

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІМ. Ю.М. ПОТЕБНІ
ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ

Кафедра автоматизованого управління технологічними процесами
(повна назва кафедри)

Кваліфікаційна робота / проект

другий магістерський
(рівень вищої освіти)

на тему Розробка системи підтримки прийняття рішення
по управлінню процесом екструзії полімерних матеріалів

Виконав: студент 2 курсу, групи 8.1510
спеціальності

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

освітньої програми

Автоматизація та комп'ютерно-інтегровані технології
(код і назва освітньої програми)

спеціалізації

(код і назва спеціалізації)

П.О. Крохмаль

(ініціали та прізвище)

Керівник к.т.н., доцент, Н.О. Міняйло
(посада, вчене звання, науковий ступінь, підпис, ініціали та прізвище)

Рецензент

ТОВ "ДПСВ Амотек Запоріжжя", зам. Директора
з адм.студ. Крест О.І.

(посада, вчене звання, науковий ступінь, підпис, ініціали та прізвище)

Запоріжжя

2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІМ. Ю.М. ПОТЕБНИ
ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
(повне найменування закладу вищої освіти)

Кафедра автоматизованого управління технологічними процесами

Рівень вищої освіти другий магістерський
(другий (магістерський) рівень)

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

Спеціалізація _____
(шифр і назва)

Освітньо-професійна програма Автоматизація та комп'ютерно-інтегровані технології
(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

М.Ю. Пазюк

"09" квітня 2021 року

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Крохмало Павлу Александровичу
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи магістра Розробка системи підтримки прийняття рішень по управлінню процесом екструзії полімерних матеріалів

керівник кваліфікаційної роботи магістра Мінсько Наталія Александрівна, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від "30" квітня 2021 року № 974-С

2. Строк подання здобувачем кваліфікаційної роботи магістра 15 листопада 2021

3. Вихідні дані кваліфікаційної роботи магістра Аналіз технічної літератури за темою роботи. Технічна документація. Результати досліджень на екструзійній установці.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Технологія екструзії полімерів. Побудова математичної моделі СППР. Розробка технічного забезпечення. Розробка програмного забезпечення. Практична реалізація.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

СППР по управлінню екструзією полімерних матеріалів. Математична модель СППР: кількість темати та потужність. Функціональний JavaScript модуль СППР. Дослідження роботи СППР. Архітектура Web-додатку. Моноструктура екструзійної лінії. Монітор реального часу сист. лінії. Графічний інтерфейс СППР: кількість темати та потужність. Висновки.

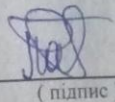
6. Консультанти розділів кваліфікаційної роботи магістра		Підпис, дата	
Розділ	Прізвище, ініціали та посада консультанта	завдання видав	завдання прийняв
1	Мінзіню Н.О., к.т.н., доцент	30.08.21	15.07.21
2	Мінзіню Н.О., к.т.н., доцент	10.07.21	01.08.21
3	Мінзіню Н.О., к.т.н., доцент	23.08.21	05.09.21
4	Мінзіню Н.О., к.т.н., доцент	04.09.21	01.10.21
5	Мінзіню Н.О., к.т.н., доцент	05.10.21	05.10.21

7. Дата видачі завдання 30 червня 2021

КАЛЕНДАРНИЙ ПЛАН

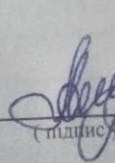
№ з/п	Назва етапів кваліфікаційної роботи магістра	Строк виконання етапів кваліфікаційної роботи магістра	Примітка
1	Вивчення особливостей тех. екструзії полімерів	30.08.21-15.07.21	виконано
2	Розрахунок та дослідження енергетичного балансу та потужності екструдера	10.07.21-22.07.21	виконано
3	Побудова математичної моделі СППР	21.07.21-01.08.21	виконано
4	Розробка технічного забезпечення розробленого малогабаритного екструдера	21.07.21-01.08.21	виконано
5	Розробка програмного забезпечення Web-давачів системи управління екструзійної лінії	23.08.21-05.09.21	виконано
6	Розробка програмного забезпечення СППР	04.09.21-16.09.21	виконано
7	Інтеграція СППР у розроблений Web-давач	15.09.21-24.09.21	виконано
8	Тестування роботи розробленої системи	23.09.21-02.10.21	виконано
9	Хронометраж роботи розробленої системи	05.10.21-25.10.21	виконано
	Хронометраж корекції, підготовка доповіді.	01.11.21-08.11.21	виконано

