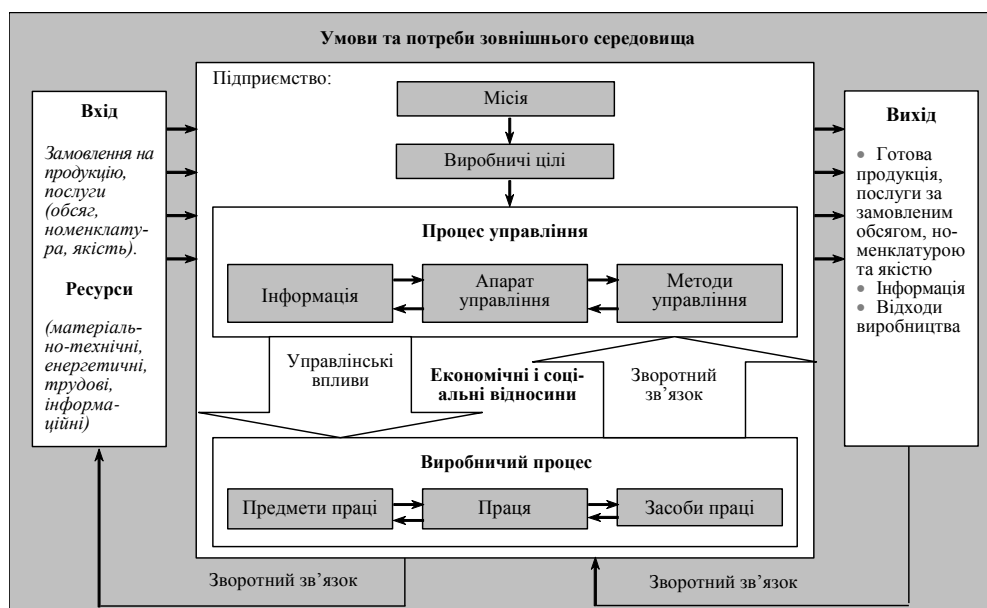


Рис. 1.1. Підприємство як складна виробнича система.

Структура ВС оцінюється взаємозв'язками та складом її елементів і підсистем. Розрізняють часову та просторову структури ВС. Вони взаємозалежні і тісно пов'язані між собою.

Структура ВС зображена на рис.1.2. На цьому рисунку показано елементи системи та зв'язки між ними. Залежно від цілей та аналізу ВС може бути зображена різноманітними схемами.

Автор Васильков В.Г. пояснює, що єдність ВС є однією з основних властивостей. Всі елементи ВС працюють на виконання однієї мети — проектування, розроблення та виготовлення необхідної продукції. Всі ВС мають вхід, процес, вихід і зворотний зв'язок.

За допомогою пристрою входження в ВС поступають вихідні ресурси, які забезпечують функціонування системи. Цей процес являється центральним та основним компонентом ВС, завдяки якому ресурси входження змінюються і отримують зовсім інші нові властивості, які вони отримують на виході. Вихід ВС є результатом функціонування системи, він може бути окремим виробом, послугою, інформацією чи всім одночасно залежно від спеціалізації виробничої системи.

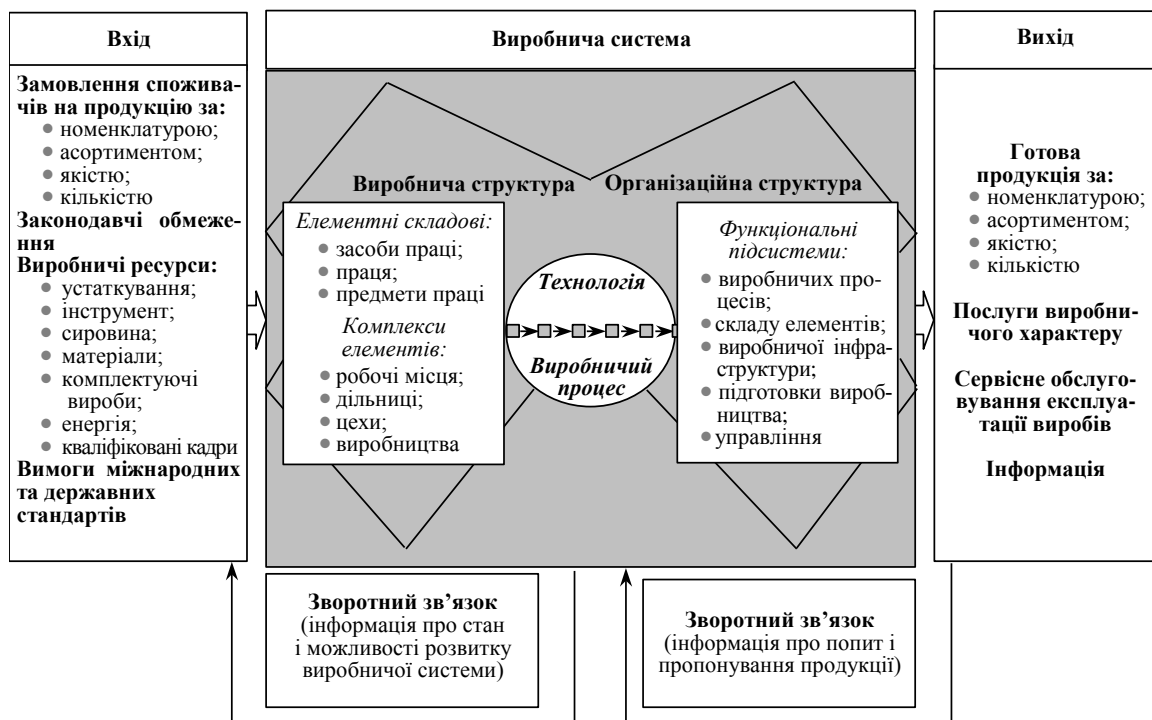


Рис. 1.2. Структура виробничої системи.

ВС, не зважаючи на їх багату різноманітність, залежно від типу виробництва, виду діяльності та галузевих особливостей, мають декілька загальних особливостей, які відрізняють їх від систем інших класів та описують своєрідність законів, принципів функціонування та розвитку. Найбільш істотними із них є: цілеспрямованість ВС; поліструктурність ВС; відкритість ВС; складність ВС; різноманітність ВС.

Саме ці особливості визначають раціональність форм організації виробничих систем та їх підсистем, що відрізняються у більшості випадків характером зв'язків між елементами.

Під час проектування та вдосконалення ВС їм надають певні властивості: результативність; надійність; гнучкість; керованість; довготривалість; структура; організація виробництва; рівень організації ВС.

Система, яка містить функціональні підсистеми, поміж яких основним є колектив робітників відповідних категорій, розглядається як підприємство.

У єдиній системі підприємства виділяють ієрархічні, функціональні, кібернетичні підсистеми.

Всі підприємства являються складними ієрархічними системами, які мають такі складові: робоче місце, дільниця, цех, виробництво (рис. 1.3) [3].

В рамках підсистеми проходять певні види діяльності. Їх самостійність, визначеність цілей та зміст надають їм можливості інтегруватися у наступні функціональні підсистеми: організація виробничих процесів; елементна складова виробництва; виробнича інфраструктура підприємства; управлінська підсистема підприємства.

Кожен етап та функціональні підсистеми підприємства мають «вхід», «процес» і «вихід». Все це обумовлює наявність у них об'єктів і суб'єктів управління, що пов'язані між собою каналами зв'язку.

З врахуванням тенденцій відповідності вимогам сучасного розвитку техніки всі створювані ВС повинні задовольняти вимогам швидкої заміни

продукції, що підлягає виготовленню. Комплекс характеристик таких систем, які б відповідали цим вимогам, визначається терміном гнучкість.

Гнучкість виробничої системи (ГВС) можна досягти використанням методів регулювання окремих параметрів процесу при збереженні моноблокової конструкції її функціональних елементів, але лише в межах технологічного ряду виробів [12].

ГВС класифікуються сферами використання, рівнем автоматизації, гнучкістю, функціональним призначенням системи керування обладнанням, розвитком структури і типом конфігурації автоматизована транспортно-складська система (АТСС) [12-14].

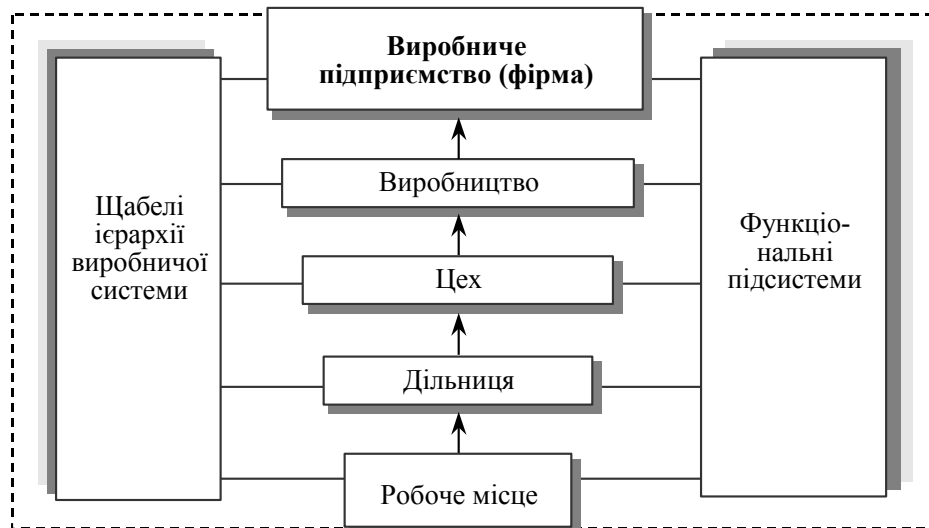


Рис. 1.3. Ієрархічна структура виробничої системи.

При проектуванні ВС перед розробником постають проблеми та недоліки створюваної системи. До недоліків можна віднести використання конкретної ВС не за призначенням або створення такої ж системи без врахування окремих елементів, що в подальшому вплине на функціонування системи та зв'язків між її елементами.

Для впровадження ВС у реальне виробництво, необхідно провести оптимізацію системи, тобто підготувати її до можливих ризиків, створити необхідні елементи та зв'язки між ними та налаштувати її до поставлених

перед нею задач. Перспективним методом оптимізації роботи виробничої системи є моделювання як ефективна та технологічно обґрунтована стратегія її подальшого функціонування.

## 1.2. Засоби математичного моделювання складних систем

На сьогодні моделювання виробничих систем застосовують у багатьох галузях промисловості. Для його реалізації існує множина різноманітних засобів. Задля забезпечення розвитку промисловості необхідно проводити не тільки модернізацію обладнання та розвивати автоматизовані системи управління технологічними процесами, але й розробляти засоби оптимізації та аналізу ТП. Таким інструментом, що дозволяє вирішити дані задачі, є моделювання.

Розглянемо деякі засоби моделювання, що презентовано на ринку сучасних інформаційних технологій.

### 1.2.1 Система моделювання ARIS Toolset

Система ARIS [16, 17] - це «концепція» та сімейство програмних продуктів, що базується на ній, які розроблені компанією IDS Scheer AG (Німеччина) для докладного опису, аналізу, подальшого вдосконалення бізнес-процесів підприємства та управління ними.

У своїй книзі [18], автор описує та характеризує дану систему моделювання як систему, що має потужний графічний інтерфейс, а також характеризує наявністю великої кількості стандартних об'єктів для опису бізнес-процесів.

Перевагою сімейства програмних продуктів ARIS є [19]: добре розвинений графічний інтерфейс; підтримка потужного сховища даних;

інтеграція з іншими програмними продуктами; динамічне моделювання; невисокі вимоги до апаратного та програмного забезпечення.

Недоліком даної системи є цінова політика компанії, що визначена розробниками, яка розрахована на європейський ринок технологій. Це значно зменшує можливість використання її на ринку України власниками підприємств та виробничих організацій, що реалізують одиничне та дрібносерійне виробництво.

### 1.2.2. Система моделювання ITHINK

Система ITHINK є інструментом управління і планування та засобом аналізу ситуації, що розроблена компанією High Performance Systems (США) [20].

ITHINK – це компактний, об'єктно-орієнтований пакет прикладних програм з Desktop-інтерфейсом, що забезпечує графічну, обчислювальну та інформаційну підтримку процедур багаторівневого системного аналізу складних процесів організації управління, бізнесу, фінансів, політики тощо.

За допомогою пакету ITHINK можна створити модель виробничо-збутового циклу підприємства, розпочинаючи від закупівлі сировини та закінчуючи його виробництвом і реалізацією. В сферу потенційного застосування системи входить і транспорт, мережі газо- і водопостачання та інші розподільні системи обслуговування виробництва.

Перевагами системи ITHINK є її функціональна простота та її гармонійність. Пакет ITHINK не вимагає спеціальних навичок і володіння складними математичними методами. Також пакет ITHINK не пред'являє великих вимог до використовуюваного апаратного забезпечення (процесор 486 або більш потужний, 8 Мб RAM, Windows 95 та наявність 13 Мб вільної пам'яті на жорсткому диску).

Недоліком пакету ITHINK є менша кількість реалізованих функцій в порівнянні з іншими системами моделювання.

### 1.2.3 Система Powersim Studio

Програмний пакет PowerSim Studio створений і поширюється фірмою Powersim Software AS (Норвегія) [21]. Використовувана в системі методологія моделювання побудована на базі класичних методів системної динаміки, що створені Дж. Форрестером. Пакет є додатком таких відомих систем SAP, SEM, BPS, але також може використовуватися як автономний додаток. Пакет має розвинені засоби візуального програмування і різні розширені можливості, в тому числі вбудовані блоки аналізу ризиків та оптимізації процесів.

Перевагами даної системи є: інтегрованість з будь-якими зовнішніми системами; можливість внесення змін в модель власними силами; можливість реалізації моделей під WEB-інтерфейс; можливість проектування ієрархічних (вкладених) моделей.

Недоліком даної системи є те, що робота з програмою оптимізації вимагає певної підготовленості користувача – користувач повинен мати уявлення про вигляд цільових функцій і критеріїв оптимізації.

### 1.2.4 Програмний пакет Extend

Пакет Extend – універсальний пакет імітаційного моделювання процесів модернізації і обслуговування, розроблений компанією Imagine That. Inc. (США) [21].

Пакет Extend має потужний графічний інтерфейс, що дозволяє створювати схеми процесів і виробляти імітаційні експерименти. Тут є можливість перегляду моделей у вигляді графіків, а також використовувати 2D та 3D анімацію.

Extend – пакет моделювання дискретних процесів, що застосовується для моделювання, аналізу і покращення будь-якого процесу, який може бути представлений у вигляді «блок-схем». Extend дозволяє створювати динамічні моделі різнорідних процесів і систем в термінах предметної області та оптимізувати побудовану модель. Він дає можливість створювати стохастичні динамічні моделі будь-якого підприємства. Динамічні моделі дозволяють оптимізувати, прогнозувати, планувати діяльність підприємств, а також проводити аналіз їх роботи на підставі отриманих моделей і видавати рекомендації щодо вдосконалення роботи конкретного підприємства.

#### 1.2.5 Засіб моделювання GPSS/H

Система GPSS/H – засіб для моделювання дискретних систем, що розроблена компанією Wolverine Software (США) [22]. В основі цього програмного продукту покладена мова імітаційного моделювання GPSS (General Purpose Simulating System – загально-цільова система моделювання).

Цей засіб надає користувачеві можливість створювати і «відчувати» імітаційні моделі різних за своєю фізичною природою і призначенню систем.

Основними поняттями мови GPSS є транзакт, блок та оператор. Транзакт GPSS – це динамічний об'єкт, під яким мають на увазі клієнт, вимога, виклик або заявка на обслуговування приладом обслуговування. Транзакти в GPSS можуть створюватися (вводиться), знищуватися (виводиться), затримуватися, розмножуватись, зливатися, накопичуватися тощо. Блок GPSS представляє собою певний самостійний елемент системи, що моделюється. Кожен блок реалізує одну або кілька операцій над транзактом, групою транзактів або параметрами транзактів, а сукупність блоків – становить моделюючу програму.

GPSS/H досить простий в оволодінні, а наявність в ньому функцій, змінних, стандартних атрибутів, графіки та статистичних блоків істотно розширює його можливості.

### 1.2.6 Система моделювання GPSS World

GPSS World – загальноцільова система імітаційного моделювання, що представляє собою потужне середовище комп'ютерного моделювання загального призначення, що розроблена для професіоналів в області моделювання [23]. Це комплексний моделюючий інструмент, який охоплює області як дискретного, так і безперервного комп'ютерного моделювання, що має найвищий рівень інтерактивності і візуального представлення інформації.

В GPSS World існує ряд анімаційних можливостей. Рівень їх реалізму змінюється в залежності від абстрактної візуалізації, що не потребує ніяких зусиль, до високо реалістичних динамічних зображень, які включають в себе складні елементи, створені користувачем.

Основне призначення GPSS – це моделювання систем масового обслуговування, хоча наявність додаткових вбудованих засобів дозволяє моделювати і деякі інші системи.

Остання версія GPSS включає в себе 53 типи блоків і 25 команд, а також більш ніж 35 системних числових атрибутів, які забезпечують поточні змінні стану, доступні в будь-якому місці моделі.

Основними об'єктами GPSS є: модель, процес моделювання, звіт і текст.

GPSS World сумісний з GPSS/PC та видає результати, що статистично не відрізняються від результатів, які видаються GPSS/PC. Цей рівень сумісності може бути досягнутий виправленням деяких відмінностей і запуском процесу моделювання.

### 1.2.7 Пакет моделювання SIMPROCESS

Пакет SIMPROCESS – це інструмент ієрархічного моделювання, який поєднує в собі співставлення процесів, моделювання дискретних подій та калькуляцію на основі активності для радикального підвищення

продуктивності моделювання і аналізу процесу [24]. Компанія-розробник SIMPROCESS – CACI Products Company (США).

Даний пакет орієнтований на організації, яким необхідно аналізувати різні сценарії розвитку підприємства і зменшувати ризики відповідно до мінливих умов діяльності.

SIMPROCESS підтримує дискретно-подієве моделювання. Анімований інтерфейс дозволяє візуалізувати динаміку моделювання «вузьких місць» в моделі.

#### 1.2.8 Програмний продукт AllFusion Process Modeler (BPWin)

AllFusion Process Modeler (раніше BPWin) – CASE-засіб для моделювання бізнес-процесів, що дозволяє створювати діаграми в нотації IDEF0, IDEF3, DFD. В процесі моделювання BPwin дозволяє переключитися з нотації IDEF0 на будь-яку гілку моделі на нотацію IDEF3 або DFD, а також створити змішану модель [25].

AllFusion Process Modeler володіє інтуїтивно-зрозумілим графічним інтерфейсом, швидко і легко освоюється, що дозволяє зосередитися на аналізі самої предметної області, не відволікаючись на вивчення інструментальних засобів. AllFusion Process Modeler допомагає швидко створювати і аналізувати моделі з метою оптимізації ділових і виробничих процесів.

AllFiision Process Modeler - підтримує відразу три стандартні нотації – EDEF0 (функціональне моделювання), DFD (моделювання потоків даних) і IDEF3 (моделювання потоків робіт). Ці три основні ракурси дозволяють комплексно описати предметну область.

До основних недоліків даного програмного продукту можна віднести: можливість розробки тільки статичних моделей, що не прив'язані до часових параметрів реальних процесів, і неоднозначність, що виникає після розробки

ієрархічної сукупності діаграм, яка може включати в себе до ста і більше діаграм.

### 1.2.9 Інструмент моделювання ProcessModel

ProcessModel – це інструмент для візуалізації, аналізу та вдосконалення бізнес-процесів різних типів, включаючи обробку транзакцій, обслуговування покупців, виробничі і складальні операції, транспортні послуги і т. д. розроблений Process-Model, Inc. (США) [26].

ProcessModel об'єднує просту технологію бізнес-діаграм з потужними можливостями імітаційного моделювання з вбудованими засобами графічної анімації.

ProcessModel дозволяє проводити аналіз «що буде - якщо» і дає можливість при мінімальних тимчасових і ресурсних витратах знайти шляхи вдосконалення процесів, навіть в тих випадках, коли інші методи неможливо використовувати.

Перевагами ProcessModel є: можливість його швидкого освоєння і легкого використання, особливо для фахівців, які володіють техніками моделювання, та інтерактивна, контекстно-залежна система допомоги, яка забезпечує всю необхідну інформацію для швидкого і легкого створення моделі предметної області.

### 1.2.10 Система імітаційного моделювання AnyLogic

Система AnyLogic – лідер серед інструментів імітаційного моделювання [27]. Дозволяє ефективно використовувати і поєднувати всі існуючі підходи до моделювання.

AnyLogic має істотну перевагу перед традиційними інструментами моделювання саме в тих проектах, розробка яких вимагає виходу за межі однієї єдиної парадигми моделювання.

Побудова моделі в AnyLogic не вимагає написання програмного коду. Якщо стандартних засобів не вистачає (або їх використання незручно), то є можливість використання мови Java. У найпростішому випадку це зводиться до опису дій, що здійснюються при переході в інший стан, спрацьовуванні таймера або отримання повідомлення. Крім того, можна додавати власний код на Java до активного об'єкту, а також використовувати сторонні бібліотеки. Це робить систему AnyLogic легко розширюваною.

Простота освоєння AnyLogic визначається знанням користувача мови Java, який використовується в комбінації з графічним середовищем розробки моделей і дає AnyLogic величезну гнучкість і виразність, що одночасно є перешкодою для розробників, які не володіють цією мовою.

У AnyLogic існує можливість створення моделей архітектури, що дозволяє інтегрувати їх з офісним і корпоративним ПЗ, включаючи електронні таблиці, БД, ERP і CRM системи та модулі, які написані на інших мовах.

Перевага AnyLogic – унікальний підхід в моделюванні, що дозволяє уявити ринок як механізм в поняттях системної динаміки, а його учасників – як агентів [28].

Широке поширення і активне використання AnyLogic, перш за все, обмежена ціною політикою компанії-розробника і рівнем використання моделювання в загальному.

#### 1.2.11 Програмний продукт моделювання Witness

Witness – програмний продукт для моделювання виробничих систем, розроблений компанією Lanner (США) [22]. Це одна з провідних систем моделювання бізнес-процесів, яка дає можливість гнучкого моделювання робочого середовища, а також моделювання наслідків різних господарських рішень і розуміння будь-яких процесів, наскільки складними вони б не були.

Witness дозволяє підтримувати зв'язок з базами даних (Oracle, SQL Server, Access тощо), має прямий доступ до всіх електронних таблиць за винятком форматів повідомлень – XML, HTML. Програмний продукт повністю інтегрований з 3D/VR Views або Post Processed VR. Він підтримує зв'язок з Microsoft Visio, забезпечує спектр прямих графічних рішень CAD і багато іншого.

Також Witness дозволяє створювати звіти (документацію) створених процесів.

#### 1.2.12 Система імітаційного моделювання Arena

Arena, програмний продукт, що розроблений компанією Systems Modeling Corporation (США) [22]. Він призначений для імітаційного моделювання процесів і систем, дозволяє створювати рухомі комп'ютерні моделі, якими можна адекватно представити багато реальних систем.

Arena забезпечена зручним об'єктно-орієнтованим інтерфейсом і має дивовижні можливості з адаптації до можливих предметних областей застосування.

Перевагою системи є те, що вона не вимагає написання програмного коду, виключно проста у використанні, але потребує значного часу для освоєння і досить глибоких знань теорії ймовірностей, математичної статистики, теорії систем масового обслуговування і мереж Петрі, що можна віднести до недоліків застосування даного продукту.

Для відображення результатів моделювання використовується анімаційна система Cinema animation. Інтерфейс Arena включає в себе різні засоби для роботи з даними, в тому числі електронні таблиці, БД, ODBC, OLE, підтримку формату DXF [29].

Наведені в розділі системи та засоби моделювання демонструють сучасний розвиток інформаційного ринку, який адаптується під потреби споживачів. У машино- та приладобудуванні, дані системи

використовуються для проектування технологічних процесів складання виробів та обробки необхідних компонентів, що дозволяє вирішити задачу технологічної підготовки приладобудівного виробництва [30].

Кожна з наведених систем моделювання (СМ) має свої переваги і недоліки.

В загальному вигляді до основних недоліків таких систем можна віднести: необхідність, ґрунтовних, початкових знань користувача для роботи з системою; суворі цінові політики розробників систем; великі часові витрати на створення моделі та проведення експериментів, а також подальшу обробку результатів.

Використання СМ має також позитивні сторони, а саме: використання ЕОМ значно скорочує час проведення експерименту в порівнянні з натуральним експериментом; рівень витрат на проведення і підготовку експерименту в СМ значно нижчий ніж вартість і підготовка реального експерименту; створена модель наділена високим рівнем гнучкості, що дозволяє оптимізувати її на основі аналізу проведених експериментів та застосовувати, за потреби, нові системи; розроблену СМ можна в будь який час реформувати, що дозволить зменшити затрати, використовуючи уже «готову» систему, замість створення нової.

Системи моделювання, що використовуються для моделювання виробничих систем в приладо- та машинобудуванні, застосовують AnyLogic, Arena, GPSS, ProcessModel. Частково це пов'язано з тим, що дані системи розроблені для російськомовних користувачів та мають дружній, зручний інтерфейс, який дозволяє на інтуїтивному рівні розуміти роботу програми та її функціональні можливості.

1.3. Стан математичного моделювання роботи виробничих систем в приладобудуванні.

Приладобудування – важлива галузь країни. Її розвиток тісно пов'язаний з рівнем життя громадян. Тому значні сили та ресурси вкладені саме у розвиток даної галузі. Аналіз ВС в приладобудуванні виконується протягом всіх етапів розвитку виробництва з метою досягнення найкращих результатів.

Комп'ютерна інтеграції виробництва в значній мірі бере початок з розвитку систем автоматизованого проектування (САПР), які сьогодні використовуються в промисловості повсюди і ефективність їх використання не викликає сумнівів [31].

Сучасне поняття САПР виходить за межі власного проектування і охоплює не тільки автоматизацію проектно-конструкторських робіт, але багато інших аспектів постановки продукції на виробництво, виготовлення виробу, його експлуатації та утилізації. Сюди входять автоматизація інженерних розрахунків, підготовка керуючих програм для верстатів з чистовим програмним керуванням, створення маршруту обробки виробу, автоматизація документообігу та керування життєвим циклом виробу. Вибіркове застосування зазначених напрямків автоматизації хоча і дозволяє підвищити ефективність виробництва, але повна віддача від вкладених засобів в автоматизацію досягається тільки при комплексному застосуванні і інтеграції всіх напрямків виробничої діяльності [32].

В наявних літературних джерелах [33–42] авторами розглянуті питання моделювання роботи ВС, проведення їх досліджень та аналіз як складних систем. В них також надано рекомендації щодо ефективного впровадження та застосування ВС в приладобудівній галузі.

Розробка нових математичних методів моделювання об'єктів та алгоритми перевірки математичних моделей на основі результатів натурного

експерименту розглядаються в дисертації [43]. Автор пропонує власні методи визначення адекватності моделі та оригінальний програмний комплекс для моделювання в середовищі MATLAB.

Смірнов Д.П. у своїй дисертації [44] досліджує підвищення ефективності аналізу і управління автоматизованими виробничими логістичними процесами за допомогою імітаційного моделювання. Автор пропонує алгоритми та методики, які реалізують моделі виробничих систем будь-якої складності, а також наводить розроблений програмний комплекс імітаційного моделювання.

Розробка моделей управління багато-продуктивної виробничої системи на основі аналізу ситуацій на ринку розглядається в дисертації [45]. Автором виконано дослідження ефективності запропонованої моделі планування виробництва продукції за допомогою розробленого програмного забезпечення (ПЗ). Тут аргументовано доцільність побудови систем керування виробництвом в класі інтелектуальних систем.

Проблематика математичного моделювання складних систем також розглядається у дисертації [46]. Автором запропонована модель для практичної реалізації, яка описує елементи системи та зв'язки між ними. Розроблено алгоритми та програми для ефективного керування системою та наведено приклади, що демонструють методику їх використання.

В роботі [47] описана імітаційна модель механообробного виробництва в системі моделювання Tecnomatix Plant Simulation. За допомогою розробленої моделі авторам вдалося домогтися підвищення продуктивності технологічного процесу в порівнянні з первісним варіантом проектування на 25%. Поліпшення отримано шляхом підбору ємності накопичувачів в системі Experiment Manager.

За допомогою цього ж програмного засобу в роботі [48] описана модель роботизованої ділянки повного виготовлення спіральних та гіпоїдних шестерень. Автори проаналізували два варіанти виробничої

дільниці: з транспортною системою і без транспортної системи, в результаті чого прийшли до висновку про реалізацію проекту першого типу.

Приклад використання Tecnomatix Plant Simulation описано в роботі [49]. Авторами розроблено імітаційну модель конвеєрної системи, що здійснює контроль та випробування коробок передач дизельних двигунів великої потужності. Особливістю моделі є комбінований склад робочих станцій, на яких операції виконуються з використанням автоматичних робочих станцій. За допомогою розробленої моделі авторами виконана серія з чотирьох експериментів, в яких визначено кількість носіїв, на яких переміщуються вироби, розмір партій виробів, оптимальна стратегія переналагодження, кількість робочих-налагоджувачів, надано рекомендації щодо різних варіантів виробничої ситуації.

Засоби моделювання наведено в статті [50], в якій автор детально розглядає існуючі на сьогоднішній день парадигми моделювання: системну динаміку, дискретно-подієве моделювання, агентне моделювання, розробку моделей на мовах програмування високого рівня. Автор зазначає, що застосування дискретно-подієвого моделювання, як однієї з основних парадигм моделювання виробничих систем, викликає певні труднощі через наявність у виробництві складних систем пріоритетів, переривань, можливості прямого взаємодії між ресурсами. Треба погодитися з автором у виборі об'єктного підходу структурування при розробці моделей складних систем. При цьому наголошується, що даний підхід поки не набув широкого поширення у розробників імітаційних моделей, навіть в тих областях де він найкращий, наприклад в агентному моделюванні.

В джерелах [51-59] також описуються питання щодо автоматизації виробничих процесів та планування підприємства.

Аналіз дисертацій та статей показав, що в одних дослідженнях глибоко опрацьовані різні аспекти, що стосуються автоматизації проектних рішень, що приймаються при проектуванні виробничих систем, в інших –

досліджуються алгоритми структурно–параметричного синтезу. Також можна відзначити, спрямованість досліджень на автоматизоване або на традиційне виробництво [31].

### Постановка задачі дисертаційних досліджень

1. Виконаний аналіз стану моделювання виробничих систем в приладо- та машинобудуванні, що представлено в множині різноманітних джерел (статті, дисертації, книги тощо), дозволив встановити основні напрямки вдосконалення та покращення роботи ВС де основним засобом їх реалізації є математичне моделювання.

2. Аналіз засобів математичного моделювання складних систем показав, що на інформаційному ринку сьогодні доступна множина програмних засобів, які застосовуються у різних галузях, зокрема в економіці. Перевагою даних систем моделювання є їх гнучкість і можливість охоплювати декілька галузей, зокрема галузь машино – та приладобудування.

3. Серед таких систем, що доцільно використовувати як засіб математичного моделювання в приладобудуванні, треба відзначити ARIS, AnyLogic, Arena, ITHINK, та GPSS. Вказані системи є лише інструментом моделювання виробничих систем, тому виникає задача створення методики їх використання для імітаційного моделювання технологічних процесів виготовлення деталей та складання приладів

На основі наведених висновків аналізу роботи виробничих систем в приладобудуванні поставлено наступні задачі дисертаційних досліджень:

- провести вибір ефективного методу опису та моделювання виробничих систем;
- надати особливості вибраного методу використання AnyLogic та особливості її застосування;
- провести опис складних систем засобами системи AnyLogic;

- розглянути моделювання динаміки роботи виробничих систем за допомогою системи AnyLogic;
- виконати математичну модель імітаційного моделювання роботи виробничої ділянки механічного цеху при виготовленні деталі;
- виконати математичну модель імітаційного моделювання роботи виробничої ділянки складального цеху при складанні хвильового редуктору;
- надати методичні рекомендації, щодо розробки методики створення імітаційних математичних моделей роботи ВС;
- надати рекомендації щодо подальшого використання створених імітаційних моделей шляхом оптимізації роботи обладнання ВС;
- розробити стартап-проект реалізації методики імітаційного моделювання складних виробничих систем;
- виконати впровадження методики імітаційного моделювання виробничих систем в приладобудуванні.

## РОЗДІЛ 2

### ОСНОВИ ПОБУДОВИ ТА МОДЕЛЮВАННЯ РОБОТИ ВИРОБНИЧИХ СИСТЕМ

#### 2.1 Особливості побудови та використання виробничих систем

На сьогодні успішність підприємства залежить від багатьох факторів: вартість і якість виготовлених товарів та послуг, що надаються, швидкість організації та модифікації виробництва, ефективний маркетинг, оптимально вибудована транспортна і виробнича логістика, правильно підібраний час виходу на ринок тощо [60]. У зв'язку з тим, що технології постійно розвиваються, виробництва розширюються, компаніям доводиться вдосконалювати свої процеси і перебудовувати внутрішню логістику, щоби не втратити конкурентоспроможність. Для цього впроваджують нові технічні розробки, методи і системи управління та змінюють внутрішню структуру підприємства

Існує множина технологій і методів моделювання, що спрямовані на дослідження і здійснення попереднього аналізу ВС. На рис.2.1 представлено основні технології та методи моделювання, які використовуються при проектуванні ВС [22].



Рис. 2.1. Технології та методи моделювання ВС.

Технології моделювання можна поділити на чотири групи: аналітичне, імітаційне, натурне, комбіноване. Кожна група має свої переваги і недоліки при моделюванні різних об'єктів. Нижче наведено порівняння технологій і методів, що спрямовані на визначення найбільш прийнятних для опису і аналізу роботи ВС.

Аналітичне моделювання є технологію опису функціонування системи за допомогою функціональних відносин (алгебраїчних, диференціальних та інтегральних рівнянь) або логічних умов [61]. Аналітичне моделювання включає в себе точні методи [62], що використовують рівняння не вище другого порядку; наближені методи [63], які засновані на тих же рівняннях, що і точні методи, але із застосуванням їх для опису багатовимірних систем з урахуванням деяких припущень; евристичні методи [64], представляють способи вирішення завдань моделювання при відсутності достатньої інформації про досліджуваний об'єкт. Перевагами аналітичного моделювання є можливість виявлення загальних теоретичних закономірностей [65] і багаторазовість використання отриманих формальних рішень. Однак досліджувати внутрішні процеси систем за допомогою даної технології можливо тільки якщо відомі математичні залежності між даними, що визначаються, і початковою інформацією про стан системи, її параметрів і змінних. Здебільшого такі залежності можна визначити тільки для досить простих систем. Крім цього, при описі складних процесів часто виникають різні математичні проблеми, що вимагають спрощення початкової моделі, що знижує точність отриманих в подальшому результатів. Аналітичні методи також не дають можливості досліджувати внутрішню структуру системи і взаємодії між її різними об'єктами, оскільки дані методи відображають лише зв'язок між вхідними і вихідними параметрами.

Технологія імітаційного моделювання дозволяє описувати системи за допомогою динамічних моделей, в яких внутрішні процеси представлено з урахуванням параметра часу і можливих впливів зовнішніх чинників, а

різного роду взаємодії регулюються встановленими набором правил, які в сукупності утворюють алгоритм функціонування даної системи. Імітаційне моделювання має декілька основних методів представлення моделі [66]:

- агентне моделювання, яке в основному використовується для опису децентралізованих систем, де модель складається з окремих об'єктів (агентів), кожен з яких має індивідуальну поведінку, при цьому має деякі зв'язки з іншими об'єктами;

- дискретно-подієве моделювання, що розглядає різні процеси як набір послідовно виступаючих подій;

- системна динаміка, що застосовується для визначення загальної стратегії на тривалі проміжки часу при дослідженні складних і великих об'єктів за рахунок визначення причинно-наслідкових зв'язків, можливих затримок і впливу впливів зовнішніх факторів.

Дана технологія дозволяє описувати широкий клас об'єктів, представляючи процеси так, як вони виглядають в досліджуваній системі. Імітаційні моделі не мають можливості формувати рішення в тому ж вигляді, що і аналітичні моделі. Вони лише служать засобом для дослідження поведінки реального об'єкта в спочатку зазначених умовах [67].

При натурному моделюванні досліджуваній об'єкт замінюється відповідним йому іншим матеріальним об'єктом, що має подібні характеристики, які необхідні для отримання подібних результатів. Натурне моделювання може бути науковим експериментом, комплексним випробуванням або виробничим експериментом [68]. Характерними особливостями наукового експерименту є використання засобів автоматизації і різних засобів обробки даних, а також наявність можливості внесення змін дослідником в процес його проведення. Комплексні випробування є різновидом експерименту, в процесі якого проводяться численні повторення випробувань для виявлення закономірностей, що пов'язані з різними характеристиками досліджуваних об'єктів. Виробничий

експеримент заснований на узагальненому досвіді, що накопичений в ході виробничого процесу і зібраний у вигляді статистичної інформації [36]. Натурне моделювання хоч і є досить наочним поданням досліджуваної системи, проте в своїй більшості воно не застосовується, оскільки не дозволяє експериментувати з встановленням різних умов тому, що зміна цих умов вимагає великих часових і фінансових витрат.

Комбіноване моделювання є об'єднанням технологій імітаційного й аналітичного моделювання. При побудові подібних моделей проводиться декомпозиція процесів системи за рахунок поділу їх на окремі підпроцеси, для яких можуть використовуватися аналітичні моделі. Решта частини системи описуються за допомогою імітаційної моделі. Дана технологія дозволяє описувати особливий клас об'єктів, які не можуть бути представлені за допомогою однієї з об'єднаних в ній технологій моделювання. До недоліків можна віднести складність побудови таких моделей, оскільки необхідно правильно розбивати досліджуваний об'єкт на частини, в залежності від використовуваних методів моделювання та установки переходів між цими методами.

У таблиці 2.1 надано порівняльну характеристику технологій моделювання, які можна застосувати для аналізу і дослідження ВС. Оскільки процес моделювання залежить від багатьох умов, в даній таблиці для наочності представлено порівняння різних технологій моделювання між собою, що надано деяким усередненим поданням про їхні можливості.

В результаті проведеного короткого аналізу існуючих і використовуваних на даний момент технологій моделювання можна зробити висновок, що найбільш придатною для дослідження і аналізу ВС є технологія імітаційного моделювання, оскільки вона дозволяє: створювати подібні за властивостями і структурі моделі різних об'єктів. При цьому зміна параметрів і умов в побудованих моделях не вимагає великих часових і фінансових витрат; а процес проведення імітаційного експерименту є

простим і наочним. Також дана технологія моделювання дозволяє враховувати наявність в досліджуваному процесі дискретних і безперервних об'єктів, випадкові зовнішні впливи, нелінійні характеристики, що дає можливість використовувати метод імітаційного моделювання при проектуванні складних ВС [65].

Таблиця 2.1. Порівняльна характеристика технологій моделювання ВС

	<b>Аналітичне моделювання</b>	<b>Імітаційне моделювання</b>	<b>Натурне моделювання</b>	<b>Комбіноване моделювання</b>
Наочність створеної моделі	Низька	Висока	Висока	Середня
Застосування для опису різних виробничих процесів	Низька	Висока	Висока	Висока
Швидкість побудови моделі	Середня	Висока	Низька	Середня
Швидкість отримання результатів моделювання	Середня	Висока	Низька	Середня
Можливість і швидкість зміни умов і параметрів моделювання	Висока	Висока	Низька	Висока
Спосіб представлення моделі	Функціональні відносини і логічні умови	Логіко-математичний опис об'єкта	Матеріальний об'єкт	Логіко-математичний опис об'єкта з включеними в нього функціональними відносинами

Як наведено раніше основними методами, що використовуються в технології імітаційного моделювання є агентне моделювання, системна динаміка та дискретно-подієве моделювання.

Основним методом, що використовується в приладобудівному виробництві в моделюванні роботи ВС є дискретно-подієвий метод.

Дискретно-подієве моделювання розглядає роботу системи, що моделюється, як набір пов'язаних між собою подій, де кожна подія настає в певний момент часу, змінюючи стан системи. Це моделювання засноване в 60-х роках минулого століття і на сьогоднішній день є найбільш розвиненим і застосовуваним для дослідження різних систем: транспортних, масового обслуговування, розподілених систем керування технологічними процесами тощо. Крім цього, множина засобів імітаційного моделювання підтримує використання методу дискретно-подієвого моделювання. Все це дає підставу вважати, що даний метод більше годиться для імітаційного моделювання ВС. Засновником дискретно-подієвого методу є Джефрі Гордон [70]. Послідовники – Томас Шрайбер [71], Петер Лоренц [72], Карпунін І.М. [73], Подкоритов Д.І. [74] та інші.

Схему алгоритму імітації функціонування досліджуваного об'єкта з використанням дискретно-подієвого методу моделювання наведено на рис. 2.2. Даний алгоритм в різних засобах імітаційного моделювання має подібний вигляд.

*Формалізація імітаційної моделі.* Результатом етапу є формальний опис досліджуваного об'єкта.

*Програмна реалізація (безпосередньо створення імітаційної моделі).* Виконання даного етапу розпочинається з вибору засобів, що необхідні для імітаційного моделювання. Далі йде алгоритмізація, складання програми та її налагодження.

*Верифікація імітаційної моделі.* Етап включає в себе різні види і способи тестування на відповідність поведінки моделі поведінки реального об'єкта.

*Робота з імітаційною моделлю, проведення експериментального дослідження.* На даному етапі в завдання дослідника входить складання плану експерименту, підбір вхідних параметрів, установка обмежень і умов. Результатом є набір вихідних параметрів.

*Аналіз отриманих результатів.* Тут вихідні параметри інтерпретуються, і за отриманими результатами досліджуються можливості реального об'єкта.

Побудова моделі ВС на наведених етапах можна представити у вигляді алгоритму, схема якого зображена на рис.2.3.

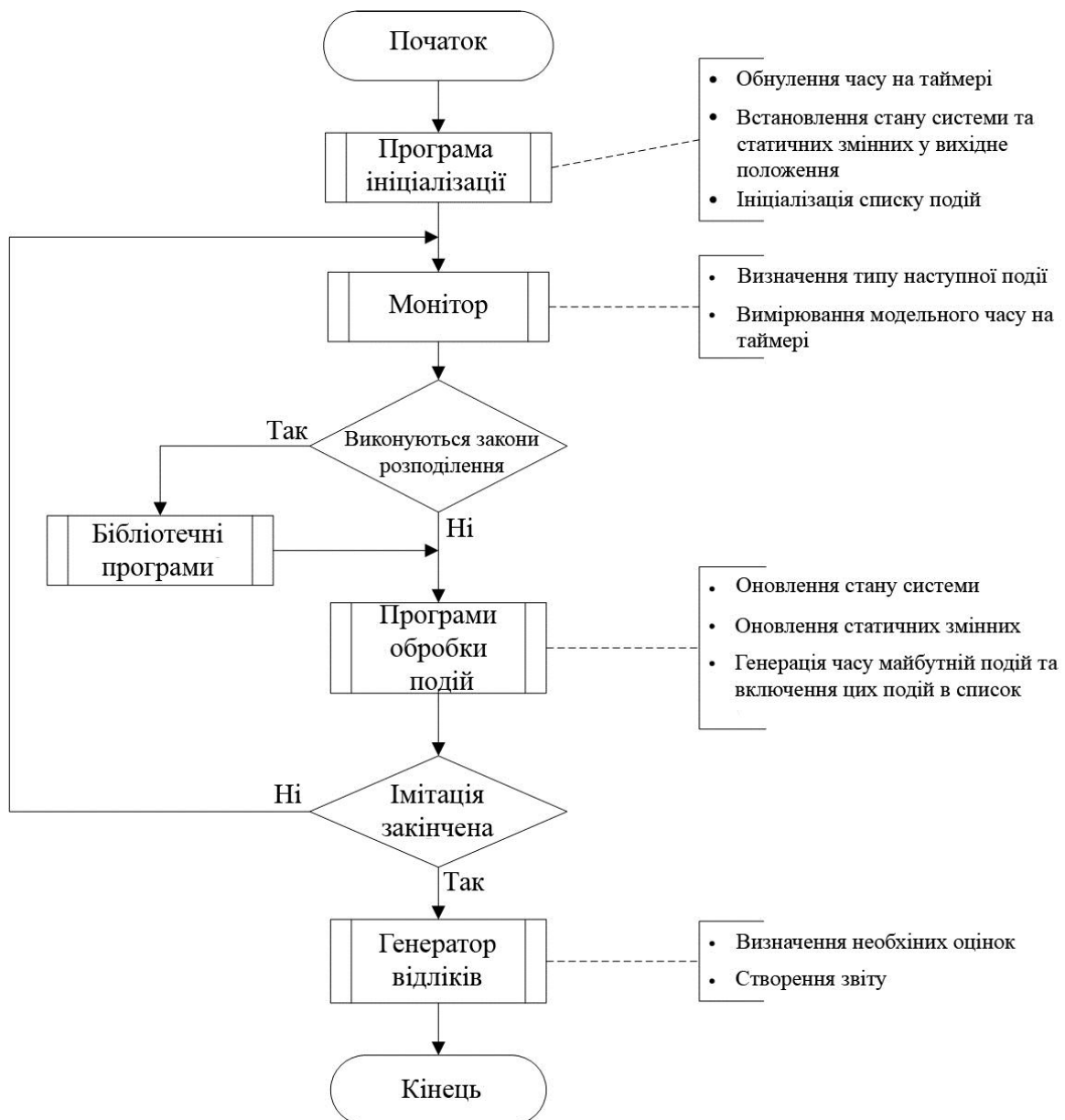


Рис. 2.2. Схема алгоритму імітації функціонування досліджуваного об'єкта з використанням дискретно-подієвого методу моделювання.

Побудова моделі, в якій основні блоки представляють собою послідовність дій, операцій, з якими пов'язані реальні і віртуальні об'єкти,

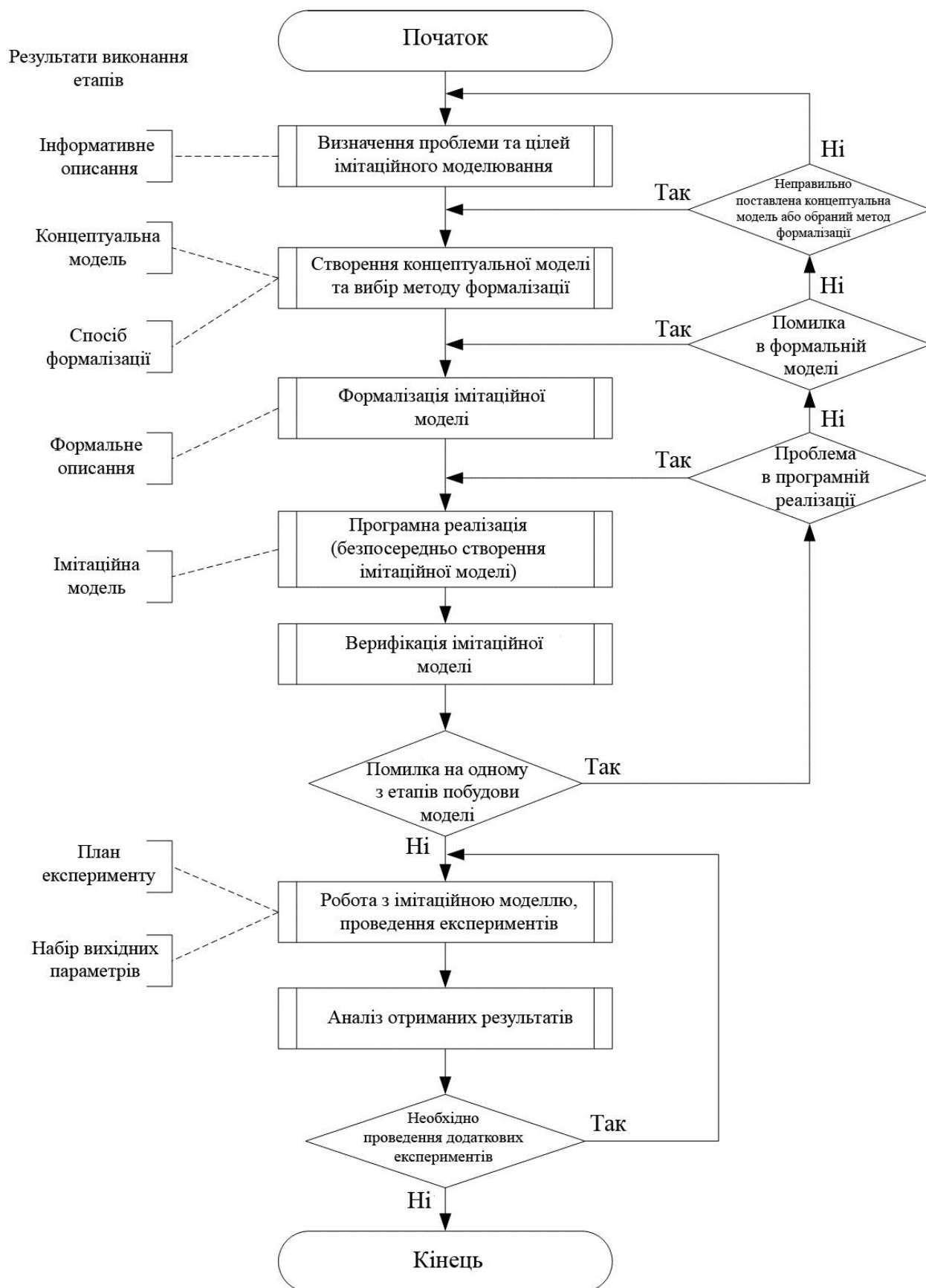


Рис. 2.3. Схема алгоритму організації імітаційного моделювання ВС.

ресурси, підрозділи, в імітаційному моделюванні визначається як функціональний підхід. Його перевагою є наочне відображення послідовності операцій, а також логіка їх взаємодії в процесі виробництва на підприємстві. Однак при цьому є недолік, що полягає в суб'єктивному підході до поділу виробничого процесу на операції.

Окрім функціонального підходу існує об'єктний опис моделей, де основним елементом є об'єкт, який може бути як певний предмет або суб'єкт, так і деяке узагальнення, наприклад, послуга, постачальник, ресурс, замовлення. Кожен об'єкт має групу атрибутів, що описують його стан, і набір операцій, які змінюють значення атрибутів і тим самим змінюють стан об'єкта.

## 2.2. Методи формалізації та моделювання роботи виробничих систем

В даному розділі описано математичний апарат, що застосовується для об'єктно-орієнтованого моделювання автоматизованих процесів і виробництв. Для моделювання виробничих систем часто застосовують різноманітні види мереж.

E-мережі («evaluation nets» – оціночні мережі) – це математичний апарат, який є різновидом розширених мереж Петрі [75], досить часто застосовується в імітаційному моделюванні. Дане розширення має деякі особливості.

1. Позиції можуть мати не більше однієї вхідної і не більше однієї вихідної дуги.

2. Позиції діляться на дві групи за кількістю міток здатних в них знаходитися: прості і черги.

3. Мітка може мати довільний набір атрибутів.

4. Переходи бувають різних типів.

5. Кожному переходу можна поставити: тимчасову затримку; функцію перетворення атрибутів міток, що проходять через даний перехід; роздільну функцію (визначає залежність, при якій здійснюється перехід).

Е-мережі можуть бути представлені за допомогою п'яти параметрів [76]:

$$E = (L, P, R, A, M),$$

де  $L$  – кінцева не порожня множина позицій;  $P$  – набір периферійних позицій  $P \subseteq L$ ;  $R$  – набір керуючих позицій  $R \subseteq L$ ;  $A$  – кінцева непорожня множина переходів;  $M$  – функція визначення початкової маркування мережі.

Периферійними позиціями вважаються ті, у яких відсутні вхідні або вихідні з'єднання з переходами. Передбачається їх використання для зовнішньої взаємодії з даною мережею. У вхідні позиції приймаються значення у вигляді міток з атрибутами, а з вихідних позицій зчитується результат роботи мережі.

Усі позиції діляться на прості позиції і позиції-черзі. При цьому в простій позиції може перебувати не більше однієї мітки в кожний момент часу, а позиція-черга може містити необмежену кількість міток.

Кожен перехід  $a \in A$  може бути описаний за допомогою трьох параметрів:

$$a_i = (S, T(a_i), Q(a_i)),$$

де  $S$  – тип переходу;  $T(a_i)$  – певний для кожного переходу мережі час затримки;  $Q(a_i)$  – визначена для кожного переходу мережі функція перетворення атрибутів.

Розвитком даного виду розширених мереж Петрі було створення модифікованих Е-мереж [77], основною відмінністю яких стало визначення нового базису у вигляді набору елементарних мереж.

Елементарна мережа в модифікованих Е-мережах є об'єднанням, яке містить один перехід, пов'язані з ним позиції і умови його роботи. Вони не

містять керуючих позицій. Замість них переходам задається роздільна функція, яка визначає залежність, при якій здійснюється перехід.

Елементарна мережу з переходом  $t$  задається наступним набором параметрів [78]:

$$E(t) = (S, P_{in}, P_{out}, f_{in}, f_{out}, d, c),$$

де  $S$  – загальна умова, що є обов'язковим для здійснення переходу  $t$ , але при цьому недостатнє;  $P_{in}$  і  $P_{out}$  – групи вхідних і вихідних позицій, пов'язані з переходом  $t$ ;  $d$  – функція визначення часу переходу (затримки);  $c$  – функція зміни атрибутів міток.

На рис.2.4 представлені типи елементарних мереж, які містять модифіковані E-мережі.

T-мережа служить для збільшення або зменшення кількості міток. При відсутності вхідних або вихідних позицій мережа являє собою або генератор об'єктів, або їх утилізатор. На рис.2.4, а наведено приклад даного типу мережі. Формально даний приклад можна описати таким чином:

$$P = C(p_1, p_2, p_3) \wedge \sim(C(p_4) \vee C(p_5));$$

$$M'(p_i) = M(p_i) - 1; M'(p_j) = M(p_j) + 1; \quad i = 1, 2, 3; j = 1, 2,$$

де  $P$  – умова спрацьовування переходу;  $C$  – предикат, що приймає значення одиниці, якщо всі аргументи (позиції) мають мітку;  $M(p)$  – маркування позиції;  $M'(p)$  – результуюча маркування позиції  $p$ .