Здобувач вищої освіти


(підпис)

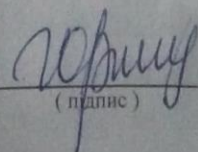
Крохмаль П.О.
(прізвище та ініціали)

Керівник кваліфікаційної роботи магістра


(підпис)

Мінзіню Н.О.
(прізвище та ініціали)

Нормоконтроль пройдено


(підпис)

Обшишкова І.Т.
(прізвище та ініціали)

РЕФЕРАТ

На пояснювальну записку кваліфікаційної роботи магістра на тему: “Розробка системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів”, що викладена на 132 сторінках, містить 70 рисунків, 20 таблиць, 35 найменувань переліку посилань, 4 додатка на 5 аркушах.

Метою роботи є розробка системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів шляхом дослідження впливу виробничих параметрів (параметри екструдера та властивості полімерної сировини) на режим роботи екструзійної лінії.

Об’єктом дослідження є автоматизована система.

Предметом дослідження є система підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів.

Наукова новизна полягає у створенні системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів та її інтеграції у прогресивний Web-додаток задля забезпечення ефективного перелаштування виробництва під нові виробничі вимоги.

Ключові слова: екструзія, тепловий баланс, архітектура, Web, Web-сокет, база даних, фреймворк, бібліотека, монітор реального часу.

ЗМІСТ

Вступ.....	8
Розділ 1 Технологія екструзії полімерів.....	13
1.1 Аналіз літературних джерел.....	13
1.1.1 Технологія екструзії полімерів.....	13
1.1.2 Теорія і практика екструзії полімерів.....	15
1.1.3 Плавлення полімерів в екструдерах.....	17
1.2 Екструзійна лінія як технологічний об'єкт.....	19
1.2.1 Одношнековий екструдер.....	20
1.2.2 Протягувач.....	21
1.2.3 Намотувач.....	22
1.3 Особливості технологічного процесу екструзії.....	23
1.3.1 Процеси, що відбуваються в момент екструзії.....	23
1.3.2 Основні параметри екструзійного процесу.....	28
1.3.3 Матеріали, що використовуються для екструзії.....	29
Розділ 2 Побудова математичної моделі СППР.....	31
2.1 Методи і технології синтезу математичних моделей процесів екструзії в гнучких виробництвах полімерних матеріалів.....	31
2.2 Розрахунок теплового балансу.....	35
2.2.1 Аналіз впливу попереднього підігріву полімеру при подачі в екструдер.....	40
2.2.2 Аналіз впливу частоти обертання шнека.....	41
2.2.3 Аналіз впливу теплоємності полімеру.....	42
2.3 Розрахунок потужності електродвигуна екструдера.....	44

2.3.1 Аналіз впливу зазору між шнеком і стінкою циліндра на загальну потужність електродвигуна екструдера	46
2.4 Структура математичної моделі СППР	48
Розділ 3 Розробка технічного забезпечення.....	50
3.1 Схема інформаційних та матеріальних потоків	50
3.2 Перелік застосованих технічних засобів автоматизації.....	52
3.2.1 Первинні перетворювачі	52
3.2.2 Вторинні прилади	54
3.2.3 Контролер Arduino MEGA	55
3.2.4 Виконавчі механізми	56
3.2.5 Інші технічні засоби	58
3.3 Принципова електрична схема управління.....	60
Розділ 4 Розробка програмного забезпечення	62
4.1 Сучасні програмні засоби для розробки Web-додатків	62
4.2.1 Застосування React як платформи для СППР	62
4.2.2 Централізоване сховище для зберігання даних Redux.....	65
4.2.3 Забезпечення нерозривного з'єднання між клієнтом і сервером за допомогою протоколу WebSocket.....	68
4.2.4 Інтеграція динамічних графіків за допомогою Recharts	68
4.2.5 Застосування графічних компонентів бібліотеки Material UI.....	69
4.2.6 Розгортання Web-сервера за допомогою Express.js	70
4.2.7 Реляційна база даних PostgreSQL	70
4.3 Архітектура розробленого Web-додатку.....	72
4.4 Розробка серверної частини	73
4.5 Розробка клієнтської частини.....	77

4.6 Розробка функціонального модуля системи підтримки прийняття рішень	91
Розділ 5 Практична реалізація.....	96
5.1 Мнемосхема екструзійної лінії.....	96
5.2 Екран монітору реального часу екструзійної лінії.....	107
5.3 Вікно оптимізації СППР	115
5.3.1 Вкладка “Кількість теплоти”	115
5.3.2 Вкладка “Потужність”.....	118
Висновки.....	122
Список використаних джерел.....	124
ДОДАТОК А – Принципова електрична схема управління.....	128
ДОДАТОК Б – Вікно мнемосхеми екструзійної лінії Web-додатку СППР..	129
ДОДАТОК В – Вікно монітору реального часу Web-додатку СППР	130
ДОДАТОК Г – Графічний інтерфейс СППР	131

ВСТУП

Екструзія – це найбільш важлива і масова технологія в полімерній промисловості. На Україні екструзія широко використовується для виробництва плівок, листів, труб, профілів і т.д., а також для спеціальних технологічних операцій, таких як компаундування, змішування та гранулювання.

Головна складність в управлінні технологічним процесом екструзії – неможливість процесу, тобто система управління технологічним процесом екструзії не в змозі вплинути на кінцевий продукт, що вийшов з формуючої фільери агрегату. Враховуючи цей фактор, удосконалення алгоритмів управління технологічним процесом екструзії займає особливе місце для покращення техніко-економічних показників виробництва полімерної продукції.

В умовах конкурентної боротьби на ринку збуту, виробники змушені постійно перестроювати виробничу лінію на нову продуктивність та марку полімерного виробу, що відрізняється від попередньої марки як типом полімеру та його фізико-хімічними властивостями, так і новими вимогами до кінцевого продукту. Ці обставини зумовлюють часту зміну як режиму функціонування та конфігурації технічних засобів автоматизації, так і зміну налаштувань системи управління технологічним процесом, щоб забезпечити відповідну ефективність виробництва та задовільну якість кінцевого продукту.

Вимоги до роботи екструзійної лінії включають в себе:

- забезпечення заданої продуктивності;
- забезпечення безперервної та синхронної роботи всіх складових агрегатів лінії (екструдер, намотувач, протягувач);
- забезпечення відповідної фізико-хімічної якості кінцевої продукції;
- ефективний перехід на новий тип сировини та вид продукції.

Фактор незворотності процесу та неможливість одразу передбачити кінцевий результат екструзії змушує виробників витрачати час та коштів на попереднє моделювання технологічного процесу з використанням нового типу пластику та проводити ретельні експерименти з метою попередження випуску бракованої продукції.

Система підтримки прийняття рішення (СППР) – комп'ютеризована система, яка через збирання та аналіз інформації може впливати на процес ухвалення рішення з управління технологічним процесом й у розрізі екструзійного виробництва, підвищувати його ефективність.

В даній кваліфікаційній роботі магістра представлено рішення щодо питань підвищення ефективності перелаштування екструзійної лінії на нові виробничі вимоги за допомогою розробки СППР по управлінню процесом екструзії полімерних матеріалів.

Мета роботи: розробити систему підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів шляхом дослідження впливу виробничих параметрів (параметри екструдера та властивості полімерної сировини) на режим роботи екструзійної лінії.

Об'єкт дослідження: технологічний процес екструзії полімерних матеріалів.

Предмет дослідження: система підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів.

Завдання роботи:

- провести аналіз технології виробництва полімерного виробу – філаменту, у складі екструзійної лінії;
- на базі енергетичного балансу та потужності екструдера побудувати математичну модель для подальшого використання у складі СППР;
- розробити програмний Web-додаток, що складається з клієнтської та серверної частини, у складі клієнтської частини реалізує монітор реального

часу оператора екструзійного виробництва та інтегровану СППР по управлінню процесом екструзії полімерних матеріалів.