X-мережа здійснює перетворення вхідних міток в одну вихідну мітку з вибором позиції. Дозволяє описувати процес розподілу і сортування об'єктів системи.

Приклад даної мережі представлений на рис.2.4, б. Його формальний опис:

$$P = C(p_1, p_2, p_3) \wedge \sim C(p_4, p_5);$$

$$M'(p_i) = M(p_i) - 1; i = 1, 2, 3;$$

$$p = F(p_4, p_5); M'(p) = M(p) + 1,$$

де  $F$  – функція вибору порожньої позиції.

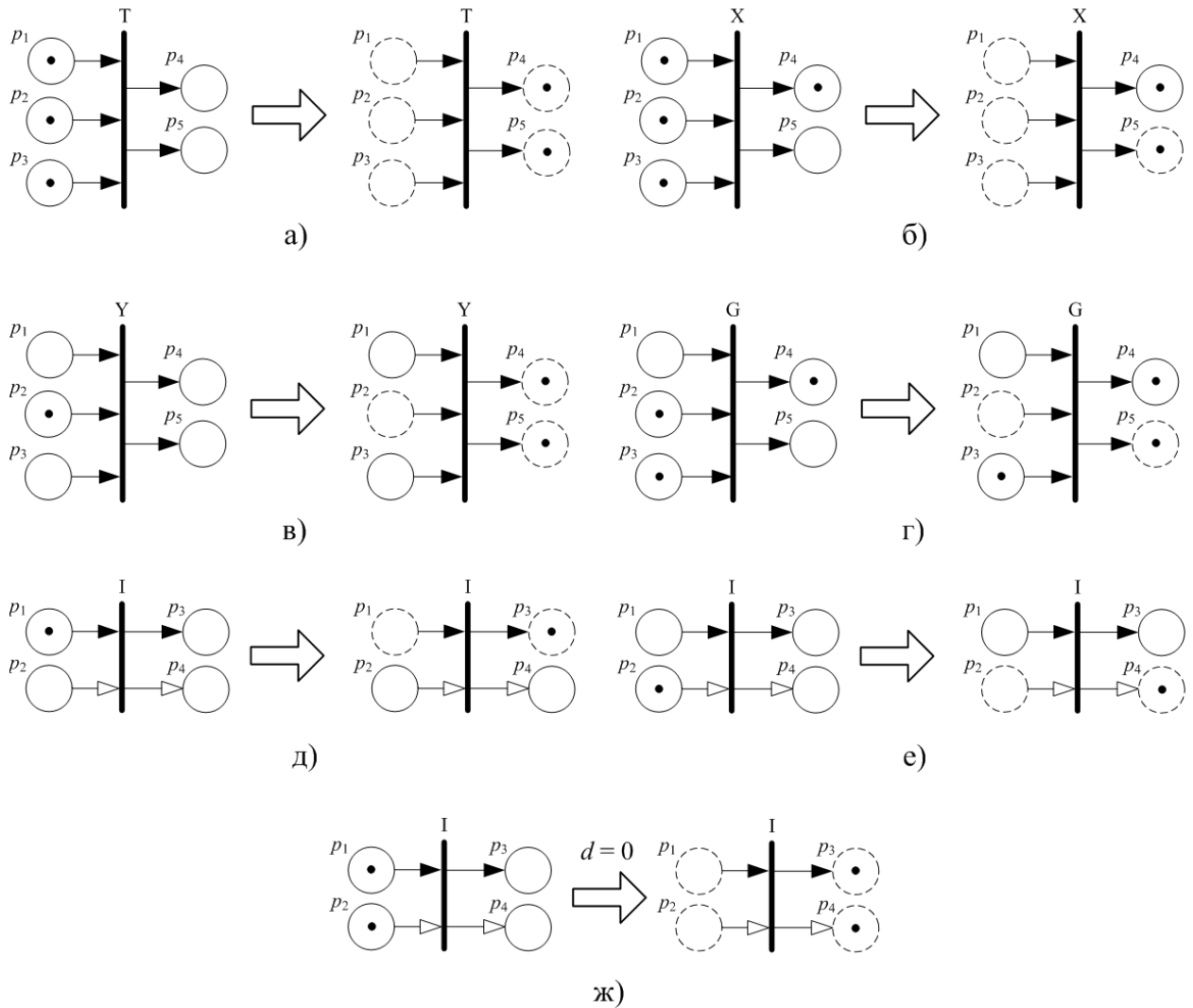


Рис. 2.4. Варіанти функціонування базового набору елементарних мереж.

Y-мережа виконує зворотню дію елементарної X-мережі. Вибирається мітка з однією вхідною позицією і додаються мітки в усі вихідні позиції. Може застосовуватися для моделювання декількох джерел надходження об'єктів (постачальники, конвеєри та ін). Приклад Y-мережі представлений на рис.2.4, в. Формальний його опис виглядає наступним чином:

$$P = (C(p_1) \vee C(p_2) \vee C(p_3)) \wedge \sim (C(p_4) \vee C(p_5));$$

$$M'(p_i) = M(p_i) - 1; i = 1, 2, 3;$$

$$p = H(p_1, p_2, p_3); M'(p) = M(p) \square 1;$$

$$M'(p_j) = M(p_j) + 1; j = 1, 2,$$

де,  $H$  – функція вибору позиції з міткою.

G-мережа об'єднує в собі X-мережі та Y-мережі. Вибирається мітка з однієї вхідної позиції і додається в одну вихідну позицію. Мережа підходить для опису маршрутизації. Приклад її роботи представлений на рис.2.1, г. Вхідні умови збігаються з умовами Y-мережі, а вихідні умови збігаються з вихідними умовами X-мережі.

I-мережа має спеціальну позицію, що дозволяє перервати поточну дію (перехід). За допомогою неї можна обмежувати операції по часу, моделювати реакцію на нестандартні ситуації тощо.

При появі мітки в позиції переривання дії, що відбуваються в переході, миттєво припиняються і мітки йдуть у вихідні позиції. Можливі три ситуації.

1. Мітка присутня тільки в основній (безперервній) вхідній позиції (рис.2.4, д):

$$P = C(p_1) \wedge \sim C(p_2) \wedge \sim C(p_3);$$

$$M'(p_1) = M(p_1) - 1; M'(p_3) = M(p_3) + 1.$$

2. Мітка міститься лише в позиції, що переривається, (рис.2.4, е):

$$P = \sim C(p_1) \wedge C(p_2) \wedge \sim C(p_4);$$

$$M'(p_2) = M(p_2) - 1; M'(p_4) = M(p_4) + 1;$$

$$d = 0.$$

3. Мітки знаходяться в обох позиціях (рис.2.4, ж):

$$P = C(p_1) \wedge C(p_2) \wedge \sim C(p_3) \wedge \sim C(p_4);$$

$$M'(p_1) = M(p_1) - 1; M'(p_3) = M(p_3) + 1;$$

$$M'(p_2) = M(p_2) - 1; M'(p_4) = M(p_4) + 1;$$

$$d = 0.$$

Доведено [79], що для опису будь-якого алгоритму, а значить і ВС, досить використовувати описаний вище набір елементарних мереж. Крім цього, E-мережі використовуються для моделювання різних систем [80–87], що доводить можливість їх практичного застосування.

### 2.3. Вибір ефективного методу опису та моделювання виробничих систем

У будь-якій методиці моделювання процесів підприємства одним з основних елементів є зв'язки, які необхідні для позначення взаємодії між об'єктами або операціями. До таких взаємодій відносяться: використання одного об'єкта іншим, зв'язок за допомогою потоку даних, послідовність дій виконуваних в часі тощо.

З проведеного аналізу, розглянутому в попередньому розділі роботи, зрозуміло, що найбільш перспективним і досить ефективним на даний момент напрямом для дослідження і функціонального аналізу складних систем є використання дискретно-подієвого підходу на основі імітаційного моделювання. Використання цього підходу в повній мірі дозволить побудувати та проводити дослідження імітаційної моделі виробничих систем.

Існує безліч різних засобів моделювання, серед програмних засобів, що підтримують даний тип моделювання і використовуються для опису ВС, виділяють: AnyLogic [88], Arena [88], ARIS [89].

Порівняльний аналіз виділених систем проводився по п'яти параметрам: можливості використання графічної мови опису моделей, мови моделювання, застосування дискретно-подієвого проектування, обліку часу виконання процесу, використання математичного апарату в якості засобу опису моделі, ієрархічним поданням моделі процесу, використання бази даних для зберігання вхідних значень.

Порівняння характеристик засобів імітаційного моделювання наведено в таблиці 2.1.

Сучасні засоби імітаційного моделювання Arena і ARIS володіють розвиненим апаратом опису та подання виробничих процесів. Однак в даних системах в якості основного засобу опису моделі не використовуються відомі і опрацьовані математичні апарати на зразок розширених мереж Петрі, які

мають високий рівень формалізації дискретно-подієвих систем, дозволяючи проводити аналітичне дослідження властивостей моделі та гарантувати можливість побудови будь-якого алгоритму дій. Продукт AnyLogic, навпаки, дозволяє описувати ВС за допомогою ієрархічних тимчасових кольорових мереж Петрі.

Таблиця 1.2. Порівняльна характеристика засобів імітаційного моделювання

<b>Характеристики</b>	<b>AnyLogic</b>	<b>Arena</b>	<b>ARIS</b>
Використання графічної мови опису моделей	Так	Так	Так
Мова моделювання	Java	Siman	SAX Basic
Дискретно-подієве проектування	Так	Так	Ні
Облік часу виконання процесу	Так	Так	Так
Використання певного математичного апарату як засіб опису моделі	Ні	Ні	Ні
Ієрархічне представлення моделі процесу	Так	Так	Так
Використання бази даних для зберігання вхідних значень	Так	Так	Так

Середовище імітаційного моделювання (ІМ) AnyLogic має також ряд переваг: наявність всіх парадигм ІМ (висока гнучкість вибору підходу), можливість вибору між парадигмами або застосування комплексного підходу. Цей продукт має всі властивості, які необхідні для розробки імітаційних моделей ВС.

Дане середовище надає можливість експортувати розроблену модель в додаток, забезпечуючи цим доступність застосування результатів дослідження. AnyLogic розроблена на універсальній мові програмування Java, що дозволяє середовищу не залежати від виду операційної системи, на якій проводяться дослідження.

AnyLogic є одним з найбільш розповсюджених середовищ ІМ з російськомовним інтерфейсом та застосованих в приладобудівній галузі, що найкраще підходить для виконання поставлених задач в магістерському дослідженні.

#### Висновки до розділу

Аналіз сучасних технологій моделювання показав, що для дослідження і аналізу ВС найбільш придатною є технологія імітаційного моделювання, оскільки вона дозволяє створювати подібні за властивостями і структурою моделі виробничих процесів, зміна параметрів і умов в побудованих моделях не вимагає великих часових і фінансових витрат, процес проведення імітаційного експерименту є простим і наочним.

Аналіз сучасних методів моделювання ВС дозволив виділити дискретно-подієвий метод як найбільш придатний для опису об'єктів предметної області, що на даний момент є найбільш перспективним і ефективним напрямком.

Аналіз сучасних засобів імітаційного моделювання показав, що такі засоби опису виробничих процесів як Arena і ARIS в якості основного засобу

опису моделі не використовують відомі математичні апарати, що не дає їм можливості проводити аналітичне дослідження властивостей моделі та гарантувати можливість побудови різних видів алгоритмів дій. Система AnyLogic, навпаки, дозволяє описувати ВС за допомогою ієрархічних тимчасових кольорових мереж Петрі, що являється найбільш зручним інструментом для виконання поставлених задач магістерської роботи.

За допомогою системи AnyLogic є можливість побудувати математичні моделі роботи діляниць виготовлення деталей та ліній складання виробів шляхом імітації всіх елементів технологічних процесів ще до початку запуску їх в реальне виробництво.

Можливості AnyLogic – це відкрита архітектура, аналіз та анімація процесів, оптимізація, а також величезна база об'єктів.

## РОЗДІЛ 3

### ВИКОРИСТАННЯ СИСТЕМИ ANY LOGIC ДЛЯ ОПИСУ РОБОТИ ТА МОДЕЛЮВАННЯ РОБОТИ ВИРОБНИЧИХ СИСТЕМ

#### 3.1. Система Any Logic та особливості її застосування

Спочатку 1990-х рр. в комп'ютерній науці спостерігався великий інтерес до побудови математично трактованого опису взаємодії паралельних процесів. Невелика група вчених займалася розробкою пакету програм для аналізу коректності паралельних і розподілених систем [90]. Такий програмний продукт був створений і названий COVERS. У цьому пакеті Аналізуюча система процесів задавалася графічно, за допомогою опису її структури і поведінки окремих паралельних компонентів, які могли взаємодіяти з оточенням та з іншими процесами і середовищем.

У 1998 р успіх цього дослідження надихнув вчених організувати комерційну компанію з місією створення нового програмного забезпечення для імітаційного моделювання. Акцент при розробці такого програмного продукту ставився на прикладні методи: моделювання стохастичних систем; оптимізація і візуалізація моделі.

Нове програмне забезпечення, в 2000 р. засновано на останніх перевагах інформаційних технологій: об'єктно-орієнтований підхід, елементи стандарту UML (Unified Modeling Language), мови програмування Java тощо.

Продукт отримав назву AnyLogic, тому що він підтримував три відомих напрямки імітаційного моделювання [90] (рис.3.1).

В цій системі реалізовано будь-які комбінації наведених напрямків в межах однієї моделі. Першій версії системі надано індекс 4 - Anylogic 4.0, тому що нумерація продовжила історію версій попередньої розробки - COVERS 3.0.



Рис 3.1. Напрямки імітаційного моделювання, що реалізовані в системі AnyLogic.

Величезний крок вперед зроблено в 2003 році, коли був випущений AnyLogic 5, що орієнтований на бізнес-моделювання. За допомогою AnyLogic стало можливим розробляти моделі в багатьох областях.

Версія AnyLogic 6 написана на мові програмування Java в популярному середовищі розробки Eclipse і є крос-платформним програмним забезпеченням, що працює як під керуванням операційної системи Windows, так і під Mac, OS і Linux.

AnyLogic включає в себе графічну мову моделювання, а також дозволяє користувачеві розширювати створені моделі за допомогою мови Java. Інтеграція компілятора Java в AnyLogic надає більш широкі можливості при створенні моделей, а також створення Java аплетів, які можуть бути відкриті будь-яким браузером. Ці аплети дозволяють легко розміщувати моделі AnyLogic на веб-сайтах. Як доповнення до Java аплетам, AnyLogic Professional підтримує створення Java додатків. Тоді користувач може запустити імітаційну модель без інсталяції AnyLogic [90].

Система AnyLogic, що розроблена компанією XJTechnologies (Росія), є середовищем комп'ютерного моделювання загального призначення. Вона є

комплексним інструментом, що охоплює основні напрями моделювання: дискретно-подієве, системну динаміку та агентне моделювання.

Використання AnyLogic дає можливість оцінити ефект конструкторських рішень в складних системах реального світу.

Професійний інструмент імітаційного моделювання AnyLogic нового покоління, що розроблений на основі сучасних концепцій в області інформаційних технологій і результатів досліджень в теорії гібридних систем і об'єктно-орієнтованого моделювання. Побудована на їх основі інструментальна система AnyLogic не обмежує користувача однією єдиною парадигмою моделювання, що є характерним для існуючих на ринку інструментів моделювання. В AnyLogic розробник може гнучко використовувати різні рівні абстрагування, різні стилі і концепції та поєднувати їх при створенні однієї і тієї ж моделі.

Програмний продукт AnyLogic заснований на об'єктно-орієнтованій концепції моделювання. Об'єктно-орієнтований підхід до представлення складних систем є найкращим на сьогодні методом управління складністю інформації, ця концепція дозволяє простим і природним чином організувати і представити структуру складної системи. Таким чином, ідеї і методи, що спрямовані на управління складністю, вироблені в останні десятиліття в області створення програмних систем, дозволяють розробникам моделей в середовищі AnyLogic організувати мислення, структурувати розробку і, в кінцевому рахунку, спростити і прискорити створення моделей .

Іншою базовою концепцією AnyLogic є представлення моделі як набору взаємодіючих паралельно функціонуючих активів. Такий підхід до моделювання інтуїтивно зрозумілий в багатьох додатках, оскільки системи реального життя складаються з сукупності активів, котрі взаємодіють із іншими об'єктами. Активний об'єкт AnyLogic – це об'єкт зі своїм власним функціонуванням, що взаємодіє з оточенням. Він може включати в себе будь-

яку кількість примірників інших активних об'єктів, які можуть динамічно породжуватися і зникати відповідно до законів функціонування системи.

Графічне середовище моделювання AnyLogic підтримує проектування, розробку, документування моделі, виконання комп'ютерних експериментів з моделлю, включаючи різні види аналізу – від аналізу чутливості до оптимізації параметрів моделі щодо деякого критерію.

При розробці моделі на AnyLogic можна використовувати концепції і засоби з деяких «класичних» областей моделювання, наприклад, в агентній моделі використовувати методи системної динаміки для подання змін стану середовища або в безперервній моделі динамічної системи врахувати дискретні події. У AnyLogic легко будуються подібні моделі з необхідним рівнем адекватності, що дозволяють відповісти на багато питань, що цікавлять дослідника. Багаті можливості анімації і візуального представлення результатів в процесі роботи моделі дозволяють зрозуміти суть процесів, що відбуваються в моделюється системі, спростити налагодження моделі [90].

Галузі та сфери використання програмного продукту AnyLogic: керування активами; бізнес-процеси; динаміка екосистем; охорона здоров'я; логістика; виробництво; ринок та конкуренція; соціальна динаміка; ланцюги поставок; телекомунікації; транспорт; охорона здоров'я і фармацевтика; соціальні та екологічні системи.

Графічне середовище AnyLogic 5 побудовано за тим же принципом, що і в Rockwell Arena. Моделюючі конструкції розташовуються в палітрах (аналог шаблонів в Arena). Для створення моделі, як і в Arena, модельовані конструкції перетягують в область моделі, а потім з'єднують.

### 3.2. Опис складних систем засобами системи Any Logic

AnyLogic має дві фази імітаційного моделювання – розробка моделі та її аналіз [91]. Розробка моделі здійснюється в середовищі редактора

AnyLogic, а аналіз моделі реалізується в середовищі виконання. У кожній фазі існують свої засоби управління. Перехід з однієї фази в іншу проводиться досить просто. Можна багаторазово використовувати перехід між фазами редагування і виконання моделі при її розробці.

Клас в програмуванні є потужним засобом, що дозволяє структурувати складну систему. Клас визначає шаблон, відповідно до якого будуються окремі екземпляри класу. Ці екземпляри можуть бути визначені як об'єкти інших активних об'єктів. У AnyLogic основним структурним блоком при створенні моделей є класи активних об'єктів. Використання активних об'єктів є природним засобом структуризації моделі складних систем: світ складається з безлічі паралельно функціонуючих і взаємодіючих між собою сутностей. Різні типи цих сутностей і представляють різні активні об'єкти.

Для того, щоб розробити модель в AnyLogic, потрібно створити класи активних об'єктів (або використовувати вже існуючі об'єкти бібліотек AnyLogic). Визначення активного об'єкта задає шаблон. Окремі об'єкти, які побудовані відповідно до цього шаблону (екземпляри активного об'єкта), можуть використовуватися як елементи інших активних об'єктів. У будь-який клас можуть бути включені кілька примірників інших класів, які можуть відрізнятися своїми параметрами. Завжди один клас в моделі є кореневим. Для нього в моделі AnyLogic породжується один екземпляр з визначеним ім'ям root. Цей екземпляр запускається виконавчою системою AnyLogic для реалізації. Ім'я класу кореневого активного об'єкта можна змінювати у вікні його властивостей [91].

Кожен активний об'єкт має структуру, а також поведінку, що обумовлена сукупністю змінних, параметрів тощо. Кожен екземпляр активного об'єкта в працюючій моделі має свою власну поведінку. Він може мати свої значення параметрів, що функціонують незалежно від інших об'єктів і взаємодіють з ними із зовнішнім середовищем.

AnyLogic використовує об'єктно-орієнтований підхід для представлення складних систем. Цей підхід дозволяє простим і природним чином організувати і представити структуру складної системи за допомогою ієрархії абстракцій. Наприклад, на деякому рівні абстракції автомобіль можна вважати єдиним, цілим об'єктом. Але більш детально його можна представити як сукупність взаємодіючих підсистем: двигуна, рульового управління, гальмівної системи тощо. Кожна з цих підсистем може бути представлена своєю структурою взаємодіючих підсистем.

Саме таку ієрархію абстракцій дозволяє створити AnyLogic при розробці моделей. Всю модель можна розглядати як єдиний об'єкт root. При детальному розгляді видно, що цей об'єкт може містити, наприклад, два примірника класу MyClass з іменами myClass і myClass1. Сам клас MyClass може мати своє складну будову, яка буде прихована в класі Root. Така ієрархія структури може бути довільної глибини.

При побудові моделі в середовищі AnyLogic не передбачена можливість використання ніяких інших засобів, крім засобів візуальної розробки, завдання чисельних значень параметрів, аналітичних записів співвідношень змінних і аналітичних записів умов настання подій. Основною парадигмою, що прийнята в AnyLogic при розробці моделей, є візуальне проектування – побудова за допомогою графічних об'єктів і піктограм ієрархій структури і поведінки активних об'єктів.

AnyLogic є надбудовою над мовою Java – одним з найпотужніших і в той же час простих сучасних об'єктно-орієнтованих мов програмування. Всі об'єкти, що визначені користувачем при розробці моделі на AnyLogic за допомогою його графічного редактора, транслюються в конструкції мови Java, а потім – відбувається компіляція всієї зібраної програми на Java, яка представляє модель у виконуваному коді.

Незважаючи на те, що при побудові моделі на AnyLogic розробник використовує конструкції мови Java, в дійсності він ніколи не розробляє

повні програми, він не програмує, а лише вставляє фрагменти коду в спеціально передбачені для цього поля вікна «Код» і вікон властивостей об'єктів моделі. Ці фрагменти висловлюють логіку конкретних кроків або дій в моделі. Як і всі інші включені в модель програмні фрагменти, ці вирази повинні бути синтаксично правильними конструкціями Java, тому розробник моделей AnyLogic повинен мати певне уявлення про цю мову.

Основним засобом специфікації поведінки об'єктів в AnyLogic є змінні, таймери і стейтчарти. Тут змінні відображають характеристики об'єкта, що змінюються. Таймери можна встановлювати на певний інтервал часу і після закінчення цього інтервалу виконувати задану дію. Стейтчарти дозволяють візуально представити поведінку об'єкта в часі під впливом подій або умов, вони складаються з графічного зображення станів і переходів між ними. Будь-яка складна логіка поведінки об'єктів моделі в AnyLogic може бути виражена за допомогою комбінації стейтчартів, диференціальних і алгебраїчних рівнянь, змінних, таймерів і програмного коду на Java. Алгебраїчні і диференціальні рівняння, як і логічні вираження, записуються в AnyLogic аналітично [91].

Інтерпретація будь-якого числа процесів, що протікають паралельно, в моделі AnyLogic прихована від користувача. Ніяких календарів подій розробник моделі на AnyLogic не веде, відстеження подій у всіх процесах, що визначені у моделі, виконується системою автоматично.

Поняття модельного часу є базовим в системах імітаційного моделювання. Модельний час – це умовний логічний час, в одиницях якого визначено поведінку всіх об'єктів моделі.

У моделях AnyLogic модельний час може змінюватися або безперервно, якщо поведінка об'єктів описується диференціальними рівняннями, або дискретно, перемикаючись від моменту настання однієї події до моменту настання наступної події, якщо в моделі присутні тільки дискретні події. Моменти настання всіх запланованих подій в дискретній

моделі виконавча система зберігає в так званому календарі подій, вибираючи звідти найбільш ранні події для виконання пов'язаних з ними дій. Значення поточного часу в моделях AnyLogic може бути отримано шляхом звернення до функції *getTime ()*.

Одиницю модельного часу розробник моделі може інтерпретувати як будь-який відрізок часу: в секундах, хвилинах, годинах або роках. Важливо щоб всі процеси, що залежать від часу, були виражені в одних і тих же одиницях. При моделюванні фізичних процесів всі параметри і рівняння повинні бути виражені в одній і тій же системі фізичних величин. Наприклад, всі фізичні величини виражаються в системі СІ, в якій одиниця часу – секунда, а одиниця довжини – метр.

Інтерпретація моделі виконується на комп'ютері. Фізичний час, що витрачається процесором на імітацію дій, які повинні виконуватися в моделі протягом однієї одиниці модельного часу, залежить від багатьох чинників. Тому одиниці фізичного і одиниці модельного часу не збігаються.

У AnyLogic прийняті два режими виконання моделей: режим віртуального часу і режим реального часу. У режимі віртуального часу процесор працює з максимальною швидкістю без прив'язки до фізичного часу. Цей режим використовується для факторного аналізу моделі, набору статистики, оптимізації параметрів моделі тощо. Оскільки анімація та інші вікна спостереження за поведінкою моделі зазвичай істотно знижують продуктивність комп'ютера, для підвищення швидкості виконання моделі ці вікна потрібно закрити.

У режимі реального часу користувач задає зв'язок модельного часу з фізичним часом, тобто встановлюється обмеження на швидкість роботи процесору при виконанні моделі. В цьому режимі задається кількість одиниць модельного часу, які повинні виконуватися процесором в одну секунду. Зазвичай даний режим включається для того, щоб візуально уявити

функціонування системи в реальному темпі настання подій, проникнути в суть процесів, що відбуваються в моделі.

Зручні засоби розробки анімаційного представлення моделі в AnyLogic дозволяють уявити функціонування модельованої системи в живій формі динамічної анімації, що дозволяє «побачити» поведінку складної системи. Засоби анімації дозволяють користувачеві легко створити віртуальний світ, що керований динамічними параметрами моделі за законами, які визначені користувачем за допомогою рівнянь і логіки модельованих об'єктів. Візуальне уявлення поведінки системи допомагає користувачеві проникнути в суть процесів, що відбуваються в системі.

Багато систем моделювання дозволяють змінювати параметри моделі тільки до запуску моделі на виконання. Система AnyLogic надає користувачеві можливість змінювати параметри моделі в ході її функціонування. Тому вікно анімації можна назвати «стендом» для проведення комп'ютерного експерименту з моделлю [91].

Таким чином, будь яку складну систему, до яких відносяться виробничі системи в приладобудуванні, можна описати засобами Any Logic.

### 3.3. Моделювання динаміки роботи виробничих систем за допомогою системи Any Logic

Розглянемо динаміку роботи складної системи на прикладі моделі імітаційного моделювання процесу виготовлення деталі в цеху підприємства [92].

Виготовлення в цеху деталі починається через випадковий час  $T_n$ . Виконанню операцій передують підготовка. Тривалість підготовки залежить від якості заготовки, з якої буде виготовлена деталь. Всього різних видів заготовок  $n_1$ . Час підготовки операції відповідає експоненціальному закону.

Частота появи різних заготовок і середні значення часу їх підготовки задані в таблиці 3.1 дискретного розподілу:

Таблиця 3.1. Частота появи різних заготовок і середні значення часу їх підготовки.

<b>Частота</b>	0,05	0,13	0,16	0,22	0,29	0,15
<b>Середній час</b>	10	14	21	22	28	25

Для виготовлення деталі послідовно виконуються  $n$  операцій із середніми часом  $T_1, T_2, \dots, T_n$  відповідно. Після кожної операції протягом часу  $T_{k1}, T_{k2}, \dots, T_{kn}$ , проходить контроль. Час виконання операцій і контролю – випадковий. Контроль не проходять  $q_1, q_2, \dots, q_n$  % деталей відповідно.

Забраковані деталі надходять на пункт контролю готової продукції та проходять на ньому перевірку протягом часу, що розподілений за експоненціальним законом із середнім значенням  $T_k$ .

В результаті цієї операції із загальної кількості деталей, які не пройшли контроль,  $q_{n+1}$  % йдуть в брак, а ті що залишилися  $(1 - q_{n+1})$  % деталей підлягають повторному виконанню операцій, після яких вони не пройшли контроль. Якщо деталь вдруге не проходить контроль, вона остаточно бракується.

Завданням дослідження процесу виготовлення деталі в цеху підприємства є розробка імітаційної моделі для визначення оцінки математичного очікування кількості деталей, що виготовлені протягом 8 годин.

Модель повинна також дозволяти визначати відносну кількість готових і забракованих деталей та середній час виготовлення однієї деталі.

Результати моделювання необхідно отримати з точністю  $\varepsilon = 0,01$  і довірчою ймовірністю  $\alpha = 0,99$ .

AnyLogi-модель процесу виготовлення в цеху деталей має включати такі сегменти: початкові дані; підготовка заготовки; операція 1; операція 2; операція 3; пункт контролю готової продукції; склад готових деталей; склад бракованих деталей. результати моделювання.

Для введення вихідних даних використовуємо елементи меню програми «**Параметр**».

Створення імітаційної моделі виконується в такій послідовності.

1. Послідовно виконати команди **Файл / Создать/ Модель** на панелі інструмент.

2. У поле «**Имя модели**» діалогового вікна **Новая модель** ввести «*Изготовление\_в\_цехе\_деталей*». Обрати каталог, в якому будуть збережені файли створюваної моделі.

3. Натиснути кнопку **Готово**. Створити область перегляду для розміщення елементів сегменту **Цех**.

4. З палітри **Презентация** перетягнути елемент **Область просмотра**.

5. На панелі **Свойства** в поле **Имя:** ввести «*цех*».

6. Встановити **Выравнивать по:** «*Верхн. левому углу*».

7. Обрати режим масштабування зі списку **Масштабирование:** «*Подогнать под окно*».

8. На сторінці **Местоположение и размер** ввести в поля:

**X:** 0, **Y:** 0, **Ширина:** 780, **Высота:** 530.

9. Перетягнути елемент **Скруглённый прямоугольник**. На сторінці **Местоположение и размер** встановити:

**X:** 280, **Y:** 300, **Ширина:** 250, в поле **Высота:** 220.

10. Перетягнути елемент **text** і на сторінці **Основные** панелі **Свойства** в поле **Текст:** ввести «*Исходные данные*».

11. У **Палитре** виділити **Основная**. Перетягнути елементи **Параметр** на елемент з ім'ям «*Исходные данные*». Додати їх так, як показано на рис. 3.2.

12. Два елементи **Параметр**, відрізняються від інших. Вони використовуються для введення даних згідно таблиці 3.1 як одновимірних масивів з іменами «*верVarЗаг*» (ймовірності варіантів заготовок) і «*срВрПодгЗаг*» (середній час підготовки варіанту заготовки).

Надалі необхідно створити розмірності масивів. В даному випадку всі розмірності є однаковими. Елементів в одному рядку таблиці 3.1 – шість. Припустимо, що число видів заготовок може збільшитися до 10. Значить розмірність одного масиву складається з 10 елементів. Для задання розмірностей масивів необхідно виконати таку послідовність дій.

1. Натиснути правою кнопкою мишки в панелі **Проекты** та в контекстному меню виберіть **Создать/Размерность**.

2. У вікні **Размерность** в поле **Имя** ввести «*КолVarЗаг*».

3. Встановити **Тип размерности: Диапазон**.

4. В поле, що відкрилося **Диапазон**: ввести «*1-10*».

5. Натиснути **Готово**.

Створення інформаційних масивів виконують, розпочинаючи з масиву «*верVarЗаг*» для ймовірностей появи видів заготовок.

1. З Палітри **Основные** перетягнути елемент **Параметр**.

2. На сторінці **Основные** панелі **Свойства** в поле **Имя**: ввести «*верVarЗаг*». **Тип**: «*double*».

3. Встановити прапорець **Массив**. Відкриється вікно **Размерности**. Натиснути по розташованій праворуч від вікна і підсвіченій зеленій кнопці.

4. Відкриється вікно **Edit dimensions**. У вікні Можливі розмірності: виділити «*КолVarЗаг*».

5. Натиснути по кнопці. **Размерности** *КолVarЗаг* з'явиться у віконці вибрані розмірності.

6. Закрити вікно це приведе до повернення на панель **Свойства**. У віконці **Размерности** розташована розмірність *КолVarЗаг*.

7. Натиснемо **Редактировать значения массива**. Відкриється однойменне діалогове вікно.

8. У лівій частині вікна стрілками показано розміщення елементів масиву. Залишимо горизонтальне. Елементи масиву мають різні значення. Тому не варто використовувати **[ВСЕ]**.

9. У правій частині вікна ввести значення елементів масиву:

«0.05, 0.18, 0.34, 0.56, 0.85, 1, 0, 0, 0, 0»

10. Натиснути **ОК**. Це приведе до повернення на панель **Свойства**. У полі **Значение по умолчанию**: з'являться введені значення шести елементів масиву. Решта чотири елементи дорівнюють нулю.

11. Аналогічним чином створити другий масив з ім'ям «*срВрПодгЗаг*» для середнього часу підготовки заготовки.

У подієву (функціональну) частину моделі необхідно включити вказані раніше сегменти. Оскільки побудова моделі це ітераційний процес, то розміщення сегментів і об'єктів AnyLogic потрібно коригувати до тих пір, поки не буду пораховано достатнім їх взаємне розташування для коректної точки зору роботи моделі та її презентації.

Даний сегмент призначений для імітації надходження заготовки, очікування в черзі, імітації безпосередньо підготовки заготовки і відправки на виконання першої операції.

1. У **Палитре** виділити **Презентация**. Перетягнути елемент **Прямоугольник**. На сторінці **Местоположение и размер** встановити: **Х**: 20, **У**: 300, **Ширина**: 240, в поле **Высота**: 150.

2. Перетягнути елемент **text** і на сторінці **Текст** панелі **Свойства** ввести: «*Подготовка заготовки*».

3. У **Палитре** виділити **Библиотека моделирования процессов**. Перетягнемо об'єкт **source** на агента **Main** і розмістимо в прямокутнику з ім'ям «*Подготовка заготовки*».

4. Для запису і зберігання параметрів деталі в додаткові поля заявок потрібно створити нестандартний клас заявки. Створити клас заявки «*Detail*».

5. В панелі **Проект** натиснути правою кнопкою миші елемент моделі верхнього рівня дерева і обрати **Создать Java класс**.

6. З'явиться діалогове вікно **Новый Java класс**. В поле **Имя:** вводим ім'я нового класу «*Detail*».

7. У полі **Базовый класс:** обрати зі списку «*Entity*» в якості базового класу. Натиснути кнопку **Далее**.

8. З'явиться друга сторінка **Мастера создания Java класса**. Додати наступні поля Java класу, які будуть потрібні в подальшому при розробці моделі: `double n; double a; double Tn1`.

Поле *n* буде використовуватися для занесення номеру виконуваної з деталлю операції. Ці номери необхідні для визначення операцій, на які потрібно повторно відправити деталі з пункту кінцевого контролю. Поле *a* призначене для занесення коду першого або повторного виконання операцій. Цей код використовується для того, щоб при повторному виявленні браку не відправляти для виконання відповідної операції деталь втретє.

9. Залишити обраними прапорці **Создать конструктор** і **Создать метод toString()**.

10. Натиснути кнопку **Готово**. З'явиться редактор коду і автоматично створений код нашого Java класу. Закрити код.

11. Тепер потрібно перетворити Java клас в тип агента. Для цього натиснути правою кнопкою миші в панелі **Проект** щойно створений Java клас і в контекстному меню обрати **Преобразовать Java класс в тип агента**.

12. З'явиться вікно с автоматично створеними параметрами нестандартного типу заявок «*Detail*».

13. З палітри **Основная** перетягнути на сегмент **Исходные данные** елемент **Переменная**. На сторінці **Основные** панелі **Свойства** дати **Имя:** «*b*», встановити **Тип:** «*double*».

14. Виділити об'єкт **source**. На сторінках панелі **Свойства** встановити властивості.

1. Виділити об'єкт **queue** і на сторінках панелі **Свойства** встановити властивості:– **Имя**: «очЗаг»; – **Тип заявки**: «Detail». **Максимальная вместимость**: встановити прапорець.

3. Виділити об'єкт «*delay*» і на сторінці **Основные** панелі **Свойства** встановити властивості:

**Имя**: «подЗаг»;

**Тип заявки**: «Detail»;

**Тип**: **Определённое время**;

**Время задержки**: «*exponential (1/entity.Tn1)*»;

**Вместимость**: «1»;

**Действия При выходе**: «*entity.a = 1*».

Кожен із сегментів операцій 1, 2 і 3 призначений для імітації виконання відповідної операції, що включає очікування в черзі, безпосереднє виконання операції, контроль її якості, відправку на пункт контролю готової продукції в разі браку, прийом на повторне виконання операції і контроль.

1. З палітри **Презентация** перетягнути три елементи **Прямоугольник** так, як на рис. 3.8. На сторінці **Местоположение и размер** панелі **Свойства** ввести: для першого прямокутника **X**: 20, **Y**: 60; для другого - **X**: 280, **Y**: 60; для третього - **X**: 530, **Y**: 60. Для всіх **Ширина**: 240, **Высота**: 180.

2. Перетягнути три елементи **text** і на сторінці **Основные** панелі **Свойства** в поле **Текст**: кожного з них ввести *Операция 1*, *Операция 2*, *Операция 3* відповідно

3. З **Библиотеки моделирования процессов** перетягнути для кожного сегменту два об'єкти **queue**, два об'єкти **delay** і один об'єкт **selectOutput**, розмістити і з'єднати як на рис. 3.5.

4. Виділити по черзі об'єкти, починаючи з об'єкта **queue** сегмента *Операция 1*, і на сторінці **Основные** панелі **Свойства** встановити властивості

згідно табл. 3.2. Якщо почати створювати черговий сегмент, то після перетягування іншого об'єкту з'явиться повідомлення: *Ограничение ознакомительной версии: нельзя создавать более 20 вложенных объектов.* Тому інші сегменти моделі потрібно розмістити на іншому агента (активному об'єкті).

1. На панелі **Проекты** натиснути правою кнопкою миші **Main**, з яким проводиться робота в даний момент, і обрати з контекстного меню **Создать/Тип агента.**

2. Відкриється вікно **Шаг 1. Создание нового типа агента.**

3. Залишити **Не использовать шаблоны типов агентов.**

4. Задати в полі **Имя нового агента:** *«Kontrol».*

5. Натиснути кнопку **Готово.**

Для створення пункту завершального контролю виконують таку послідовність дій.

1. З **Презентации** перетягнути три елементи **Прямоугольник** і розмістити. На сторінці **Местоположение и размер** панелі **Свойства** для верхнього прямокутника ввести: **X: 110, Y: 70, Ширина: 580, Высота: 240.** Для нижніх прямокутників: **X: 190, Y: 330, Ширина: 250, Высота: 140; X: 460, Y: 330, Ширина: 250, Высота: 140.**

2. Перетягнемо три елементи **text** і на сторінці **Основные** панелі **Свойства** в полі **Текст:** кожного з них замість наявного там слова **text** ввести – *«Пункт окончательного контроля, Склад готовых деталей, Склад бракованных деталей»* відповідно.

3. З **Библиотеки моделирования процессов** перетягнути два об'єкти **selectOutput**, об'єкт **queue**, об'єкт **delay** і один об'єкт **selectOutput5**, розмістити в верхньому прямокутнику і з'єднати. Порти *брOn1, брOn2* і *брOn3* з'єднуються з входом об'єкта *selectOutput*. Вихід *T (true)* об'єкта *окКон3* з'єднується зі входом об'єкта *selectOutput*.

4. Виділити по черзі об'єкти, починаючи з лівого об'єкта **selectOutput**, і на сторінці **Основные** панелі **Свойства** установити. У всіх об'єктах повинен бути встановлений прапорець **Отображать имя і На презентации**.

За результатами створеної моделі виконано ряд експериментів. Перший експеримент відповідає постановці завдання. В кожному наступному експерименті частина параметрів, що встановлені в попередньому експерименті залишаються незмінними, а інші – змінюються. Вказуються тільки нові значення параметрів в рядку, що підлягають зміні в наступному експерименті. Наприклад, у другому експерименті зменшено середній час надходження заготовок з  $T_n = 35$  до  $T_n = 25$ , а інші параметри – залишилися незмінними.

Всього змінювалися значення шести параметрів. Крім того, у восьмому експерименті були зменшені середні часи підготовки варіантів заготовок (значення елементів одновимірного масиву).

За результатами експериментів можна зробити висновки про ефективність роботи цеху з виготовлення деталей і визначити «вузькі» місця.

#### Висновки до розділу

За результатами опису системи імітаційного моделювання засобами системи Any Logic та надання методики моделювання складних систем на прикладі виготовлення деталі в цеху підприємства можна зробити такі висновки:

– в системі Any Logic застосовано три напрямки моделювання, що дозволяють ефективно застосовувати її для імітаційного моделювання складних систем в різноманітних галузях виробництва, охорона здоров'я і фармацевтика управління активами і проектами, в телекомунікації та інформаційних системах, в соціальних та екологічних системах тощо;

– система є інструментом імітаційного моделювання нового покоління, що розроблений на основі сучасних концепцій в області інформаційних технологій і результатів досліджень в теорії гібридних систем і об'єктно-орієнтованого моделювання, яка дозволяє шляхом анімації і візуального представлення процесів в безперервній моделі динамічної системи врахувати дискретні події, що є характерними для приладобудівного виробництва;

– моделювання складних систем засобами Any Logic передбачає два етапи його застосування – розробка моделі та її аналіз шляхом проведення комп'ютерного експерименту. Це дає можливість вибрати найкращі параметри досліджуваного процесу;

– система Any Logic є надбудовою над мовою Java, що дозволяє користувачу системи розширювати створені моделі за допомогою цієї мови шляхом створення Java аплетів, які можуть бути відкриті будь-яким браузером, та ці аплети дозволяють легко розміщувати моделі AnyLogic на веб-сайтах;

– наведений приклад роботи динамічних систем шляхом моделювання процесу виготовлення деталі показав ефективність застосування системи AnyLogic для моделювання складних виробничих систем в приладобудуванні.

## РОЗДІЛ 4

### ПРАКТИЧНЕ ВИКОРИСТАННЯ РОЗРОБЛЕНОЇ МЕТОДИКИ ТА СИСТЕМИ ДЛЯ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ ВИРОБНИЧИХ ПРОЦЕСІВ В ПРИЛАДОБУДУВАННІ

4.1. Імітаційне моделювання роботи виробничих виробничої дільниці механічного цеху при виготовленні деталі

З метою практичної апробації методики імітаційного моделювання виробничих систем засобами Anylogic 8.0.3 розроблено модель процесу виготовлення деталі «корпус».

Креслення деталі та технологічний процес її виготовлення представлено в Додатку А. Процес виготовлення деталей складається з 30 операцій, що виконуються на восьми видах технологічного обладнання, ці дані представлено в таблиці 4.1.

Відповідно з послідовністю виконання операцій технологічного процесу обробки деталі на дільниці механічного цеху, де розміщено верстати кожного виду, створено схему імітаційної моделі засобами системи Anylogic.

При побудові цієї схеми використано блоки панелі інструментів системи **Source**, **moveTo**, **quene**, **service**, **moveTo1**, **selectOutput**, **moveTo2**, **delay** та **sink**.

Для створення імітаційної моделі послідовно виконано команди **Файл / Создать / Модель** на панелі інструментів та у полі **Имя модели** діалогового вікна **Новая модель** введено *«Изготовление\_детали»*. Також обрано каталог, в якому були збережені файли створюваної моделі.

Для успішної реалізації створюваної моделі створено компоненти: параметрів, змінних та блоку набору ресурсу. Для того, щоб задати компоненти агента **Параметр** окремо для кожного з наявних верстатів обробки деталі в меню **Палитра** обрано елемент **«Агент»/Компоненты**

**агента/Параметр.** Для восьми наявних верстатів надано окремі довільні назви цих параметрів: «*n\_stanok1*»; «*n\_stanok2*»; «*n\_stanok3*»; «*n\_stanok5*»; «*n\_stanok7*»; «*n\_stanok11*»; «*n\_stanok29*»; «*n\_stanok30*».

Таблиця 4.1. Перелік операцій, обладнання та часу виконання обробки.

№	Назва операції	Обладнання	Час виконання, хв
005	Лиття під тиском		10,112
010	Фрезерна	6P12	0,255
015	Слюсарна	Верстак слюсарний	0,2
020	Слюсарна	Верстак слюсарний	0,5
025	Свердлильна	2H125	0,455
030	Слюсарна	2H125	0,2
035	Координатно-розточна	2E440 A	2,15
040	Слюсарна	2E440 A	0,2
045	Свердлильна	НС-12	1,334
050	Слюсарна	НС-12	0,2
055	Токарна	CTX beta1250TC 4A	1,372
060	Координатно-розточна	2E440 A	0,926
065	Слюсарна	2E440 A	0,2
070	Координатно-розточна	2E440 A	0,316
075	Слюсарна	2E440 A	0,55
080	Фрезерна	6P12	0,551
085	Слюсарна	6P12	0,2
090	Фрезерна	6P12	0,251
095	Слюсарна	6P12	0,2
100	Фрезерна	6P12	0,238
105	Слюсарна	6P12	1,1
110	Фрезерна	6P12	0,279
115	Слюсарна	6P12	0,2
120	Свердлильна	НС-12	0,284
125	Слюсарна	НС-12	0,2
130	Свердлильна	НС-12	0,901
135	Слюсарна	НС-12	0,2
140	Слюсарна	Верстак слюсарний	1,15
145	Промивання	Верстак з витяжкою	0,25
150	Контроль	Стіл контролю	2,5

У властивостях **Параметри** встановлено **Тип** – «*int*» для кожного з них, а також **Значення по умовчанию** прирівняно до «1». У властивостях параметрів залишено галочку **Отображать імя**.

Наступний пункт – створено змінні для відображення та взаємодії часових значень. Для цього в меню **Палитра** обрано елемент

«Агент»/Компоненты агента/Переменная. Створено вказаним методом змінні для всіх 30 операції, що задані в технологічній документації. У властивостях змінних присвоєно кожній ім'я  $t_1, t_2, t_3 \dots t_{30}$ . Обрано Тип = «double» та в графі Начальное значение вказано час виконання для кожної операції, яку представляє створена змінна.

Для імітації роботи обладнання створено Блок **Resource Pool**, який задає набір ресурсів певного типу. Такий блок знаходиться в меню **Палитра/Библиотека моделирования процессов/Блоки**. Вісім таких блоків будуть імітувати роботу верстатів. Кожен із них мають ім'я відповідно до назви параметрів, які також відображають роботу обладнання («n\_stanok1»; «n\_stanok2»; «n\_stanok3»; «n\_stanok5»; «n\_stanok7»; «n\_stanok11»; «n\_stanok29»; «n\_stanok30»).

У властивостях Блоку задано Тип = *Статический*; **Количество задано = Напрямую**.

Після цього створюється схема функціонування алгоритму моделі, який складається з блоків (рис.4.1). Блок **Source**, що знаходиться в меню **Палитра/Библиотека моделирования процессов/Блоки** за параметрами присвоєно наступні значення: **Прибывают согласно = Времени между прибытиями**; **Время между прибытиями =  $t_2$** ; **За 1 раз создается несколько агентов = Да**; **Количество агентов прибывающих за 1 раз =  $n_1$** .

Потім задаються додаткові змінні « $n_1$ » та параметри « $t_2$ », їх створення аналогічне вже описаному.

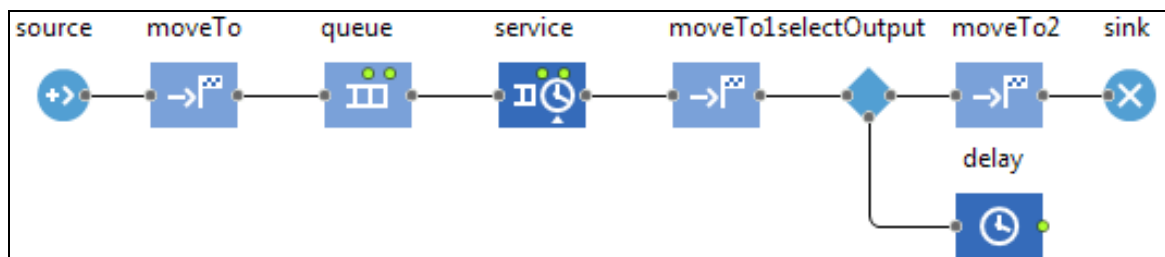


Рис.4.1. Алгоритм моделі роботи дільниці механічного цеху виготовлення деталі «корпус».

Наступним блоком в алгоритмі є **moveTo**, що розташований – **Палитра/Библиотека моделирования процессов/Блоки**. Блоку присвоєно наступні параметри властивостей: **Агент** = *перемещается*; **Место назначения** = *Узел сети / ГИС*; **Узел** = *area*; **Перемещение задается** = *Расстоянием / скоростью*. Зображення параметрів даного блоку показане на рис 4.2.

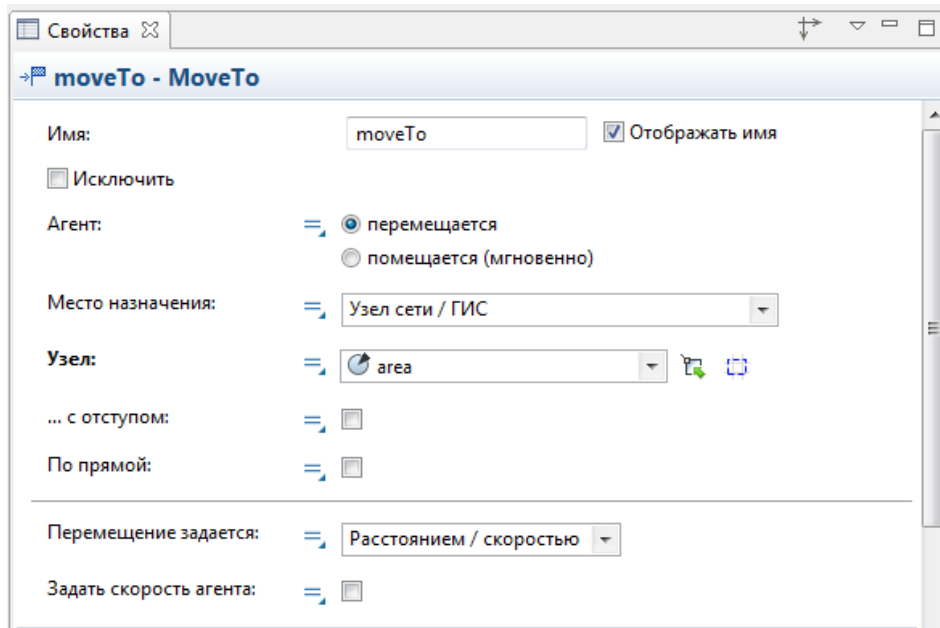


Рис.4.2 Параметры блоку moveTo.

Далі створено блок **queue**, який знаходиться – **Палитра/Библиотека моделирования процессов/Блоки**. Цей блок отримав такі параметри: **Максимальная вместимость** = *Да*; **Место агентов** = *area*.

Блок **service**, який знаходиться **Палитра/Библиотека моделирования процессов/Блоки**. Його параметри: **Захватить** = *(альтернативный) набор ресурсов*; **Наборы ресурсов** = *{{stanok}}*; **Максимальная вместимость** = *Да*; **Время задержки** = *t*; **Место агентов** = *area*.

Наступний блок **selectOutput** – **Палитра/Библиотека моделирования процессов/Блоки**. Параметри: **Выход true выбирается** = *При выполнении условия*; **Условие** = *agent.inState(detail.s31)*.

Далі блок **delay** – Палитра/Бібліотека моделювання процесов/Блоки. Його параметри: Тип = *Определенное время*; **Время задержки** = 0,001; **Максимальная вместимость** = *Да*.

Також, аналогічно блок **sink** – Палитра/Бібліотека моделювання процесов/Блоки. Параметри: Тип агента = *detal*.

На рис.4.3 наведено всі змінні, параметри та блоки, що створені для функціонування моделі.

Кожен з вибраних блоків відповідає за окрему роботу моделі, їх загальні характеристики представлено в таблиці 4.2. Блок **Source** створює заявки заготовок деталі типу *detal*. Заявки створюються кожні  $t_2$  хвилини в кількості –  $n1$ . Значення параметрів  $t_2$  і  $n1$  можна задавати в інтерфейсі моделі в полях «*Время между прибытиями деталей в цех*» та «*Количество деталей, прибывающих за один раз*».

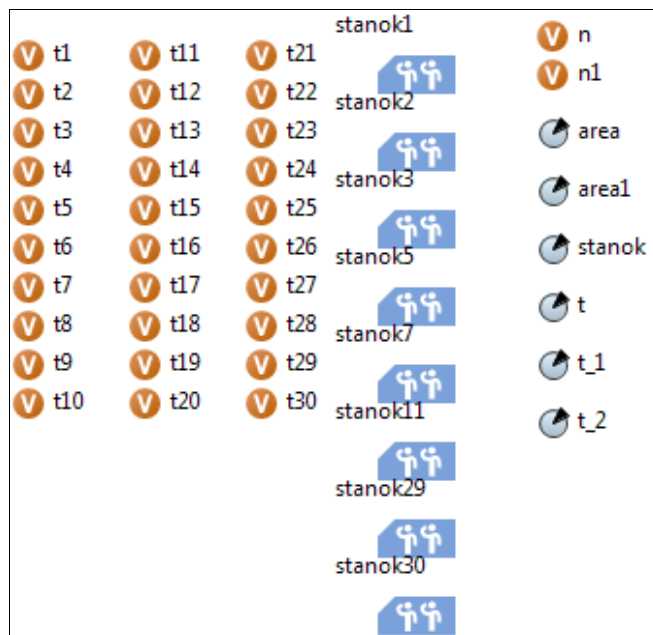


Рис.4.3. Змінні, параметри та блоки, що створені для функціонування моделі.

Агенти типу *detal* можуть перебувати в одному з 31 станів. Залежно від стану заявки, програма визначає на яку операцію обробки необхідно відправити заявку. При створенні заявки блоком **Source** вона перебуває в першому стані  $s1$ . Після обробки заявки на першому верстаті її стан

Таблиця 4.2. Загальні характеристики використаних блоків.