Наукова новизна:

1. На основі енергетичного балансу та потужності екструдера встановлено взаємозв'язок між попереднім підігрівом, теплоємністю полімерної сировини, частотою обертання шнека та загальної кількості теплоти, котру необхідно підвести з боку нагрівачів, щоб забезпечувати ефективний прогрів полімерної пробки, а також зазору між шнеком і циліндром на загальну потужність електродвигуна екструдера з метою підвищення ефективності просування полімерної пробки вздовж гвинтового каналу шнека.

2. Засобами прогресивних Web-технологій запропоновано реалізацію програмного додатку, що виконує функції та вирішує задачі класичних SCADA-систем, але має значні переваги завдяки гнучкості в процесі розробки та подальшої підтримки, а також багатоплатформеності.

3. Вперше для системи управління процесом екструзії запропоновано імплементацію системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів для забезпечення ефективного переналаштування виробництва під нові виробничі вимоги.

Практичне значення:

1. На основі енергетичного балансу та потужності екструдера, розроблено математичну модель СППР по управлінню процесом екструзії полімерних матеріалів.

2. Засобами високорівневої мови програмування JavaScript математичну модель СППР у вигляді функціонального модуля інтерпретовано в програмне середовище JavaScript.

3. В результаті застосування прогресивних Web-технологій, таких як React, Redux, WebSocket, MaterialUI та Recharts для клієнтської частини, Express.js, WebSocket, COM та PostgreSQL для серверної частини, розроблено Web-додаток, який вирішує задачі дистанційного диспетчерського

управління екструзійним виробництвом та дає змогу гнучко налаштовувати інтерфейс оператора-технолога та здійснювати моніторинг стану агрегатів екструзійної лінії та перебігу технологічного процесу екструзії.

4. Розроблений мовою JavaScript функціональний модуль системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів, до складу якого входить математична модель енергетичного балансу та потужності екструдера, інтегровано в розроблений Web-додаток, що дозволило розраховувати рекомендації, щодо підвищення ефективності роботи екструзійної лінії відносно поточного режиму роботи екструзійної лінії.

Особистий внесок

Теоретичні дослідження виконані автором самостійно. Внесок автора полягає у створенні системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів у складі прогресивного Web-додатку.

Апробація результатів

Результати кваліфікаційної роботи магістра доповідались та обговорювались на наукових конференціях Інженерно-наукового інституту ім. Ю. М. Потебні Запорізького національного університету та на I-й Всеукраїнській науково-практичній конференції здобувачів вищої освіти, аспірантів та молодих вчених “Актуальні питання сталого науково-технічного та соціально-економічного розвитку регіонів України” від 19-21 жовтня 2021 року.

Публікації

Основні результати кваліфікаційної роботи магістра опубліковані в тезах доповідей:

1. Крохмаль. П.О., Мінняйло Н.О. Багаторівнева система управління процесом екструзії полімерних матеріалів. *Молода Наука-2021* : збірник наукових праць студентів, аспірантів, докторантів і молодих вчених. м. Запоріжжя, 13 травня 2021 р. Запоріжжя: ЗНУ, 2021. Т. V. С. 119–121.

2. Крохмаль П.О., Міняйло Н.О. Web-додаток до системи управління процесом екструзії полімерних матеріалів. *Актуальні питання сталого науково-технічного та соціально-економічного розвитку регіонів України* : матеріали I-ої Всеукраїнської науково-практичної конференції здобувачів вищої освіти, аспірантів та молодих вчених. м. Запоріжжя, 19-21 жовтня 2021 р. Запоріжжя, 2021. С. 214–216.

Структура та обсяг кваліфікаційної роботи магістра

Кваліфікаційна робота складається з вступу, 5 розділів, висновків, 35 використаних джерел. Текст кваліфікаційної роботи викладено на 132 сторінках, містить 70 рисунків, 20 таблиць, 4 додатка на 5 аркушах.