Блок	Характеристика
Source	Створює агентів
Move To	Переміщає агента в нове місце
Queue	Моделює чергу або буфер агентів, які очікують прийняття наступним блоком в діаграмі процесу
Service	Захвачує для агента задану кількість ресурсів, затримує агента на визначений час, а потім звільняє захвачені ресурси
Select Output	Направляє агентів, що входять в об'єкт на один з двох вихідних портів згідно з заданою вірогідністю.
Delay	Затримує агетів на заданий період часу
Sink	Знищує агентів, що надійшли

змінюється на  $s_2$ , таким чином, програма розуміє в якій послідовності, куди відправляти її далі. Кожному стану відповідає своя область, в яку слід перемістити заготовку і час обробки на необхідному обладнанні. За переміщення заявки відповідає блок **moveTo**. Місце призначення заявки в даному блоці визначається змінною *area* типу *AreaNode*. Дана змінна може приймати значення *node1*, *node2*, *node3* і т.д. – що є прямокутними вузлами розмітки простору. При прибутті заявки в блок **moveTo**, в залежності від її стану, визначається необхідний вузол.

Блок **queue** вибудовує заявки в чергу в разі зайнятості ресурсу верстата з черговістю «перший прийшов, перший пішов».

При попаданні заявки в блок **service**, також в залежності від стану заявки, визначається місце очікування затримки в змінну *area*, місце затримки в змінну *area1* (тип *PointNode*), необхідний ресурс в змінну *stanok* (тип *ResourcePool*) і час обробки заявки *t*:

```
if (agent.inState(detail.s1)) {area=node1;area1=point1; stanok=stanok1,
t=t1;};
```

```
if (agent.inState(detal.s2)) {area=node2;area1=point2; stanok=stanok2,
t=t2;}; t1=t2 ... t=30
```

Після обробки заявки в блоці **service**, заявки відправляється в блок **moveTo1**, який відправляє її назад в прямокутний вузол *area*. Далі йде розгалуження блоком **selectOutput**. Даний блок перевіряє стан заявки. Якщо стан заявки *s31* (пройдено 30 станів, тобто пройдено 30 операцій обробки), то заявка відправляється на вихід з діаграми моделі. Інакше, заявка відправляється в блок **moveTo** для завершення відсутніх операцій.

Блок **delay** не несе змістовного навантаження і необхідний для правильного функціонування блоку **moveTo** після розгалуження.

Для використання графічного зображення роботи моделі в 2D, створено область перегляду – **Палитра/Область просмотра**. Задано ім'я області у її параметрах **Имя** = «*v1*». Обрано **Масштабирование** = *подогнать под окно*. Встановлено розміри вікна області перегляду **Масштабирование и размер: X = 10; Y = 150; Ширина = 1000; Высота = 600**. Розташовуємо в області прямокутники з назвами операцій, які зображуватимуть обладнання, що виконує операції. Для кожної з переліку операцій додано діаграму, яка графічно показує коефіцієнт завантаження та його числове значення.

В створеній області розміщено перелік операцій у стовпчик, які під час роботи моделі виділяють виконувані операції в заданий проміжок часу та виникаючі черги на них. Дана область перегляду зображена на рис 4.4.

Для графічного зображення в 3D створено ще одну область перегляду з іменем «*v1*» та становлено розміри вікна області перегляду **Масштабирование и размер: X=1570; Y=150; Ширина=1000; Высота=600**.

Області перегляду використовуються як для навігації редактору під час створення моделі, так і для навігації по вікну презентації під час виконання моделі. Для того, щоб перейти до іншої області перегляду в режимі створення моделі потрібно натиснути мишею в графічному редакторі. Для задання його активним треба натиснути кнопку панелі інструментів **Області**



Рис. 4.4. Область перегляду створеної моделі в 2D.

перегляду, вибрати зі списку область перегляду, до якої необхідно перейти.

Для переходу до іншої області перегляду в режимі виконання моделі треба натиснути правою кнопкою миші в області вікна презентації, обрати пункт контекстного меню **Область** і вибрати зі списку, до якої саме області перегляду необхідно перейти.

В графічних вікнах, що розташовані праворуч від переліку операцій в області перегляду і вказують на заданий час виконання операції, можна змінювати час обробки для кожної конкретної операції.

В області перегляду моделі зображено 2D представлення дільниці цеху (рис.4.5), в якій схематично зображене обладнання, що використовується при обробці відповідно до операцій, та коефіцієнти завантаження обладнання. Для можливості зміни кількості обладнання в графічній частині, розташовані елементи керування кількістю обладнання для кожного з 8 робочих місць, які пов'язані з параметрами створеного раніше обладнання: («*n\_stanok1*»;

«n\_stanok2»; «n\_stanok3»; «n\_stanok5»; «n\_stanok7»; «n\_stanok11»; «n\_stanok29»; «n\_stanok30»).



Рис.4.5. Схема 2D представлення дільниці цеху моделі обробки корпусу.

Представлена модель імітаційного моделювання механічної обробки корпусу надає результати виконання кількості повної обробки корпусів за годину, а також відомості про кількість готових виробів за проведений експеримент.

Після проведення модельного експерименту з обробки 50 корпусів, отримано наступні результати (рис.4.6): коефіцієнти завантаження, на деякому обладнанні приймають значення близьке до 0.9, що свідчить про високий рівень їх завантаження, але кількість оброблених корпусів за годину складає всього 5.49 шт.

Проведення експерименту даної моделі по обробці 50 корпусів показало, що на робочих позиціях, де виконуються слюсарна та фрезерна операції виникає черга, у зв'язку з великою кількістю даних операцій на цих позиціях згідно з технологічним процесом (рис. 4.7).

Для збільшення темпу виготовлення, підвищення продуктивності процесу виготовлення деталей та оптимізації роботи обладнання пропонується змінити кількість використовуваних робочих місць, а саме: збільшити кількість обладнання, на якому виконується фрезерна операція, до 3, а слюсарна операція – до 5.

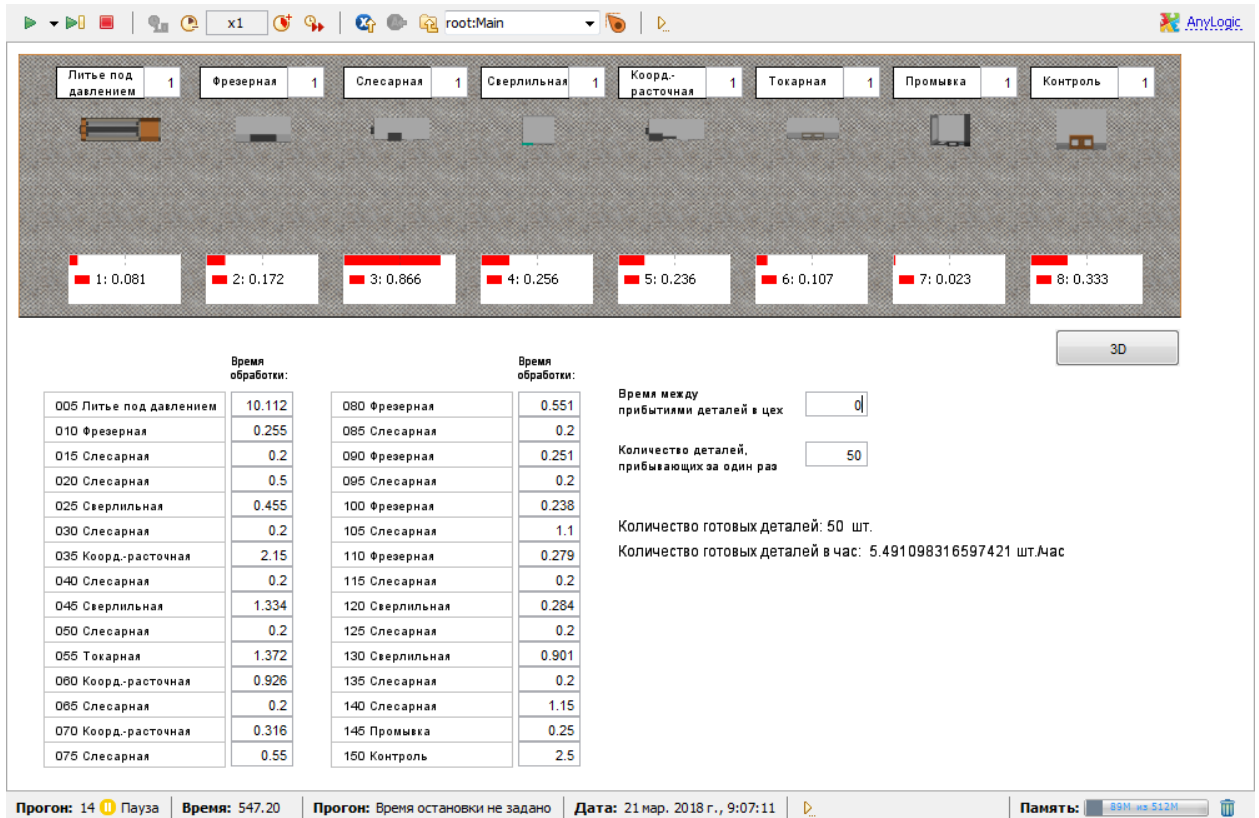


Рис.4.6 Результаты проведенного эксперимента по обработке 50 корпусів.

Під час комп'ютерного експерименту шляхом використання імітаційної моделі обробки корпусу в області перегляду схематично зображуються жовтим кольором раніше створені заявки, які імітують етапи проходження операцій, також зображено зеленим кольором виконувану операція в даний момент часу і червоним – чергу, яка виникла на робочому місці (рис.4.7.).

Проведений модельний експеримент показав, що при використанні одночасно трьох фрезерних верстатів замість одного, а також п'яти слюсарних верстатів замість одного, при виготовленні 50 корпусів отримано наступні результати: коефіцієнти завантаження обладнання складають 0.72; кількість оброблених корпусів за годину становить 13.8 (рис.4.8).

Аналізуючи отримані результати проведенного експерименту можна зробити висновок, що запропонована зміна кількості використовуваного обладнання значно підвищує ефективність технологічного процесу шляхом збільшення кількості готової продукції за годину з коефіцієнту 5.49 до 13.8

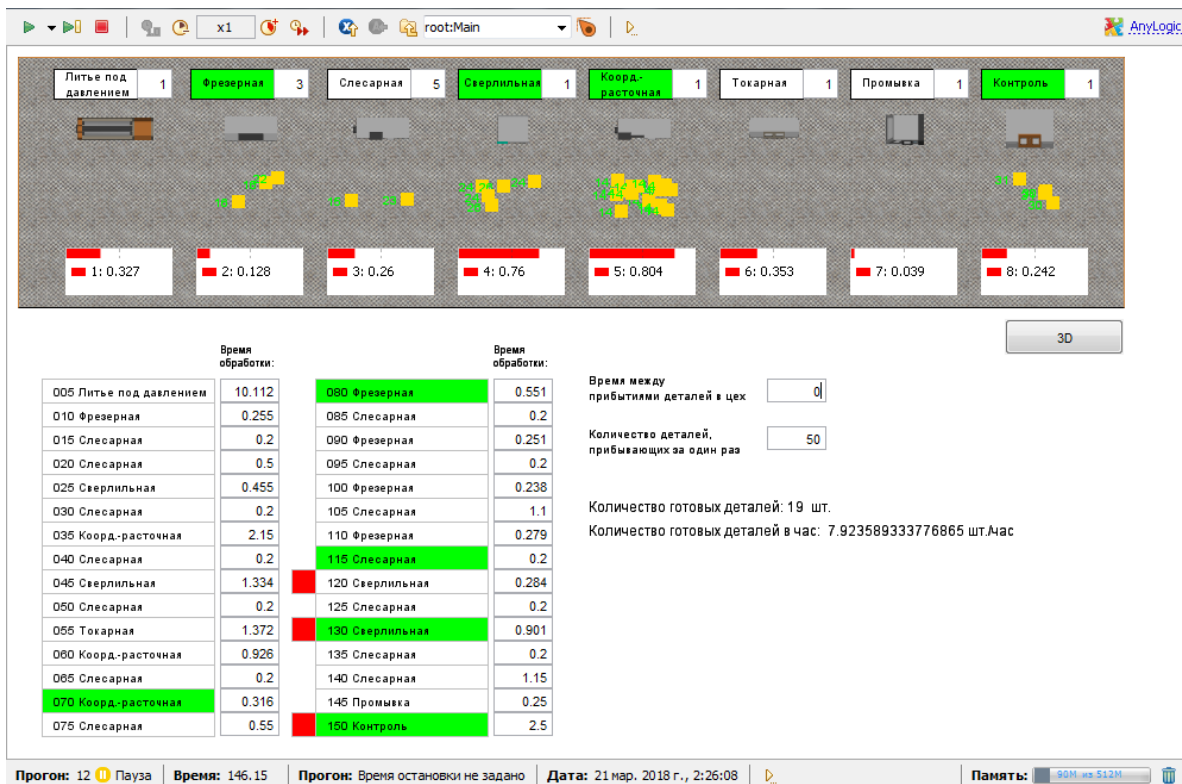


Рис.4.7 Зображення утворення черги при обробці 50 корпусів при імітаційному моделюванні технологічного процесу.

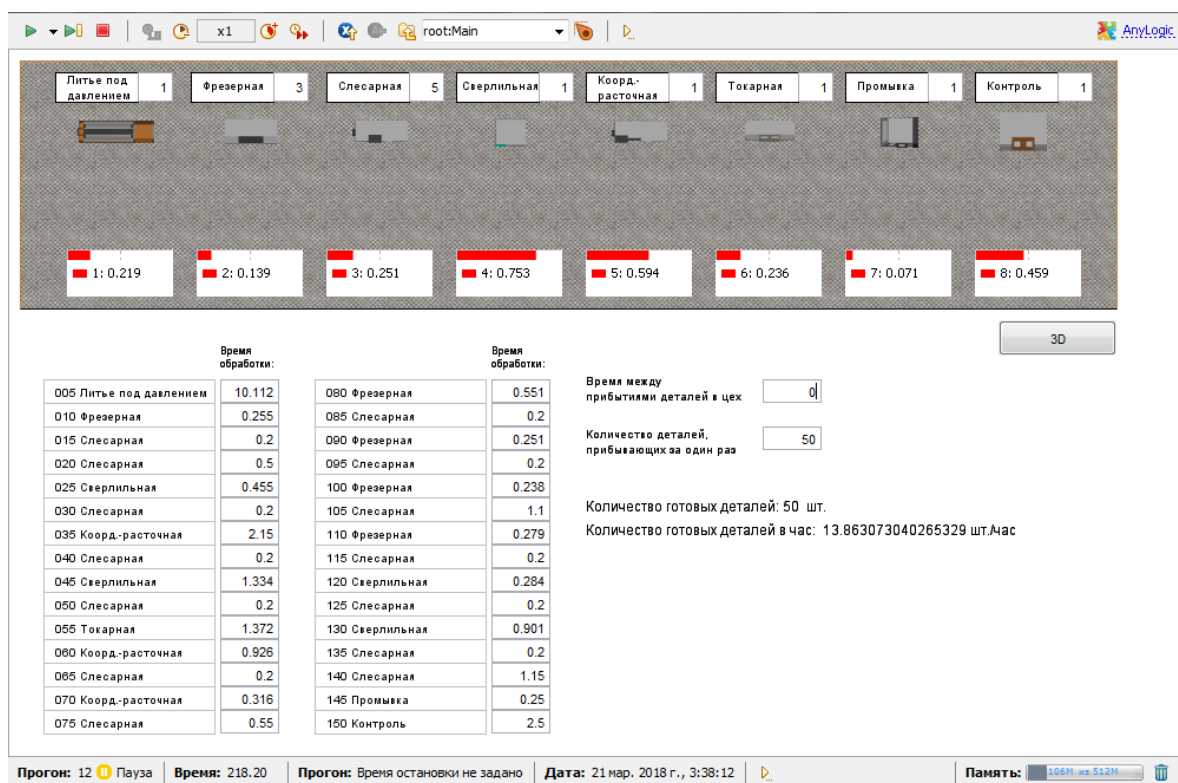


Рис.4.8 Результаты проведенного компьютерного эксперимента по обробці 50 корпусів.

що є перевагою процесу технологічного процесу і може бути враховано при створенні реальної лінії виготовлення корпусу і подальшого його реалізації в умовах мінімальних витрат на використання обладнання .

#### 4.2. Імітаційне моделювання роботи ділянки складального цеху при складанні хвильового редуктору

З метою практичної апробації методики імітаційного моделювання виробничих систем засобами Anylogic 8.0.3 розроблено імітаційну модель лінії складання хвильового редуктору.

Креслення редуктора, схема складального складу, технологічна схема складання та технологічний процес складання на картах представлено в ДОДАТКУ Б. Техноогічна схема складання редуктора зображена на рис. 4.9. Хвильовий редуктор складається з двох основних складальних одиниць – вхідний вал та вихідний вал, що компонуються в готовий виріб. Для успішної побудови імітаційної моделі, з урахуванням технології складання редуктора, лінія складання повинна мати три робочі позиції, кожна з яких буде імітувати процес складання окремої складальної одиниці. В процесі складання хвильового редуктора використано автоматизоване обладнання: автоматичний прес, робот-маніпулятор та автоматичний гайковерт. Час виконання складання вхідного валу  $t_1=220$  с.; складання вихідного валу  $t_2=160$  с.; загальне складання редуктора  $t = 45$  с.

Процес складання редуктора складається з трьох етапів: складання вхідного валу, складання вихідного валу та складання редуктора із зібраних валів. На ділянці складання розташовані три робочих місця, на одному з них виконується складання редуктора, а на двох інших – збираються вхідний і вихідний вали. Біля робочих місць для складання вхідного і вихідного валів є стелажі з деталями для складання кожного валу. Складені редуктори розміщуються на стелажих зберігання готових виробів. Кожен стелаж



Для реалізації описаного в моделі, створено схему роботи імітаційної моделі складання хвильового редуктору, яка представлена на рис 4.11.

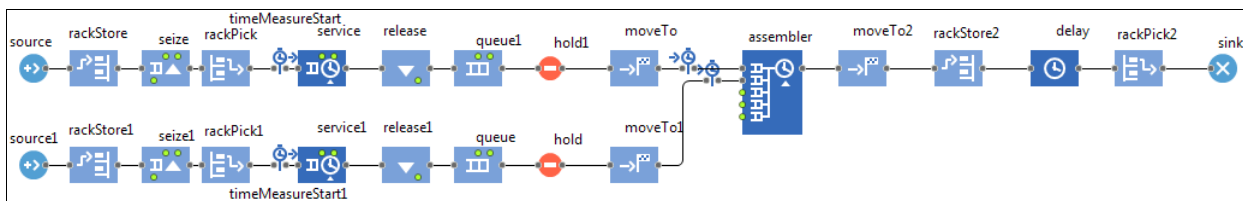


Рис. 4.11. Схема імітаційної моделі складання хвильового редуктору.

Для створення імітаційної моделі послідовно виконано команди **Файл / Создать / Модель** на панелі інструментів та у полі **Имя модели** діалогового вікна **Новая модель** введено «Сборка\_редуктора». Також обрано каталог, в якому були збережені файли створюваної моделі.

Для успішної реалізації створюваної моделі створено компоненти: агентів, змінних та блоку набору ресурсу (рис.4.12). Для того, щоб задати компоненти **Параметр**, окремо для кожного з наявних агентів, які задіяні в роботі моделі, в меню **Палитра** обрано елемент «Агент»/Агент. Для шести агентів надано окремі смислові назви: «val1»; «val2»; «izdelie»; «pogruchik»; «pogruchik2»; «pogruchik3».

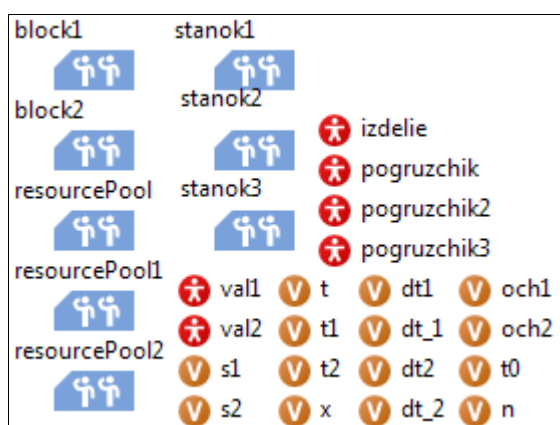


Рис.4.12. Параметри, блоки та агенти, що створені для функціонування імітаційної моделі.

У властивостях **Параметрів** встановлено значення **Одиночний агент** = «Да» для кожного з них, а також залишено галочку **Отображать имя**.

Наступний пункт – створено змінні для відображення та взаємодії часових значень. Для цього в меню **Палитра** обрано елемент «Агент»/Компоненты агента/Переменная. Створено вказаним методом змінні для складання окремих вузлів редуктора та допоміжні змінні для побудови зв'язків моделі між агентами. У властивостях змінних присвоєно кожній довільне ім'я: «s1»; «s2»; «t1»; «t2»; «t»; «x»; «dt1»; «dt\_1»; «dt2»; «dt\_2»; «och1»; «och2»; «t0»; «n». Для змінних «t1»; «t2»; «t» обрано **Тип** = «double» та в графі **Начальное значение** вказано час виконання для кожної складальної операції, яку представляє створена змінна. Для змінних «dt1»; «dt\_1»; «dt2»; «dt\_2»; «t0» також обрано **Тип** = «double». Решті змінних: «och1»; «och2»; «s1»; «s2»; «n»; «x» обрано **Тип** = «int».

Для імітації роботи складання обладнанням створено **Блок Resource Pool**, який задає набір ресурсів певного типу. Такий блок знаходиться в меню **Палитра/Библиотека моделирования процессов/Блоки**. Кожен з них підписаний відповідно до логічної назви виконуваних параметрів: («block1»; «block2»; «resourcePool»; «resourcePool1»; «resourcePool2»; «stanok1»; «stanok2»; «stanok3»). У властивостях Блоку задано **Тип** = *Статический*; **Количество задано** = *Напрямую*; **При уменьшении кол-ва** = «ресурси сохраняются».

При побудові імітаційної схеми складання хвильового редуктора використано блоки панелі інструментів системи: **source**, **source1**, **rackStore**, **rackStore1**, **seize**, **seize1**, **rackPick**, **rackPick1**, **service**, **service1**, **release**, **release1**, **queue1**, **queue**, **hold1**, **hold**, **moveTo**, **moveTo1** та **assembler**.

Опишемо виділені блоки. Блок **Source**, що знаходиться в меню **Палитра/Библиотека моделирования процессов/Блоки** за параметрами присвоєно наступні значення: **Прибывают согласно** = *Вызовам функции inject*; **Местоположение прибытия** = *Узел сети*; **Новый агент** = *vall*.

Наступним блоком в алгоритмі є **rackStore**, що розташований – **Палитра/Библиотека моделирования процессов/Блоки**. Блоку присвоєно наступні параметри властивостей: **Стеллаж/Зона хранения** = *palletRack*; **Ячейка** = *Выбирается автоматически*; **Выбрать ячейку, ближайшую к** = *Началу стеллажа/зоны хранения*; **Переместить агента в** = *ячейку (с повышением уровня)*; **Время поднятия на уровень** = 0. Зображення встановлених параметрів блоку показано на рис. 4.13.

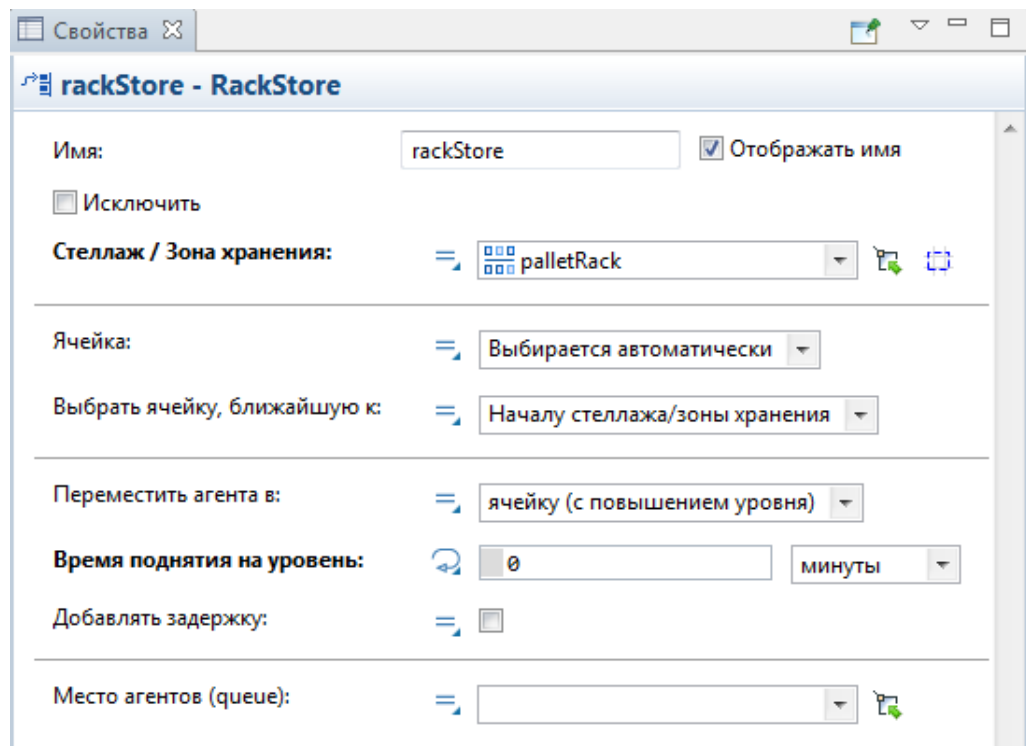


Рис.4.13. Параметры блоку rackStore.

Для створення блоку **seize**, який знаходиться – **Палитра/Библиотека моделирования процессов/Блоки**, встановлено наступні параметри: **Захватить** = *(альтернативный)набор ресурсов*; **Набор ресурсов** = *block1*; **Правило захвата** = *захватить весь набор сразу*; **Максимальная вместимость** = *Да*.

Наступним створеним блоком є **rackPick**, що розташований – **Палитра/Библиотека моделирования процессов/Блоки**, присвоєно параметри: **Стеллаж/Зона хранения** = *palletRack*; **Место назначения** = *Узел*

*сети*; **Взять агента из** = ячейки (с повышением уровня); **Время поднятия на уровень** = 0.

Аналогічно блок **service**, знаходиться **Палитра/Библиотека моделирования процессов/Блоки**. Йому присвоєно параметри: **Наборы ресурсов** = *stanok1*; **Вместимость очереди** = 100; **Время задержки** = *t1*.

Блок **release** знаходиться в розділі **Палитра/Библиотека моделирования процессов/Блоки**. Він містить такі параметри: **Освободить** = *Все ресурсы, захваченные данным блоком* *Seize*; **Объекты Seize** = *seize*; **Движущиеся ресурсы** = *Возвращаются в базовую точку*.

**Queue** – блок, що знаходиться **Палитра/Библиотека моделирования процессов/Блоки**, має параметри: **Максимальная вместимость** = *Да*.

Наступний блок **hold**, що розташований **Палитра/Библиотека моделирования процессов/Блоки**, містить параметри: **Режим** = *Вручную*.

Блок **moveTo**, як і аналоги, знаходиться **Палитра/Библиотека моделирования процессов/Блоки**. Має наступні параметри: **Место назначения** = *Узел сети*; **Агент** = *перемещается*.

Наступний блок **assembler** – **Палитра/Библиотека моделирования процессов/Блоки** має такі параметри: **Новый агент** = *izdelie*; **Захватить** = *Набор ресурсов*; **Набор ресурсов** = *stanok3*.

Блоки **source** і **source1** створюють заявки вхідного і вихідного валу *val1* і *val2*. Блоки **rackStore** і **rackStore1**, використовуючи ресурси типу *pogruchik* і *pogruchik2*, поміщають заявки – коробки з заготовками до відповідних складів. Блоки **seize** і **seize1** використовують ресурси верстатів, для того, щоб коробки із заготовками потрапляли до своїх верстатів за чергою. Далі, використовуючи ресурси типу *pogruchik* і *pogruchik2*, блоки **rackPick** і **rackPick1** забирають відповідні коробки із заготовками зі складів і перевозять їх до складальних робочих місць. У блоках **service** і **service1** виникає затримка коробок з заготовками на відповідний час збирання валів.

Після складання кожного з валів ресурси робочих місць звільнюються блоками **release** і **release1**.

Далі, у тому разі, якщо робоче місце складання редуктора не звільнилось, коробки з заготовками будуть поставлені в чергу в блоках **queue1** і **queue**. Блоки **hold1** і **hold** необхідні для того, щоб складені вхідний і вихідний вали потрапляли на складання редуктора одночасно. Блоки **moveTo** і **moveTo1** переміщують зібрані вали в зону їх заключного складання готового редуктора. Блок **assembler** виконує складання валів редуктору, поєднуючи типи заявок **val1** і **val2** в нову заявку типу «*izdelie*». Далі відбувається переміщення зібраного валу на склад готових виробів, які через час залишають склад складального цеху підприємства.

Для використання графічного зображення роботи імітаційної моделі в 2D створено область перегляду – **Палитра/Область просмотра**. Задано ім'я цієї області у її параметрах **Имя** = «*v1*». Обрано **Масштабирование** = *подогнать под окно*. Встановлено розміри вікна області перегляду **Масштабирование и размер: X = 1290; Y = 0; Ширина = 1000; Высота = 600**. В області перегляду розміщено текстові форми, що поєднані з агентами для відображення необхідних показників під час роботи моделі: загальний час складання, черга складання вхідного валу, черга складання вихідного валу, кількість готових виробів (рис.4.14). Там також розташована діаграма завантаження обладнання та відповідні їх коефіцієнти (рис.4.15). В нижньому лівому куті області перегляду знаходиться інтерфейс керування часом складання кожного з вузлів, який за потребою (за умовою зміни обладнання) можна змінити на необхідний. час. Область перегляду зображена на рис .4.16.

Общее время сборки от входного вала до готового изделия	273
Общее время сборки от выходного вала до готового изделия	215
Очередь входных валов перед сборкой	0
Очередь выходных валов перед сборкой	11
Количество изготовленных изделий (шт.)	50
Скорость изготовления изделий (шт./час)	5.652863662988232

Рис.4.14. Діаграма показників завантаження обладнання.

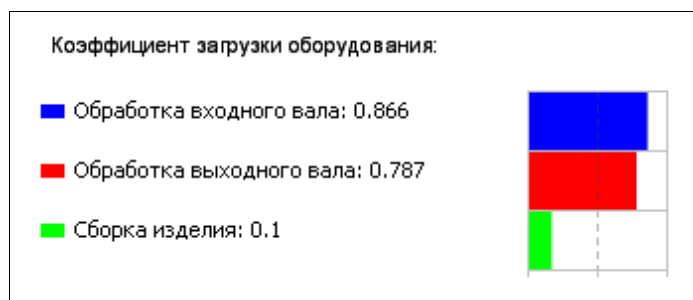


Рис.4.15. Діаграма коефіцієнтів завантаження обладнання.

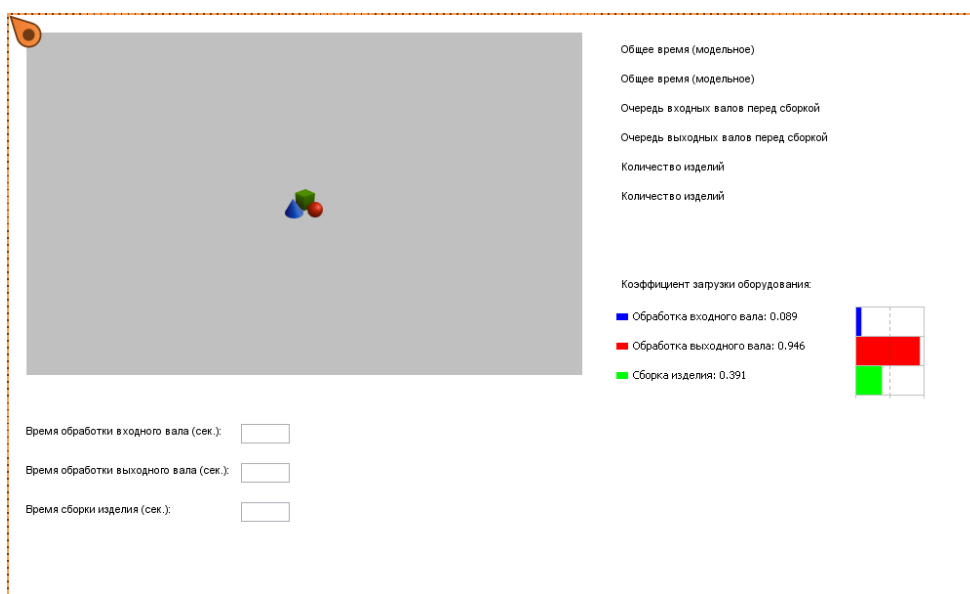


Рис.4.16. Область перегляду імітаційної моделі складання редуктора.

Для зручності та наочності процесу складання редуктора за допомогою меню **Палитра/3D об'єкти** створено область перегляду функціонування автоматизованої лінії в 3D, яка надає більш детальне представлення про процес складання шляхом імітаційного моделювання. Графічна схема

дільниці автоматизованої лінії складання хвильового редуктора представлена на рис.4.17.

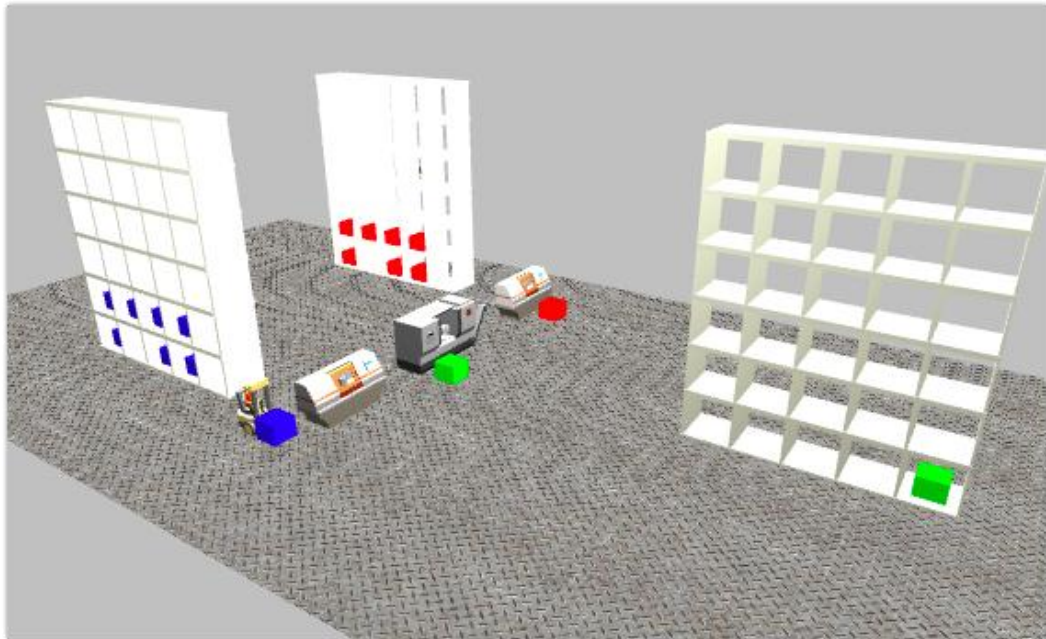


Рис.4.17. Схема області перегляду процесу складання редуктора шляхом імітаційного моделювання в 3D.

Проведення комп'ютерного експерименту з технологічним процесом складання хвильових редукторів, який зображений на рис. 4.18, на основі використання імітаційної моделі показало, що при виготовленні 50-ти готових одиниць хвильового редуктору завантаженість кожного з постів складає:

- пост складання вхідного валу – 86,1% навантаження;
- пост складання вихідного валу – 78.1% навантаження;
- пост складання готового виробу – 17.4% навантаження.

Крім того, аналіз результатів проведеного експерименту шляхом імітаційного моделювання показав, що за таких умов складання 50-ти редукторів виникає черга на посту зі складання вихідного валу в 12 складальних одиниць.

Таким чином, за результатами імітаційного моделювання роботи автоматизованої лінії складання хвильових редукторів можна зробити наступні висновки:

- з метою забезпечення оптимального виробництва потрібно враховувати чергу на кожному з постів складання, а для запобігання простою обладнання потрібно змінити час початку виробництва на посту складання вхідного валу з врахуванням кількості редукторів, що складаються;

- використовуючи систему імітаційного моделювання AnyLogic можлива реалізація достовірного та результативного експерименту, що дозволяє розв'язати задачу оптимізації завантаження робочих місць;

- результати виконання моделювання складання 50-ти хвильових редукторів показали нераціональне використання робочих ресурсів обладнання, що може призвести до зайвих фінансових та часових витрат;

- встановлено очевидні недоліки автоматизованої лінії при побудові постів складання, зафіксовано утворення черги на позиції складання вихідного валу.

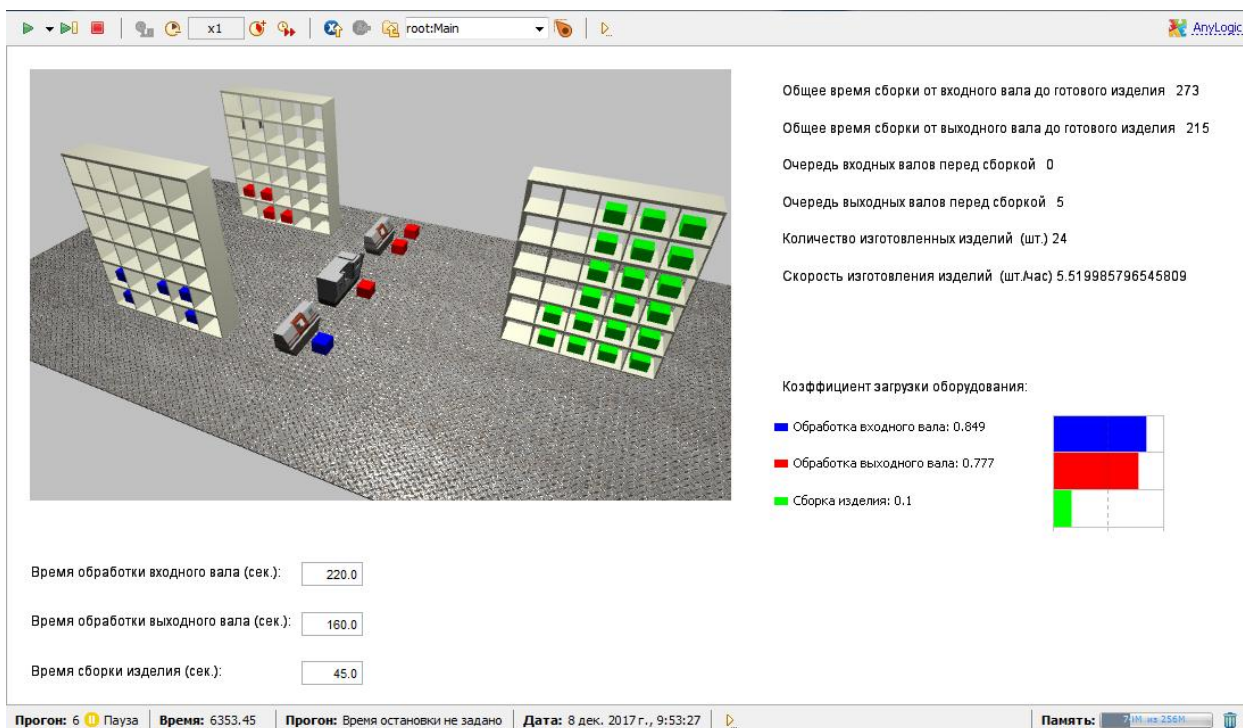


Рис.4.18. Інтерфейс реалізації комп'ютерного експерименту за допомогою імітаційної моделі технологічного процесу складання редукторів.

Для усунення визначених недоліків в роботі автоматизованої лінії складання хвильових редукторів пропонується виконати зміни в роботі обладнання, враховуючи час складання кожного з вузлів. Це дозволить усунути або звести до мінімуму можливість утворення черги при заданих початкових вхідних параметрах шляхом необхідного корегування технологічного процесу.

4.3. Методичні рекомендації зі створення імітаційної моделі виробничої системи та проведення відповідного комп'ютерного експерименту

На сьогодні основне призначення імітаційного моделювання на підприємствах пов'язане з перебудовою та оптимізацією існуючих виробничих процесів або створення нових ВС. Для постійного використання засобів імітаційного моделювання необхідно регулярно оновлення інформації про досліджуваний об'єкт з метою підтримки актуальності моделі. Однак збирання інформації про виробничу систему (об'єкт дослідження) зазвичай є досить трудомістким процесом, що вимагає певних часових витрат.

З метою використання імітаційного моделювання як засобу аналізу поточної ситуації на підприємстві в будь-який момент часу пропонується методика супроводу виробничої системи на основі даних, що отримані в результаті виконання магістерської роботи. Дана методика складається з декількох етапів:

- проведення детального аналізу виробничої системи (технологічного процесу виготовлення об'єкту моделювання);

- аналіз технологічного процесу виготовлення виробу (операцій, обладнання та інструменту, що використовується, а також можлива заміна обладнання);

- розробка алгоритму роботи виробничої системи, що моделюється;

- вибір блоків імітаційної моделі на панелі інструментів системи AnyLogic;
- завдання параметрів створюваної моделі;
- побудова моделі реальної системи за алгоритмом моделювання ВС;
- збирання необхідної інформації про всі процеси, що проходять або виникають у системі, яка моделюється;
- зміна створеної моделі з метою оптимальної відповідності поставленим задачам технологічного процесу шляхом зміни параметрів;
- проведення комп'ютерного експерименту зі всіма можливими варіантами впливу на результати моделювання;
- аналіз отриманих результатів моделювання ВС;
- оптимізація створеної імітаційної моделі для підвищення ефективності технологічного процесу;
- реалізація рекомендацій щодо подальшого покращення технологічного процесу.

Висновки до розділу.

Наведені в даному розділі приклади застосування системи AnyLogic для імітаційного моделювання складних виробничих систем при виготовлення деталі на дільниці механічного цеху та складання хвильового редуктора в складальному цеху дозволили зробити наступні висновки:

- створена імітаційна модель технологічного процесу виготовлення деталі, що складається з 30 різноманітних операцій технологічного процесу, які виконуються на 8 видах технологічного обладнання, та реалізований інтерфейс дала можливість виконати комп'ютерне моделювання процесу з різними параметрами отриманої моделі. Це дозволило оптимізувати завантаження обладнання виробничої дільниці, визначити раціональний склад використовуваного обладнання при різних розмірах партій виготовлення деталей та виявити «вузькі» місця технологічного процесу;

– створена імітаційна модель технологічного процесу складання хвильового редуктора, що складається з двох основних складальних одиниць – вхідний вал та вихідний вал, що компонується в готовий виріб, дозволила виконати комп'ютерне моделювання процесу складання та визначити, що з урахуванням технологічної схеми складання редуктора лінія складання повинна мати три робочі позиції, кожна з яких буде імітувати процес складання окремої складальної одиниці. Представлена імітаційна модель лінії складання дозволила отримати результати виконання технологічних операцій, завантаження обладнання кожного робочого посту, наявність черги при виконанні операцій та ефективність процесу виготовлення редукторів. Результати виконання моделювання складання 50-ти хвильових редукторів показали нераціональне використання робочих ресурсів обладнання, що може призвести до зайвих фінансових та часових витрат;

– наведена методика використання імітаційного моделювання засобами системи AnyLogic, що включає послідовний аналіз об'єкта моделювання, технологічного процесу її виготовлення, використовуюваного обладнання, інструменту та часу виконання операцій, вибір блоків імітаційної моделі на панелі інструментів системи, завдання параметрів моделі та подальше проведення комп'ютерного експерименту з метою оптимізації технологічного процесу дозволяє підвищити ефективність приладобудівного виробництва.

## РОЗДІЛ 5.

РОЗРОБКА СТАРТАП-ПРОЕКТУ «ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ  
ВИРОБНИЧОЇ СИСТЕМИ В ПРИЛАДОБУДУВАННІ»

## 5.1. Опис ідеї проекту

Розглянувши в попередніх розділах використання імітаційного моделювання у виробництві, було створено виробничу систему імітації виготовлення виробу в системі імітаційного моделювання AnyLogic. В цьому розділі буде проведено аналіз стартап-проекту, який визначить змоги даного продукту вийти на ринок і конкурувати з продуктами, що вже зайняли на ньому своє місце.

Таблиця 5.1. Опис ідеї стартап-проекту.

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Створення імітаційної моделі виробничої системи для проектування технологічних процесів	1. Застосування в різних галузях (управління, проектування, планування, налаштування, прогнозування тощо)	Маючи високу гнучкість, дозволяє адаптувати систему до умов різних виробництв а також дозволяє користуватися системою не маючи відповідних знань з області призначення даної системи
	2. Проектування технологічних процесів	Підвищення ефективності автоматизації технологічного проектування та якості отриманих технологічних процесів

Отже, пропонується нові напрямки проектування технологічних процесів, підвищення ефективності автоматизації технологічного проектування, що реалізовано шляхом комп'ютерної розробки виробничої системи, яка проводить імітацію процесу виробництва приладів та дозволяє покращити і підвищити ефективність налаштування виробничого процесу.

Далі проведено аналіз потенційних техніко-економічних переваг ідеї порівняно із пропозиціями конкурентів:

– визначаємо попереднє коло конкурентів (проектів-конкурентів) або товарів-замінників чи товарів-аналогів, що вже існують на ринку, та проводимо збір інформації щодо значень техніко-економічних показників для ідеї власного проекту та проектів-конкурентів відповідно до визначеного вище переліку;

– проаналізуємо пропонований проект для власної ідеї щодо сильних, слабких та нейтральних характеристик порівняно з конкурентами (табл. 5.2).

Такими конкурентами визначено:

Конкурент 1– Математична модель складних систем в Arena;

Конкурент 2– Математична модель складних систем в Automod;

Конкурент 3– Математична модель виробничих систем в Promodel.

Таблиця 5.2. Визначення сильних, слабких та нейтральних характеристик ідеї проекту.

№ п/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Конкурент1	Конкурент2	Конкурент3			
1	2	3	4	5	6	7	8	9
1.	Створення бібліотек	+	+	-	-		+	
2.	Необмежені потки випадкових чисел	+	+	+	-			+
3.	Розробка власного інтерфейсу	+	-	-	-	+		
4.	Імпорт креслення	+	+	+	+		+	
5.	Використання спеціалізованих мов програмування	+	+	-	-			+

Після порівняння характеристик проекту з конкурентами був визначений перелік слабких, сильних та нейтральних характеристик і властивостей ідеї потенційного товару, що є підґрунтям для формування його конкурентоспроможності.

## 5.2. Технологічний аудит ідеї проекту

В межах даного підрозділу проведено аудит технології, за допомогою якої можна реалізувати ідею створення проекту.

Визначено технологічну здійсненність ідеї проекту, яка передбачає аналіз таких складових (таблиця 5.3):

Таблиця 5.3. Технологічна здійсненність ідеї проекту.

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Проектування технологічних процесів ВС (виробничих систем)	Традиційні мови програмування	Наявні, але потребують значних знань в програмуванні (Java)	Доступні
2	Проектування технологічних процесів ВС	Спеціальний програмний інструментарій (бібліотеки та надбудови до існуючих систем)	Наявні, потребують незначних доопрацювань	В повному об'ємі недоступні, лише демо-версії
3	Проектування технологічних процесів ВС	Готові виробничі системи в вигляді «оболонки», які не містять виробничих баз	Наявні, потребують незначних доопрацювань	В повному об'ємі недоступні, лише демо-версії

Проаналізувавши таблицю можна зробити висновок, що даний проект можна реалізувати тому що всі необхідні технології реалізації даних ідей наявні, але обмежена можливість їх реалізування через недосконале знання мови програмування Java. Необхідно залучати спеціалістів з програмування для реалізації даного проекту.

## 5.3. Аналіз ринкових можливостей запуску стартап-проекту

Визначаємо ринкові можливості, які можна використати під час ринкового впровадження проекту, та ринкові загрози, які можуть перешкодити його реалізації.

Це дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій

проектів-конкурентів. Спочатку проведемо аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 5.4).

Таблиця 5.4. Попередня характеристика потенційного ринку стартап-проекту.

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, грн/ум.од	500 000
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Мінімальна база ВС, швидкість обробки інформації, представлення результатів проектування
5	Специфічні вимоги до стандартизації та сертифікації	Не змінна
6	Середня норма рентабельності в галузі (або по ринку), %	60%

За попереднім оцінюванням ринок має зростаючу динаміку і хороший попит на запропонований нами продукт, тому робим висновок, що ринок є привабливим для входження, хоча на ньому вже існують вітчизняні і іноземні фірми, які працюють багато років і роблять продукцію хорошої якості, але за рахунок нової технології і переваг, які вона надає продукт є конкурентоспроможними.

Далі визначаємо потенційні групи клієнтів, їх характеристики, та формуємо орієнтовний перелік вимог до товару для кожної групи (табл. 5.5).

Таблиця 5.5. Характеристика потенційних клієнтів стартап-проекту.

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	2	3	4	5
1	Системи моделювання виробничих систем	Підприємства з виготовленням продукції технологічного призначення	- наповненість бази системи - ціна продукту - обслуговування	до продукції:- можливість доповнення та редагування; - заміна бібліотеки на свою; до компанії постачальника: - оновлення бібліотек
2	Швидкість та ефективність	Підприємства з виготовленням продукції технологічного призначення	- швидкість обробки інформації - ефективність обробки інформації	до продукції: - простота використання;- можливість редагування; до компанії постачальника: - оперативність налаштування.

Продовження таблиці 5.5

1	2	3	4	5
3	Представлення результатів проектування	Підприємства з виготовленням продукції технологічного призначення	- використання людиною, яка не володіє відповідними знаннями; - наявність інтерфейсу;	до продукції: - простота використання; - доступний інтерфейс; до компанії постачальника: - оперативність налаштування та навчання.

Отже потенційною групою клієнтів продукту мають стати заводи, фірми з різною кількістю робочих центрів і з різною кількістю замовлень, які побачать економічну вигоду у виготовленні виробів за новою методикою.

При застосуванні даної технології існують певні загрози. Для попередження таких ситуацій необхідно якісне обладнання, а також працювати з такими програмами повинні висококваліфіковані фахівці. Також, повинно своєчасне технічне обслуговування даного продукту (таблиця 5.6).

Таблиця 5.6. Фактори загроз.

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Збут	Зацікавлений покупець визначає постачальників за ефективністю роботи розробленої системи	Необхідно пропонувати моделі виробничих систем в невеликі компанії, які не бажають тратити великі гроші на дані системи
2	Конкуренція	Наявність більш великих та наповнених систем	Впровадити методику використання нашої системи впродовж певного часу безкоштовно для зацікавлення покупців
3	Корисність	Може не відповідати вимогам певних підприємств	Пропонувати системи на підприємства, які зацікавлені в оновленні проектування
4	Обладнання	Застаріле програмне забезпечення	Застосування нового наявного інформаційного та програмного забезпечення

В таблиці 5.6 ми визначили фактори загроз які перешкоджають ринковому впровадженню нашого проекту, а також можливу реакцію на фактор щоб звести до мінімуму його вплив.

Але поряд із колом загроз існують і певні можливості (таблиця 5.7).

Таблиця 5.7. Фактори можливостей.

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	2	3	4
1	Наукова новизна	Оновлення системи можна проводити на будь-якому етапі роботи з нею	Проводити оновлення та налаштування системи
2	Корисність	Може бути важливим доповненням для проектування на певних підприємствах	Пропонувати системи на підприємства, які зацікавлені в оновленні проектування
3	Метод проектування виробничих систем	Оптимальний та зрозумілий метод проектування ВС	Спробувати використовувати інші методи проектування ВС для порівняння результатів
4	Попит	Потреба вчасне вдосконалення продукту	Модернізація продукту, розробка нових ідей
5	Економічні	Політика протекціонізму; підтримка інноваційного виробництва.	Підвищення/пониження ціни на продукт; зменшення податкового тиску

В таблиці 5.7 ми визначили фактори можливостей які сприяють ринковому впровадженню нашого продукту, а вигоди які компанія може отримати відповідно від реакції на той чи інший фактор.

Таблиця 5.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1	2	3
1. Тип конкуренції - олігополія	Характеризується невеликою кількістю фірм (від 2 до 10), обгороджених бар'єрами, які перешкоджають вступу в галузь нових фірм, мають контроль над цінами, але при змові з іншими олігополістами. Головною рисою олігополії є те, що кількість фірм така мала відносно розмірів ринку, що кожна з фірм-олігополістів визнає тісний взаємозв'язок одна з одною	Пропонувати системи на підприємства, які зацікавлені в оновленні проектування, проводити оновлення та налаштування системи
2. За рівнем конкурентної боротьби- локальний	Характеризується місцем використання системи	Пропонувати різновиди наповнень бібліотек даних
3. За галузевою ознакою - внутрішньогалузева	Характеризується однією галуззю використання	Пропонувати різновиди наповнень бібліотек

## Продовження таблиці 5.8

1	2	3
4. Конкуренція за видами товарів: - товарно-родова	Характеризується товарами одного виду	Пропонувати різновиди наповнень бібліотек
5. За характером конкурентних переваг - нецінова	Проводиться головним чином за допомогою вдосконалення якості продукції, технології виробництва, інновацій та нанотехнологій, патентування і брендуння і умов її продажу, «сервизации» збуту.	Спробувати використовувати інші методи проектування для порівняння результатів
6. За інтенсивністю - не марочна	Роль торгової марки незначна, хоча самі марки можуть бути присутніми на ринку	Створити (міні) бренд, який міг би легко запам'ятовуватися

В даній таблиці ми проаналізували ринок збуту нашого продукту і визначили загальні риси конкуренції на ньому.

Після аналізу конкуренції проведемо більш детальний аналіз умов конкуренції в галузі.

Таблиця 5.9. Аналіз конкуренції в галузі за М. Портером.

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
Складові аналізу	-Arena -Automod -Promodel	-GPSS -Extend sim	-Америка -Росія	Машинобудівні та приладобудівні підприємства	Системи побудовані за допомогою інших методів
Висновки:	Інтенсивність конкурентної боротьби з боку прямих конкурентів – є значною	- є можливості входу в ринок - є потенційні конкуренти Строки виходу їх на ринок – в цьому році	Постачальники диктують умови роботи на ринку: - відповідно оформлені системи; - дають високі гарантії.	Клієнти диктують умови роботи на ринку: - Відповідне наповнення бібліотек; - Простота інтерфейсу та роботи з системою	Обмеження для роботи на ринку через товари замінники

Отже, відповідно до наведеного вище аналізу головними силами, які діють на конкуренцію в галузі є постачальники і споживачі. Також все

більшого значення набуває інтенсивність конкуренції між існуючими конкурентами.