РОЗДІЛ 1 ТЕХНОЛОГІЯ ЕКСТРУЗІЇ ПОЛІМЕРІВ

1.1 Аналіз літературних джерел

1.1.1 Технологія екструзії полімерів

У [1] автором надано узагальнений результат теорії і практики екструзії полімерних матеріалів за останні 15 років:

– огляд актуальних технологічних розробок екструдерів різного типу та призначення, включаючи сучасні високошвидкісні одношнекові екструдери;

– розгляд конструкції та основних складових одношнекового екструдера: типи електродвигунів та їх особливості, різновиди геометрії отворів подачі сировини у шнек, правильна форма бункера, система нагріву та охолодження;

– фундаментальні основи екструзії, що включають теорію термодинаміки, рівняння балансів, перенесення тепла та ключові властивостей полімерів, і як вони безпосередньо впливають на сам процес екструзії;

– функціональний аналіз процесу екструзії: транспортування сировини, пластифікації, рух розплаву та формування профілю екструдата;

– діагностика екструдерів та вимоги для ефективного вирішення технологічних проблем пов'язаних з роботою екструдера;

– актуальні досягнення у сфері моделювання та симуляції екструзійного процесу з використанням 2-d та 3-d моделей.

Автор [1], розглядаючи процес екструзії, використовує рівняння балансу, що описують перенесення маси, імпульсу та енергії та виділяє два важливих параметра екструзійного процесу – тиск розплаву та температура, що є показниками того як добре функціонує екструдер.

Моніторинг та керування тиском розплаву, як заявляє автор [1], дає змогу відстежувати стабільність протікання екструзії та попереджувати аварійні ситуації.

Автор підкреслює, що рівномірність температури та тепловий режим, що виробляється внаслідок дії сил тертя шнека через бокові стінки циліндру та тепло, що підводиться з боку кільцевих нагрівачів значною мірою впливають як на швидкість матеріального потоку, так і на фізико-хімічні властивості полімеру.

Автор [1] виділяє, що для ефективного вирішення проблем в процесі екструзії повинні бути виконані наступні вимоги:

- правильні технічні засоби автоматизації;
- глибоке розуміння екструзійного процесу;
- обов'язковий збір та зберігання технологічних даних для подальшого аналізу;
- інформація про поточний стан обладнання;
- достовірна інформація про сировину.

В [1] також описані аналітичні та чисельні методи аналізу і комп'ютерного моделювання екструзії полімеру.

Отримання якісної полімерної продукції передбачає застосування методів комп'ютерного моделювання та математичних підходів для вирішення завдань з підбору оптимальних технологічних параметрів та покращення техніко-економічних показників.

Таким чином, процес екструзії являє собою складний технологічний процес, який складається з сукупності складних з боку моделювання взаємопов'язаних процесів – переносу твердих тіл, пластифікації, переносу розплаву, змішуванні та формуванні профілю виробу.

1.1.2 Теорія і практика екструзії полімерів

В [2] розглянуті теоретичні основи і практичні аспекти процесу екструзії полімерів.

Оскільки процес екструзії представляє собою деформацію матеріалу та його течію в геометричному просторі, автор [2] розглядає цей процес як сукупність трьох рівнянь, що описують природу рідин – нерозривності, руху і енергії.

Особливу увагу приділено описанню та розрахунку складових конструкції основного екструзійного обладнання – одношнекових та двошнекових екструдерів, формуючих агрегатів для виготовлення листів та труб, іншого допоміжного обладнання.

Автор [2] підкреслює, що якість екструзії в одношнековому екструдері безпосередньо залежить від того як виконуються три ключові операції: транспортування твердих частинок, їх плавлення та продавлення через формуючу фільтру.

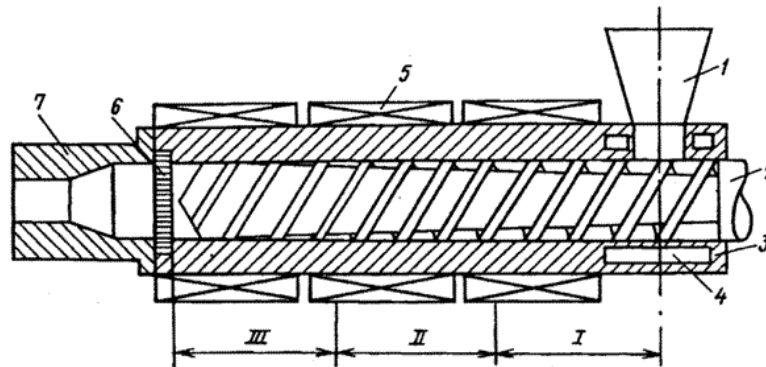
Щоб забезпечити ефективне просування матеріалу в циліндрі, шнек повинен мати як найменший коефіцієнт тертя через матеріал і як найбільший коефіцієнт тертя через стінку циліндру.