Таким чином в межах структурного підходу до аналізу конкуренції тип конкуренції – олігополістична конкуренція.

Після всіх аналізів визначається та обґрунтовується перелік факторів конкурентоспроможності. Поки проект не впроваджено в життя, це важко зробити точно, можна дати лише попередню оцінку конкурентоспроможності.

Таблиця 5.10. Обґрунтування факторів конкурентоспроможності.

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Оболонка	Оболонка є в наявності
2	Інтерпретатор	Здійснює перетворення інформації
3	Метод представлення ВС	Вибрано метод проєкцій
5	Рівень мови створення ВС	Використано AnyLogic
7	Можливість редагування;	В головному коді програми
8	Оновлення бібліотек	В головному коді програми, за нестачею інформації
9	Простота використання	Можна використовувати не будучи спеціалістом в даній предметній області

В таблиці 5.10 на основі аналізу проведеного в таблиці 5.9 визначили та обґрунтували фактори конкурентоспроможності нашого проекту.

Таблиця 5.11. Порівняльний аналіз сильних та слабких сторін «моделі ВС».

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з моделлю ВС						
			-3	-2	-1	0	+1	+2	+3
1	Оболонка	1					+		
3	Метод представлення	0				+			
4	Підхід побудови алгоритмів в бібліотеках	1					+		
5	Рівень мови придбання знань	1					+		
6	Тип набутих знань	2						+	
7	Можливість доповнення та редагування;	1					+		
8	Оновлення бібліотек	0				+			
9	Простота використання	0				+			

Порівняльний аналіз сильних і слабких сторін показав, що надійність, довговічність і час роботи, за таку ж саму ціну на продукт, дає перевагу над іншими продуктами і тому проект може стати конкурентоспроможним на ринку.

Таблиця 5.12. SWOT- аналіз стартап-проекту.

<p><b>Сильні сторони:</b></p> <ul style="list-style-type: none"> <li>- необмежені потки випадкових чисел;</li> <li>- використання спеціалізованих мов програмування;</li> <li>- імпорт креслення.</li> </ul>	<p><b>Слабкі сторони:</b></p> <ul style="list-style-type: none"> <li>- розробка власного інтерфейсу;</li> <li>- недостача кількості навчених розробників</li> </ul>
<p><b>Можливості:</b></p> <ul style="list-style-type: none"> <li>- корисність;</li> <li>- попит;</li> <li>- отримання нових замовлень на продукт;</li> <li>- збільшення продажів;</li> </ul>	<p><b>Загрози:</b></p> <ul style="list-style-type: none"> <li>- збут;</li> <li>- конкуренція;</li> <li>- втрата клієнтів через недостатню технічну підтримку;</li> <li>- поява якісніших технологій у конкурентів;</li> </ul>

В таблиці 5.12 проводимо перелік сильних та слабких сторін проект. А також ринкових загроз та ринкових можливостей який складаємо на основі факторів загроз і можливостей який ми склали раніше. Ринкові загрози та можливості на відміну від факторів ще не є реалізованими на ринку та мають певну ймовірність здійснення.

На основі SWOT-аналізу розробляємо альтернативи ринкової поведінки для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок.

Визначені альтернативи аналізуються з точки зору строків та ймовірності отримання ресурсів.

Таблиця 5.13. Альтернативи ринкового впровадження стартап-проекту.

№ п/п	Альтернатива (орієнтовний комплекс заходів) поведінки ринкової	Ймовірність отримання ресурсів	Строки реалізації
1	Стратегія нейтралізації ринкових загроз сильними сторонами стартапу	80%	1 рік
2	Стратегія компенсації слабких сторін наявними ринковими можливостями	60%	1 рік

Проводимо аналіз розроблених нами альтернатив ринкового впровадження і з зазначених альтернатив обираємо ту яка має найбільшу ймовірність отримання ресурсів, а також є найшвидшою в реалізації. Отже обираємо стратегію нейтралізації ринкових загроз сильними сторонами стартапу наявними ринковими можливостями.

#### 5.4. Розроблення ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів.

Таблиця 5.14. Вибір цільових груп потенційних споживачів.

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Машинобудівні та приладобудівні підприємства	70%	90%	80%	15%
Які цільові групи обрано: Стратегія концентрованого маркетингу					

За результатами аналізу потенційних груп споживачів ми обрали цільові групи, для яких будемо пропонувати свою програму для оптимізації робочих центрів та визначили стратегію охоплення ринку: стратегію концентрованого маркетингу.

Для роботи в обраному сегменті ринку необхідно сформувані базову стратегію розвитку.

Таблиця 5.15. Визначення базової стратегії розвитку.

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
	Стратегія спеціалізації	Стратегія спеціалізації	Передбачає концентрацію на потребах одного цільового сегменту, без прагнення охопити увесь ринок. Мета тут полягає в задоволенні потреб вибраного цільового сегменту краще, ніж конкуренти	Стратегія спеціалізації

За базову стратегію розвитку було взято стратегію спеціалізації, що передбачає концентрацію на потребах одного цільового сегменту, без прагнення охопити увесь ринок.

Наступним кроком є вибір стратегії конкурентної поведінки (табл. 5.16).

Таблиця 5.16. Визначення базової стратегії конкурентної поведінки.

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
	Ні	Пошук нових споживачів	Так - методи та засоби - схожість систем - побудова алгоритму моделювання роботи ВС	Стратегія заняття конкурентної ніші

За базову стратегію конкурентної поведінки була прийнята стратегія зайняття конкурентної ніші, коли компанія в якості цільового ринку вибирає один або декілька ринкових сегментів малого розміру. Головне завдання компанії при цьому – це постійна турбота про підтримку і розвиток своєї конкурентної переваги, формування лояльності і прихильності споживачів, підтримка вхідних бар'єрів.

На основі вимог споживачів з обраного сегменту до постачальника і продукту, а також в залежності від стратегії розвитку та стратегії конкурентної поведінки розробляємо стратегію позиціонування яка визначається у формування ринкової позиції, за яким споживачі мають ідентифікувати проект.

Таблиця 5.17. Визначення стратегії позиціонування.

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	База бібліотек	Стратегія спеціалізації	- Наукова новизна - Метод отримання знань	- Позиціонування за співвідношенням "ціна – якість"
2	Швидкість та ефективність	Стратегія спеціалізації	Корисність - Примінення в наукових цілях	- Позиціонування за сферою застосування - Стратегія позиціонування за однією ознакою
3	Представлення результатів проектування	Стратегія спеціалізації	- Вид представлення	

Компанія за стратегію розвитку обрала спеціалізацію, і за цільові групи було обрано машино- та приладобудівні підприємства, хоча у них вже є постачальники, але за рахунок нової технології компанія буде забирати клієнтів у конкурентів, і проводити підтримку та реалізовувати розвиток своєї конкурентної переваги.

#### 5.5. Розроблення маркетингової програми стартап-проекту

Під час розроблення маркетингової програми першим кроком є розробка маркетингової концепції товару, який отримає споживач. У таблиці 5.18 підсумовуємо результати аналізу конкурентоспроможності товару.

Таблиця 5.18. Визначення ключових переваг концепції потенційного товару.

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	2	3	4
1	Оболонка	Оболонка є в наявності	Можливе покращення оболонки
2	Інтерпретатор бібліотек	Здійснює швидке та якісне перетворення інформації	+
3	Метод представлення знань	Метод продукцій	Можливе замінення методу в разі необхідності
4	Підхід пошуку інформації необхідних бібліотеках	Прямий ланцюжок міркування	+

Продовження таблиці 5.18

1	2	3	4
5	Рівень мови придбання знань	Використання AnyLogic 7.0	Можливе оновлення до більш сучасної версії
6	Тип набутих знань	Символьний	Необхідно додати, табличний та зчитування рисунків
7	Можливість доповнення та редагування;	В головному кодї програми	Створити власний редактор бібліотек
8	Оновлення бібліотек	В головному кодї програми, за нестачею інформації	Поповнення новими матеріалами
9	Простота використання	Можна використовувати не будучи спеціалістом в даній предметній області	+

За рахунок ключових переваг товару і стратегії диференціації, що передбачає надання товару важливих з точки зору споживача відмінних властивостей за такою ж ціною як і у конкурентів буде розроблено маркетингову програму стартап-проекту.

Таблиця 5.19. Опис трьох рівнів моделі товару.

Рівні товару	Сутність та складові	
I. Товар за задумом	Створення технологічного процесу	
II. Товар у реальному виконанні	Властивості/характеристики	Size
	1. Комп'ютерна програма	15 mb
	2. Оригінальний код програми	
	Якість: стандарти, нормативи, довідники.	
	Пакування: Диск.	
III. Товар із підкріпленням	Марка: AnyLogic +	
	До продажу: доставка, налаштування.	
Після продажу: обслуговування		
За рахунок чого потенційний товар буде захищено від копіювання: за рахунок наявності авторського права.		

В таблиці 5.19 створено три - рівневу модель нашого товару, що включає задум товару та його вигоди, основні характеристики готового товару, спосіб його пакування та захисту від копіювання та плагіату.

Наступним кроком є визначення цінових меж, якими необхідно керуватися при встановленні ціни на потенційний товар, це передбачає аналіз цін товарів конкурентів, та доходів споживачів продукту (табл. 5.20).

Таблиця 5.20. Визначення меж встановлення ціни.

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	500 грн	7000грн	2500грн	1000-3000 грн

В таблиці проаналізовано ринкові ціни на товари аналоги та замінники, а також середній рівень доходів споживачів. За отриманими даними буде встановлена верхня та нижня межа на нашу програму.

Таблиця 5.21. Формування системи збуту.

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Звична купівля з деяким змінами, або модифікована закупівля. Вона передбачає придбання дещо змінених товарів (послуг), або зміну ціни на товар (послугу), або зміну кількості постачання).	Доставка товару покупцю, його встановлення та налаштування.	Канал нульового рівня	Власна система збуту. Виробник безпосередньо продає товар клієнту і використовує три способи прямого продажу : - Торгівля через магазини - Посилкова торгівля - Торгівля в роздріб

Спираючись на специфіку закупівельної поведінки цільових клієнтів було обрано власну систему збуту, коли виробник безпосередньо продає товар клієнту через торгівлю в магазинах, посилками чи в роздріб. Також заглибину каналу збуту було обрано канал нульового рівня, тому що компанія хоче мати тісні контакти із споживачами на обмеженому цільовому каналі.

Таблиця 5.22. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Бути на зв'язку	- електронна пошта - моб. телефон	Проектування технологічних процесів	Зацікавлення	«Думати та вибирати правильні рішення залиш нам»

Висока якість і швидке виготовлення виробів є головною концепцією товару, за рахунок яких він є конкурентоспроможним на ринку.

Спираючись на специфіку закупівельної поведінки цільових клієнтів було обрано власну систему збуту, коли виробник безпосередньо продає товар клієнту.

Ціноутворення відбувається на основі аналізу товарів – аналогів і відбувається під час фінансово-економічного аналізу проекту з залученням експертів.

### Висновки до розділу

Ринкова комерціалізація проекту опирається на наявний попит, динаміку ринку та рентабельність роботи на ринку. Клієнти на даному ринку зазвичай займаються звичною купівлею з деякими змінами, яка передбачає придбання дещо змінених товарів, або зміну ціни на товар, саме на це і розрахований даний стартап-проект, так як за рахунок створеної виробничої системи відбудеться підвищення ефективності та покращення технологічного процесу виготовлення виробів на підприємстві, вона матиме кращі властивості при роботі порівняно із товарами конкурентів.

Спираючись на специфіку закупівельної поведінки цільових клієнтів і розробивши власну систему збуту є хороші перспективи впровадження даного продукту на ринок. Бар'єром входження на ринок є кількість товарів-аналогів, але порівняно із ними дана виробнича система оптимізує роботу, має достатню базу даних і можливість швидкого переналагодження системи за рахунок чого вона може стати конкурентоспроможним на ринку.

Впровадження на ринок розроблена на основі стратегії зайняття конкурентної ніші, коли компанія в якості цільового ринку вибирає один або декілька ринкових сегментів малого розміру.

Отже, подальша імплементація продукту є доцільною за рахунок сильних сторін продукту і наявного попиту на ринку.

## ЗАГАЛЬНІ ВИСНОВКИ ЗА РЕЗУЛЬТАТАМИ МАГІСТЕРСЬКОЇ ДИСЕРТАЦІЇ

1. Виконаний аналіз стану моделювання виробничих систем в приладо- та машинобудуванні, що представлено в множині різноманітних джерел (статті, дисертації, книги тощо), дозволив встановити основні напрямки вдосконалення та покращення роботи ВС де основним засобом їх реалізації є математичне моделювання.

2. Аналіз засобів математичного моделювання складних систем показав, що на інформаційному ринку сьогодні доступна множина програмних засобів, які застосовуються у різних галузях, зокрема в економіці. Перевагою даних систем моделювання є їх гнучкість і можливість охоплювати відразу декілька галузей, зокрема галузь машино – та приладобудування. Встановлено, що серед таких систем, які доцільно використовувати як засіб математичного моделювання в приладобудуванні, треба відзначити ARIS, AnyLogic, Arena, ITHINK, та GPSS.

3. Аналіз сучасних технологій моделювання показав, що для дослідження і аналізу ВС найбільш придатною є технологія імітаційного моделювання, оскільки вона дозволяє створювати схожі за властивостями і структурі моделі виробничих процесів, зміна параметрів і умов в побудованих моделях не вимагає великих часових і фінансових витрат, процес проведення імітаційного експерименту є простим і наочним.

4. Аналіз сучасних засобів імітаційного моделювання показав, що такі засоби опису виробничих процесів не використовують відомі математичні апарати, що не дає їм можливості проводити аналітичне дослідження властивостей моделі та гарантувати можливість побудови різних видів алгоритмів дій. Система AnyLogic, навпаки, дозволяє описувати ВС за допомогою ієрархічних тимчасових кольорових мереж Петрі, що являється

найбільш придатним інструментом для виконання поставлених цілей магістерської роботи.

5. Встановлено, що система AnyLogic є інструментом імітаційного моделювання нового покоління, що розроблений на основі сучасних концепцій в області інформаційних технологій і результатів досліджень в теорії гібридних систем і об'єктно-орієнтованого моделювання, яка дозволяє шляхом анімації і візуального представлення процесів в безперервній моделі динамічної системи врахувати дискретні події, що є характерними для приладобудівного виробництва;

6. Наведений в дисертації приклад роботи динамічних систем на прикладі моделювання процесу виготовлення деталі показав ефективність застосування системи AnyLogic для моделювання складних виробничих систем в приладобудуванні.

7. Створено імітаційну модель технологічного процесу виготовлення деталі та реалізований інтерфейс роботи з системою моделювання AnyLogic, що дозволив виконати комп'ютерне моделювання процесу з різними параметрами отриманої моделі з метою оптимізації завантаження обладнання виробничої дільниці, визначення раціонального складу використовуваного обладнання при різних розмірах партій виготовлення деталей та виявлення «вузьких» місць технологічного процесу.

8. Створено імітаційну модель технологічного процесу складання хвильового редуктора, що дозволила виконати комп'ютерне моделювання процесу складання та визначити завантаження обладнання кожного робочого посту, наявність черги при виконанні операцій та ефективність процесу виготовлення редукторів. Результати виконання моделювання складання 50-ти хвильових редукторів показали на нераціональне використання робочих ресурсів обладнання, що може призвести до зайвих фінансових та часових витрат.

9. Наведена методика використання імітаційного моделювання засобами системи AnyLogic, що включає послідовний аналіз об'єкта моделювання, технологічного процесу її виготовлення, використовуваного обладнання, інструменту та часу виконання операцій, вибір блоків імітаційної моделі на панелі інструментів системи, завдання параметрів моделі та подальше проведення комп'ютерного експерименту з метою оптимізації технологічного процесу дозволяє підвищити ефективність приладобудівного виробництва.

10. Створено стартап-проект за темою дисертаційної роботи, метою якого є визначення змоги розробленого продукту вийти на ринок і працювати з конкурентами, що вже зайняли на ньому своє місце.

11. Впроваджено результати магістерської роботи в навчальних процес в читанні лекцій та проведенні комп'ютерного практикуму при вивченні дисципліни «Моделювання технологічних процесів» для студентів спеціальності «Автоматизація та комп'ютерно-інтегровані технології».

12. Виконано практичну апробацію результатів магістерської роботи на підприємстві СТ «Альянс-5».

## СПИСОК ЛІТЕРАТУРИ

1. Клименко Т.В. Організація діяльності підприємства: Конспект лекцій. – Лозова: 2017. – 146 с.
2. Васильков В.Г. Організація виробництва: Навч. посібник. – Київ.: КНЕУ, 2003. – 524 с.
3. Горелов Д.О. Організація виробництва: Конспект лекцій. – Харків.: ХНАДУ, 2012. – 544 с.
4. Михайличенко О.В. Історія науки і техніки: Навчальний посібник / Михайличенко О.В. [Текст з іл.] – Суми: СумДПУ, 2013. – 346 с.
5. Градов А.П. Стратегия экономического управления предприятием: производственная система как объект стратегического управления: учеб. пособие. СПб. : СГОГУ, 2001.
6. Васильев С.В. Производственный менеджмент: учебно-методическое пособие / С.В. Васильев. – Великий Новгород, 2003. – 99 с.
7. Сеница Л.М. Организация производства: учеб. пособие. – Минск: ИВЦ Минфина, 2006. – 521 с.
8. Тальнов Ю.Н., Нетишина С.И. Учебно-методический комплекс лекционных занятий по курсу «Производственный менеджмент» (в схемах) – Тольятти: Волжский университет им. В.Н. Татищева, 2007г. 143с.
9. Кононова В.Ю. Модернизация производственных систем на российских промышленных предприятиях: современное состояние и перспективы // Российский журнал менеджмента. Том 4, № 4, 2006. С. 119 – 132.
10. Речкалов В. Производственная система на примере TPS [Электронный ресурс] / Владимир Речкалов. – 2012. – Режим доступа до ресурсу: <http://tospeople.com/2012/01/proizvodstvennaya-sistema/>.
11. Гриньова В.М., Салун М.М. Організація виробництва: Підручник. К., 2009. – 582 с.

12. Гнучкі комп'ютеризовані системи: проектування, моделювання і управління: Підручник / Л.С. Ямпольський, П.П. Мельничук, Б.Б. Самотокін, М.М. Поліщук, М.М. Ткач, К.Б. Остапченко, О.І. Лісовиченко.– Житомир: ЖДТУ, 2005.– 680 с.
13. Промышленная робототехника / Л.С. Ямпольский, В.А. Яхимович, Е.Г. Вайсман и др.: Под ред. Л.С. Ямпольского.– К. : Техніка, 1984.–264с.
14. Ямпольський Л.С., Поліщук М.М., Ткач М.М. Елементи робототехнічних систем а модулі ГВС. – К.: Вища шк., 1992. – 431с.
15. Журавлев С.С. Краткий обзор методов и средств имитационного моделирования производственных систем/электронный журнал "Проблемы информатики", 2009, С. 47–53.
16. Шеер А.В. Моделирование бизнес-процессов. – М.: Весть-МетаТехнология, 2000.
17. Шеер А.В. Бизнес-процессы. Основные понятия. Теория. Методы. – М.: Весть-МетаТехнология, 1999.
18. Замятина О.М. Моделирование систем: Учебное пособие. – Томск: Изд-во ТПУ, 2009. – 204 с.
19. ARIS [Электронный ресурс] – Режим доступа до ресурсу: <http://www.kpms.ru/Automatization/ARIS.htm>.
20. Кузнецов Ю.А., Перова В.И Применение пакетов имитационного моделирования для анализа математических моделей экономических систем: Учебно-методический материал по программе повышения квалификации «Применение программных средств в научных исследованиях и в преподавании математики и механики». Нижний Новгород, 2007. 98 с.
21. Динамическое моделирование на PowerSim Studio [Электронный ресурс] – Режим доступа до ресурсу: <http://www.cemi-genkey.ru/technology1.html>.

22. Варжапетян А.Г. Назначение и структура GPSS/H [Электронный ресурс] / А.Г. Варжапетян. – 2004. – Режим доступа до ресурсу: [http://www.uamconsult.com/book\\_379\\_chapter\\_14\\_3.1.\\_Naznachenie\\_i\\_struktura\\_GPSS/H.html](http://www.uamconsult.com/book_379_chapter_14_3.1._Naznachenie_i_struktura_GPSS/H.html).
23. Общая характеристика системы GPSS Word [Электронный ресурс]. – 2015 – Режим доступа до ресурсу: <http://its.kpi.ua/ts/SiteAssets/SitePages/%D0%9D%D0%B0%D0%B2%D1%87%D0%B0%D0%BB%D1%8C%D0%BD%D1%96%20World.ppt>.
24. Simprocess. Don't just see the future. Change it! [Электронный ресурс] – Режим доступа до ресурсу: <http://simprocess.com/>.
25. Фастовский Э.Г. Основы работы с CASE-средством AllFusion Process Modeler [Электронный ресурс] / Э.Г. Фастовский – Режим доступа до ресурсу: <http://khpi-iip.mipk.kharkiv.edu/library/technpnm/labs/lab01.html>.
26. Королько И.В. Компьютерное моделирование бизнес процессов / И.В. Королько. // Проблемы современной экономики : глобальный, национальный и региональный контекст: сборник научных статей. В 2 ч. Ч. 1. – 2013. – №1. – С. 379–385.
27. AnyLogic. Моделирование для обоснованных решений [Электронный ресурс]. – 2017. – Режим доступа до ресурсу: <https://www.anylogic.ru>.
28. Моделирование рынка коммуникаций [Электронный ресурс]. – 2017. – Режим доступа до ресурсу: <https://www.anylogic.ru/anylogic-models-telecom-market-in-argentina/>.
29. Arena – система имитационного моделирования [Электронный ресурс]. – 2016. – Режим доступа до ресурсу: <http://www.interface.ru/sysmod/arena.htm>.
30. Штерензон В.А. Моделирование технологических процессов: конспект лекций / В.А. Штерензон. Екатеринбург: Изд-во Рос. гос. проф.-пед. ун-та, 2010. 66 с.

31. Сергеев А.И. Методология автоматизации ранних этапов проектирования производственных систем в машиностроении : дис. докт. техн. наук : 05.13.12 / Сергеев Александр Иванович – Оренбург, 2016. – 311 с.
32. Сергеев. А.К Особенности подготовки инженерных кадров в области компьютерной интеграции производства! А.И. Сергеев // Материалы Всероссийской научно-методической конференции «Университетский комплекс как региональный центр образования, науки и культуры».; Оренбургский гос. ун-т. – Оренбург: ОГУ. 2012.–С. 173–176.
33. Псигин, Ю.В. Основы математического моделирования производственных процессов : учебное пособие / Ю.В. Псигин ; под ред. Н.И. Веткасова. – Ульяновск : УлГТУ, 2006. – 108 с.
34. Веткасов, Н.И. Применение методов теории графов и линейного программирования для решения производственных и технологических задач : методические указания / Н.И. Веткасов, Ю.В. Псигин. – Ульяновск : УлГТУ, 2001. – 36 с.
35. Псигин, Ю.В. Основы математического моделирования : программа курса и методические указания / Ю.В. Псигин, С. И. Рязанов; под общ. ред. Н.И. Веткасова. – Ульяновск: УлГТУ, 2007. – 40 с.
36. Кириличев Б.В. Моделирование систем: Учебное пособие. – М.: МГИУ, 2009. – 274 с.
37. Норенков И.П. Основы автоматизированного проектирования. – М.: МГТУ им. Н.Э. Баумана, 2000. – 360 с.
38. Карташов Л.Е. Моделирование гибких производственных систем с временным резервированием : дис. канд. техн. наук: 05.13.20 / Севастопольский национальный технический ун-т. – К., 2005. – 233 с.
39. Пидмогильный Н.В. Оптимизация управления производственными системами на региональном уровне: Дис... канд. техн. наук: 05.13.06 / Национальный технический ун-т Украины "Киевский политехнический ин-т". – К., 2000. – 223 с.

40. Приходько Р.А. Стохастические модели календарного технического обслуживания производственных систем : Дис... канд. техн. наук: 01.05.02 / Национальная металлургическая академия Украины. – Д., 2006. – 175 с.
41. Золотарев А.А. Методы оптимизации распределительных процессов. – М.: Инфра-Инженерия, 2014. – 160 с.
42. Лапа М.В. Интеллектуальный метод гибкого проектирования технологических процессов изготовления деталей приборов : Дис... канд. техн. наук: 05.11.14 / Национальный технический ун-т Украины "Киевский политехнический ин-т". – К., 2004. – 181 с.
43. Нгуен В.Д. Модели имитационной аппроксимации экономических систем : дис. канд. техн. наук : 05.13.18 / Нгуен Ван Дык – Иркутск, 2015. – 134 с.
44. Смирнов Д.П. Разработка методик и алгоритмов объектно-ориентированного моделирования автоматизированных производственных логистических процессов на основе Е-сетей : дис. канд. техн. наук : 05.13.06 / Смирнов Дмитрий Петрович – Москва, 2017. – 144 с.
45. Полюдова Г.Р. Ситуационное управление многопродуктовой производственной системой на основе имитационного моделирования : дис. канд. техн. наук : 05.13.10 / Полюдова Гульнара Рустамовна – Уфа, 2006.
46. Соколова И.С. Энтропийно-вероятностное моделирование сложных стохастических систем : дис. канд. физ.-мат. наук : 05.13.18 / Соколова Ирина Сибэгатулловна – Челябинск, 2013. – 116 с.
47. Проничев Н.Д. Имитационное моделирование производственной системы механообрабатывающего цеха / Н.Д. Проничев, В.Г. Смелов, В.В. Кокарева, А.Н. Малыхин // Известия Самарского научного центра Российской академии наук, т. 15, №6(4), 2013. – С. 937–943.
48. Макарова И.В. Применение технологии имитационного моделирования в исследовании сборочного процесса

автомобилестроительного предприятия / Макарова И.В., Хабибулин Р.Г., Беляев Э.И., А.Г. Димеев // Сборник докладов шестой всероссийской научно-практической конференции «Имитационное моделирование. Теория и практика» (ИММОД-2013). Том 2. – Издательство «ФЭН» Академии наук РТ. Казань. 2013. – С. 105–108.

49. Толуев. Ю.И. Имитационная модель производственной линии на базе сложной конвейерной системы / Ю.И. Толуев, Т.П. Змановская // Автоматизация в промышленности. – № 7. 2013. – С. 37– 41.

50. Борщев А.В. Инвентаризация языков имитационного моделирования для бизнес-приложений / А.В. Борщев // Автоматизация в промышленности. – №7, 2014. – С. 43–48.

51. Автоматизация производственных процессов в машиностроении : учеб. / под ред. Н.М. Капустина. – 2-е изд., стереотип. – М.: Высш. шк., 2007. – 415 с.

52. Базров Б.М. Основы технологии машиностроения: учебник / Б.М. Базров. – М.: Машиностроение, 2005. – 736 с.

53. Волчкевич Л.И. Автоматизация производственных процессов: учеб. пособие / Л.И. Волчкевич. – М.: Машиностроение, 2005. – 380 с.

54. Капустин Н.М. Автоматизация машиностроения: учебник / Н.М. Капустин, Н.П. Дьяконова, П.М. Кузнецов; под ред. Н.М. Капустина. – 3-е изд., стер. – М.: Высш. шк., 2007. – 223 с.

55. Рогов В.А. Средства автоматизации производственных систем машиностроения /В.А. Рогов, А.Д. Чудаков. – М.: Высшая школа, 2005.

56. Шишмарев В.Ю. Автоматизация производственных процессов в машиностроении: учеб. / В.Ю. Шишмарев. – М.: ИЦ "Академия", 2007. – 368 с.

57. Самойлова Е.М., Виноградов М.В. интерактивная среда моделирования Electronics Workbench. //Автоматизация и управление в

машино- и приборостроении. Межвуз. научн.сб. г.Саратов. СГТУ, 2000. С.122-124.

58. Иванов И.Н. Организация производства на промышленных предприятиях : учеб. пособие для вузов – Москва : ИНФРА-М, 2009. – 352 с

59. Афитов Э.А. Планирование на предприятии. – Минск : Высшая школа, 2001. – 284 с.

60. Ребрин Ю.И. Основы экономики и управления производством. – Таганрог: ТРТУ, 2000. – 145 с.

61. Гулятьев А.В. Визуальное моделирование в среде MATLAB: учебный курс. – Спб.: Питер, 2000. – 432 с.

62. Собкин Б.Л. Автоматизация проектирования аналого-цифровых приборов на микропроцессорах. – М.: Машиностроение, 1986. – 128 с.

63. Ту Ю.Т. Современная теория управления. – М.: Машиностроение, 1971. – 472 с.

64. Джонс Дж. К. Методы проектирования. – М.: Мир, 1986. – С. 325-326.

65. Васильев К.К., Служивый М. Н. Математическое моделирование систем связи: учебное пособие. – Ульяновск: УлГТУ, 2008. – 170 с.

66. Колесов Ю.Б. Объектно-ориентированное моделирование сложных динамических систем. – СПб.: СПбГПУ, 2004. – 239 с.

67. Духанов А.В. Имитационное моделирование сложных систем: курс лекций. – Владимир: ВлГУ, 2010. – 115 с.

68. Родионов И.Б. Теория систем и системный анализ: курс лекций [Электронный ресурс]. Режим доступа до ресурсу: <http://victor-safronov.ru/systems-analysis/lectures/rodionov.html>.

69. Лычкина Н.Н. Имитационное моделирование экономических процессов: учебное пособие для слушателей программы eMBA. – М.: Академия АйТи, 2005. – 164 с.

70. Gordon G. A general purpose systems simulation program // Proceedings of the December 12-14, 1961, eastern joint computer conference: computers - key to total systems control. 1961. P. 87-104.

71. Schriber T. The Nature and Role of Simulation in the Design of Manufacturing System // Simulation in Computer Integrated Manufacturing. – The Society for Computer Simulation, 1987. P. 5-18.

72. Schulze T., Lorenz P., Hinz V. Simulation und Visualisierung 2000. SCS, Ghent, 2000.

73. Карпухин И.Н. Исследование и реализация систем дискретно-событийного имитационного моделирования на основе графовых моделей: дис. канд. техн. наук: 05.13.11 / Карпухин Илья Николаевич. – М., 2013. – 321 с. 26.

74. Подкрытов Д.И. Разработка отказоустойчивых мульти-агентных средств имитационного моделирования систем с дискретными событиями: дис. канд. техн. наук: 05.13.11 / Подкрытов Дмитрий Игоревич. – Новосибирск, 2012. – 114 с.

75. Nutt G.J. The formulation and application of evaluation nets, Ph.D. dissertation. University of Washington. 1972.

76. Engelfriet J., Rozenberg G. Elementary net systems. In lectures on Petri nets I: basic models, ser. lecture notes in computer science // Springer Berlin Heidelberg. 1998. Vol. 1491. P. 12-121.

77. Илюшечкина Л.В., Костин А.Е. Модифицированные E-сети для исследования систем распределённой обработки информации // Автоматика и вычислительная техника. 1988. №6. – С. 27-35.

78. Илюшечкина Л.В. Разработка средств моделирования для исследования систем распределённой обработки информации: дис. канд. техн. наук: 05.13.01 / Илюшечкина Людмила Валентиновна. – М., 2002. – 240 с.

79. Костин А.Е. Модели и алгоритмы организации распределенной обработки данных в информационных системах: докт. дис. М.: МИЭТ, 1989. – 221 с.
80. Kostin A. E. Microsim – E-net tool for modeling networks and distributed data processing systems // Proc. of the first symposium on computer networks. – Istanbul, 1996. P. 34-46.
81. Гагарина Л.Г., Илюшечкина Л.В., Костин А.Е. Исследование локальной вычислительной сети Ethernet с использованием модифицированных E-сетей // Известия вузов. Электроника. – М: МИЭТ, 2002. № 2. – С. 72-78.
82. Гагарина Л.Г., Чудайкина С.А. Логистические системы и сети Петри // Оборонный комплекс. – М.: ВИМИ. 2004. №3. – С. 24-27.
83. Илюшечкина Л.В. Разработка средств моделирования для исследования систем распределённой обработки информации: дис. канд. техн. наук: 05.13.01 / Илюшечкина Людмила Валентиновна. – М., 2002. – 240 с.
84. Костина С.А. Моделирование логистических процессов в распределённых производственных системах сетями Петри: дис. канд. техн. наук: 05.13.06 / Костина Светлана Александровна. – М., 2005. – 212 с.
85. Rzecki K. Using the Evaluation Nets Modeling Tool Concept as an Enhancement of the Petri Net Tool // Proceedings of the Fedcsis. – Krakow, 2013. P. 1007-1012.
86. Ефремов А.А. E-сетевое моделирование надежности последовательно-параллельных технических систем с восстановлением // Известия ТПУ. – Томск, 2006. №7. – С. 97–101.
87. Казимир В.В., Серая А.А. Модели верификации планов ликвидации аварий на угольных шахтах // Математические машины и системы. – Киев: ИПММС НАНУ, 2012. №1. – С. 129-138.

88. Harmon P., Wolf C. Business Process Modeling Survey [Electronic resource] // BPTrends [Official website]. URL: [http://www.bptrends.com/bpt/wp-content/surveys/Process\\_Modeling\\_Survey-Dec\\_11\\_FINAL.pdf](http://www.bptrends.com/bpt/wp-content/surveys/Process_Modeling_Survey-Dec_11_FINAL.pdf).

89. Борщев А.В. Имитационное моделирование: состояние области на 2015 год, тенденции и прогноз [Электронный ресурс]. Режим доступа до ресурсу: <http://simulation.su/uploads/files/default/2015-immod-14-22.pdf>.

90. Маликов, Р.Ф. Практикум по имитационному моделированию сложных систем в среде AnyLogic 6 [Текст]: учеб. пособие / Р.Ф. Маликов. – Уфа: Изд-во БГПУ, 2013. – 296 с.

91. Боев В.Д., Кирик Д.И., Сыпченко Р.П. Компьютерное моделирование: Пособие для курсового и дипломного проектирования. – СПб.: ВАС, 2011. – 348 с.

92. Боев В.Д. Компьютерное моделирование: Пособие для практических занятий, курсового и дипломного проектирования в AnyLogic7: — СПб.: ВАС, 2014. — 432 с.

# ДОДАТКИ

Справ. №		Перв. примен.		Формат	Зона	Поз.	Обозначение	Наименование	Кол.	Примечание	
								<u>Документация</u>			
				A1				Хвильовий редуктор з кулачковим генератором хвиль			
								<u>Сборочные единицы</u>			
						1		Вихідний вал СК	1		
						2		Вхідний вал СК	1		
								<u>Детали</u>			
						6		Корпус редуктора	1		
						10		Кришка вхідного валу	1		
						16		Кришка вентелятора	1		
								<u>Стандартные изделия</u>			
						7		Прокладка ГОСТ 15180-86	1		
						9		Стопорне кільце ГОСТ 13942-86	1		
						11		Гвинт ГОСТ 7798-70	6		
Изм. № подл.		Подп. и дата		Взам. инв. №		Инв. № дубл.		Подп. и дата			
Изм. Лист		№ докум.		Подп.		Дата					
Разраб.		Нестеренко А.О.								Лит.	
Пров.		Вислоух С.П.								Лист	
										Листов	
Н.контр.										1	
Утв.										2	
Редуктор хвильовий з кулачковим генератором хвиль											



**005** Комплектувальна.

Комплектувати деталі відповідно до складального креслення – 2.5хв.

**010** Підготовча.

1. Розконсервувати деталі. - 2хв.
2. Промити деталі спиртом. – 2хв.
3. Просушити деталі стисненим повітрям. – 2хв.
4. Контроль якості деталей. – 9 хв.

**015** Складальна операція вузла Вхідний вал СК.

1. Пресувати кулачок поз.2 на вхідний вал поз.1. – 0.9хв.
2. Вставити втулку поз.3 на вхідний вал поз.1. - 0.1хв.
3. Пресувати підшипник поз.4 на вхідний вал поз.1. – 0.9хв.
4. Змастити підшипник маслом Літол-24 ГОСТ 21150-87. - 0.1хв.
5. Пресувати стопорне кільце поз.5 на вхідний вал поз.1. – 0.9хв.
6. Пресувати підшипник поз.8 на кулачок поз.2. – 0.9хв.
7. Загвинтити диск поз.6 на кулачок поз.2 гвинтом поз.7 (8 шт.) - 1.2хв.
8. Змастити підшипник маслом Літол-24 ГОСТ 21150-87. - 0.1хв.
9. Пресувати стопорне кільце поз.9 на вхідний вал поз.1. – 0.9хв.
10. Пресувати підшипник поз.10 на вхідний вал поз.1. – 0.9хв.
11. Змастити підшипник маслом Літол-24 ГОСТ 21150-87. - 0.1хв.
12. Пресувати стопорне кільце поз.11 на вхідний вал поз.1. – 0.9хв.
13. Пресувати гнучке зубчасте колесо поз.12 на підшипник поз.8. - 0.9хв.
14. Контролювати візуально правильність складання. – 0.5хв.

**020** Складальна операція вузла Вихідний вал СК.

1. Пресувати підшипник поз.2 на вихідний вал поз.1. – 0.9хв.
2. Змастити підшипник маслом Літол-24 ГОСТ 21150-87. - 0.1хв.
3. Пресувати втулку поз.3 на вихідний вал поз.1. – 0.9хв.
4. Пресувати стопорне кільце поз.4 на втулку поз.3. – 0.9хв.

5. Пресувати підшипник поз.5 на вихідний вал поз.1. – 0.9хв.
6. Змастити підшипник маслом Літол-24 ГОСТ 21150-87. - 0.1хв.
7. Пресувати стопорне кільце поз.6 на вихідний вал поз.1. – 0.9хв.
8. Пресувати циліндр вихідного валу поз.7 на вихідний вал поз.1. – 0.9хв.
9. Пресувати штифт поз.8 в циліндр вихідного валу поз.7. – 0.9хв.
10. Загвинтити жорстке зубчасте колесо поз.9 на циліндр вихідного валу поз.7 гвинтом поз.10 (4 шт.) гайкою поз.11 (4 шт.) та шайбою-пружиною поз.12 (4 шт.).
11. Пресувати кришку вихідного валу поз.13 на підшипник поз.2 та підшипник поз.5. – 0.9хв.
12. Загвинтити кришку вихідного валу поз.13 до стопорного кільця поз.4 через циліндр вихідного валу поз.7 гвинтом поз.14(3 шт) -1.2хв. .
14. Пресувати сальник поз.16 в кришку вихідного валу поз.13. – 0.9хв.
15. Контролювати візуально правильність складання. - 0.5 хв.

**025** Загальне складання.

1. Вставити прокладку поз.7 в корпус редуктора поз.6 – 0.2хв.
2. Вставити вінець зубчастий поз.8 в корпус редуктора поз.6 – 0.2хв.
3. Вставити прокладку поз.9 в корпус редуктора поз.6 – 0.2хв.
4. Вставити вхідний вал СК поз.У1 в корпус редуктора поз.6. – 2.2хв.
5. Загвинтити кришку вихідного валу поз.10 гвинтом поз.11(6 шт) та пружиною-шайбою поз.12(6 шт) до корпуса редуктора поз.6. – 1.2хв.
6. Пресувати сальник поз.13 в кришку вхідного валу поз.10. – 0.9хв.
7. Загвинтити крильчатку поз.14 гвинтом поз.14. – 0.2хв.
8. Загвинтити кришку вентилятора поз.16 гвинтом поз.17 до корпуса поз.6. – 0.8хв.

9. Вставити прокладку поз.18 в корпус редуктора поз.18. – 0.2хв.
10. Вставити вихідний вал СК поз.У2 в корпус редуктора поз.6. – 1.4хв.
11. Загвинтити вихідний вал СК поз.У2 в корпус редуктора поз.6 гвинтом поз.6 з пружиною-шайбою поз.15. – 1.2хв.
12. Встановити рим болт поз.20 в корпус редуктора поз.6. – 0.2хв.
13. Встановити пробку заливну поз.21 в корпус редуктора поз.6. – 0.2хв.
14. Встановити пробку контролю масла поз.22 в корпус редуктора поз.6. – 0.2хв.
15. Встановити пробку зливну поз.23 в корпус редуктора поз.6. – 0.2хв.
16. Контролювати візуально правильність складання. – 0.5хв.

**030** Контрольна.

1. Контролювати роботу під навантаженням.
2. Контролювати вібрації редуктора.

**035** Випробування.

Випробувати на надійність довготривалому використанні.

ДОДАТОК А

**ДОДАТОК Б**

ДОДАТОК В

ДОДАТОК Г

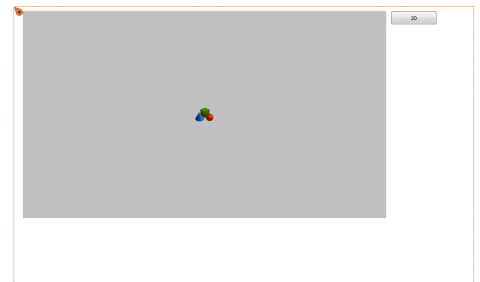
ДОДАТОК Д

## Модель: Изготовление корпуса

null	null
Основная	
Единицы модельного времени	минуты
Численные методы	
Метод решения дифф.ур-й	Эйлер
Метод решения алгебр. ур-й	Измененный Ньютон
Метод решения смешанных ур-й	RK45+Newton
Абсолютная точность	1.0E-5
Временная точность	1.0E-5
Относительная точность	1.0E-5
Фиксированный шаг по времени	0.001
Специфические	
Имя Java пакета	izgotovlenie_detali
Имя файла	D:\Документы\AnyLogic\izgotovlenie_detali2 2\izgotovlenie_detali2\izgotovlenie_detali2.alp

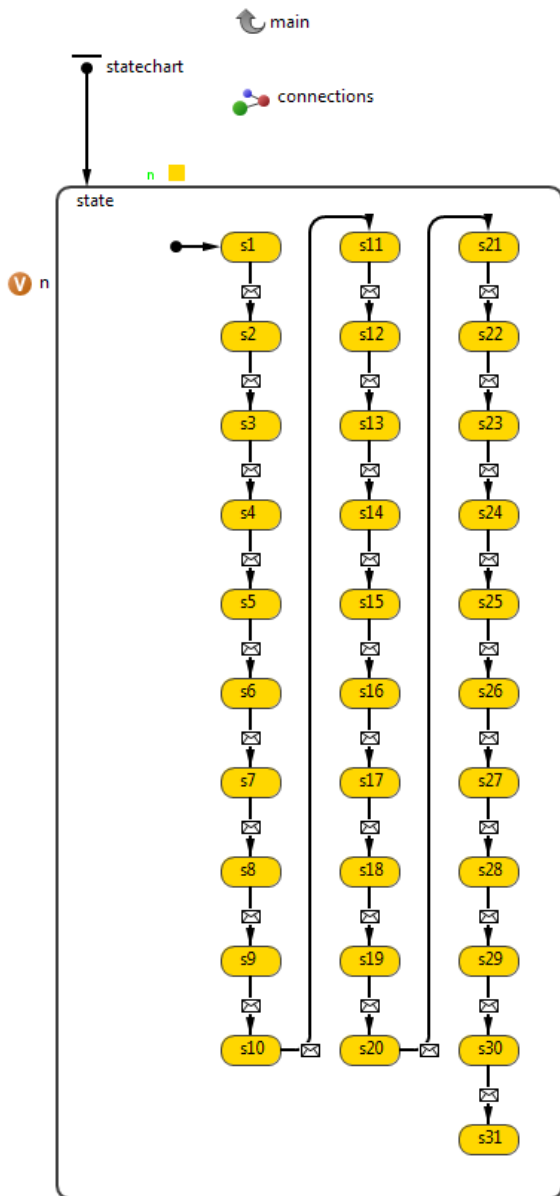
## Тип агента: Main

null	null
Действия агента	
Действие при запуске	v1.navigateTo();
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false
Пространство и сеть	
Тип пространства	Непрерывное
Ширина (динамич.)	500
Высота (динамич.)	500
z высота (динамич.)	0
Тип расположения	Задается пользователем
Применить расположение при запуске	true
Тип сети	Задается пользователем
Создать сеть при запуске	true
Выполнять шаги	false
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false



## Тип агента: detal

null	null
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 : MPS)
Поворачивать анимацию согласно направлению движения	false
Пространство и сеть	
Тип пространства	Непрерывное
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false



## Простой эксперимент: Simulation

null	
Основная	
Максимальный размер памяти	512
Тип агента	Main
Модельное время	
Режим выполнения	Реальное время со скоростью
Реальное время	1.0
Остановить	Нет
Начальное время	0.0
Начальная дата	Wed Mar 21 00:00:00 GMT 2018
Случайность	
Тип генератора случайных чисел	Фиксированное начальное число (воспроизводимые "прогоны")
Начальное число генератора случ. чисел	1
Упорядочение одновременных	LIFO (в обратном порядке)

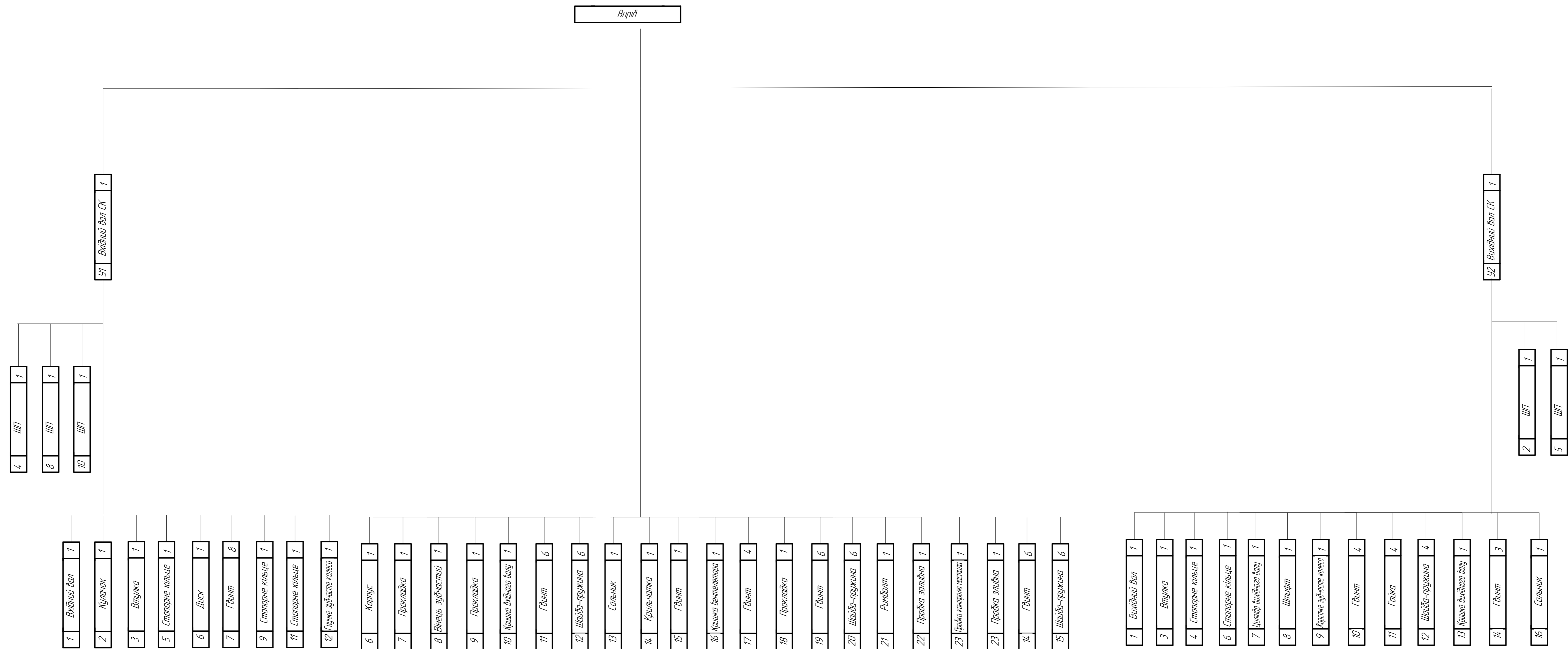
null	
событий	
Окно	
Заголовок	izgotovlenie_detali : Simulation
Разрешить изменение масштаба и перетаскивание	true
Во весь экран	false
Подтверждение закрытия	false
Специфические	
Включить сглаживание	true
Включить расширенную анимацию элементов модели	true
Адаптивная частота обновлений	true
Выбор приоритета	1 : 2
Загрузить корневой объект из файла состояния	false

## izgotovlenie\_detali

Запустить

## База данных: База данных

null	
Журнал	
Logging	true



Изм.	Кол.	Лист	№рек.	Подп.	Дата	<b>Схема структурного склада редуктора</b>	Стандия	Масса	Масштаб
Разраб.	Исполн.	Провер.	Вспомог. СК				Лист	Листов	1
Т.контр.							<b>ПБФ, ПБ-61М</b>		
Н.контр.									
Утв.									

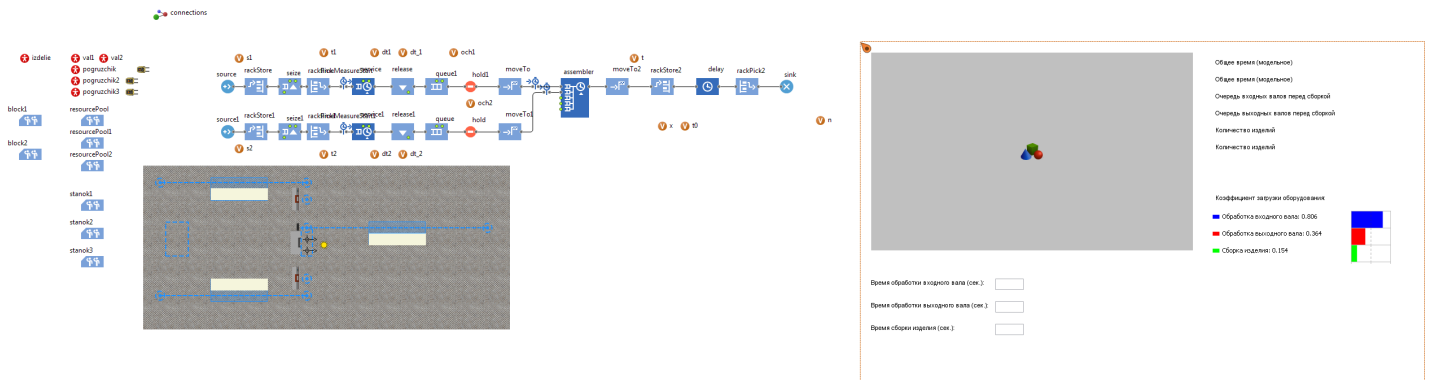


## Модель: Сборка волнового редуктора

null	null
Основная	
Единицы модельного времени	минуты
Численные методы	
Метод решения дифф.ур-й	Эйлер
Метод решения алгебр. ур-й	Измененный Ньютон
Метод решения смешанных ур-й	RK45+Newton
Абсолютная точность	1.0E-5
Временная точность	1.0E-5
Относительная точность	1.0E-5
Фиксированный шаг по времени	0.001
Специфические	
Имя Java пакета	model7
Имя файла	D:\Документы\AnyLogic\sborka_vala1\sborka_vala1.original.alp

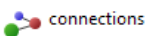
## Тип агента: Main

null	null
Действия агента	
Действие при запуске	source.inject(5); source1.inject(5); v1.navigateTo();
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 / getScale().pixelsPerUnit(METER) / 60 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false
Пространство и сеть	
Тип пространства	Непрерывное
Ширина (динамич.)	500
Высота (динамич.)	500
z высота (динамич.)	0
Тип расположения	Задается пользователем
Применить расположение при запуске	true
Тип сети	Задается пользователем
Создать сеть при запуске	true
Выполнять шаги	false
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false



## Тип агента: val1

null	null
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 / getScale().pixelsPerUnit(METER) / 60 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false
Пространство и сеть	
Тип пространства	Непрерывное
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false




## Тип агента: pogruzchik

null	null
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 / getScale().pixelsPerUnit(METER) / 60 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false

null	
Пространство и сеть	
Тип пространства	Непрерывное
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false


 main

 connections


## Тип агента: pogruchik2

null	
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 / getScale().pixelsPerUnit(METER) / 60 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false
Пространство и сеть	
Тип пространства	Непрерывное
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false

 main

 connections

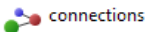

## Тип агента: val2

null	
null	

null	
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 / getScale().pixelsPerUnit(METER) / 60 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false
Пространство и сеть	
Тип пространства	Непрерывное
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false



main

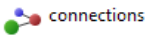


connections



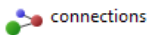
## Тип агента: izdelie

null	
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 / getScale().pixelsPerUnit(METER) / 60 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false
Пространство и сеть	
Тип пространства	Непрерывное
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false



## Тип агента: pogruzchik3

null	null
Агент в диаграмме процесса	
Использовать в диаграммах процесса как	Агент
Движение	
Скорость	(10 / getScale()).pixelsPerUnit(METER) / 60 : MPS)
Поворачивать анимацию согласно направлению движения	true
Также наклонять и вертикально	false
Пространство и сеть	
Тип пространства	Непрерывное
Java для экспертов	
Параметр класса	false
Специфические	
Logging	true
Автоматич. создание наборов данных	true
AOC_DATASETS_UPDATE_TIME_PROPERTIES	- Recurring Event Properties
Ограничить количество хранимых значений	false



## Простой эксперимент: Simulation

null	null
Основная	
Максимальный размер памяти	256
Тип агента	Main
Модельное время	
Режим выполнения	Реальное время со скоростью
Реальное время	1.0
Остановить	Нет
Начальное время	0.0
Начальная дата	Mon Dec 04 00:00:00 GMT 2017
Случайность	
Тип генератора случайных чисел	Случайное начальное число (уникальные "прогоны")
Упорядочение одновременных событий	LIFO (в обратном порядке)

null	
Окно	
Заголовок	Model7 : Simulation
Разрешить изменение масштаба и перетаскивание	true
Во весь экран	false
Подтверждение закрытия	false
Специфические	
Включить сглаживание	true
Включить расширенную анимацию элементов модели	true
Адаптивная частота обновлений	true
Выбор приоритета	1 : 2
Загрузить корневой объект из файла состояния	false

## Model7

Запустить