Температура в зоні дозування, значною мірою впливає на в'язкість матеріалу, що призводить до зміни тиску потоку матеріалу і як наслідок до перегріву чи недогріву сировини в зоні дозування.

Автор [2] підкреслює, що конструктивне виконання шнека, частота обертання електродвигуна та спосіб завантаження сировини значною мірою впливають на продуктивність екструдера.

Не менш важлива роль у [2] відведена процесу моделюванню одношнекового екструдера як моделі, що складається з окремих функціональних ділянок екструдера – зони завантаження I і пластифікації II, зони дозування (видавлювання) III та зони переходу з II до III.

На рисунку 1.1 зображена схема одношнекового екструдера та розподіл його робочого простору на функціональні ділянки:



1 – бункер; 2 – шнек; 3 – циліндр; 4 – порожнина для циркуляції води;

5 – нагрівач; 6 – решітка з сіткою; 7 – фільтр

I – зона завантаження; II – зона пластифікації; III – зона дозування розплаву

Рисунок 1.1 – Схема одношнекового екструдера

Зона дозування, як підкреслює автор [2], відіграє велику роль в одношнековому екструдері оскільки що зона дозування екструдера є досить вивчена саме вона визначає його продуктивність.

В зоні дозування існує три основні потоки: Q_D – прямий потік, Q_p – зворотній потік та Q_l – потік витіку, звідки продуктивність екструдера Q виражається наступним чином:

$$Q = Q_D - Q_p - Q_l \quad (1.1)$$

На ці потоки (формула 1.1), а значить на саму продуктивність, значною мірою впливають наступні фактори: геометрія шнека, швидкість обертання шнека, тиск та в'язкість розплаву.

Таким чином, запропоновані теоретичні розрахунки, що приводяться в [2] та описана математична модель одношнекового екструдера можуть бути використані для подальших досліджень в кваліфікаційній роботі магістра.

1.1.3 Плавлення полімерів в екструдерах

В [3] детально описується математичне моделювання зон дозування, завантаження та плавлення екструдера, шляхом використання сучасних математичних і обчислювальних методів, щоб максимально достовірно описати процеси, що відбуваються в пластифікованому екструдері.

Автори [3] підкреслюють, що зона дозування екструдера досить добре вивчена, але процес моделювання зони плавлення та завантаження ще залишається з деякими обмеженнями і упущеннями, тому вони намагаються заповнити недоліки з використанням сучасних математичних методів.

За основу побудови математичних моделей процесів течії рідини та теплообміну в зонах екструдера автори використовують закони збереження маси, кількості руху та енергії для стаціонарного режиму:

– рівняння нерозривності маси в прямокутних координатах (x, y, z) :

$$\frac{d}{dx}(pv_x) + \frac{d}{dy}(pv_y) + \frac{d}{dz}(pv_z) = 0; \quad (1.2)$$

– рівняння руху в прямокутних координатах (x, y, z) :

$$p \left(v_x \frac{dv_x}{dx} + v_y \frac{dv_y}{dy} + v_z \frac{dv_z}{dz} \right) = -\frac{dP}{dx} + \frac{d\tau_{xx}}{dx} + \frac{d\tau_{xy}}{dy} + \frac{d\tau_{xz}}{dz}; \quad (1.3)$$

– рівняння енергії (x, y, z) :

$$\begin{aligned} pC \left(v_x \frac{dT}{dx} + v_y \frac{dT}{dy} + v_z \frac{dT}{dz} \right) &= \frac{d}{dx} \left(k \frac{dT}{dx} \right) + \frac{d}{dy} \left(k \frac{dT}{dy} \right) + \frac{d}{dz} \left(k \frac{dT}{dz} \right) + \\ + 2\tau_{xx} \frac{dv_x}{dx} + 2\tau_{yy} \frac{dv_y}{dy} + 2\tau_{zz} \frac{dv_z}{dz} + \tau_{xy} \left(\frac{dv_x}{dy} + \frac{dv_y}{dx} \right) + \tau_{yz} \left(\frac{dv_y}{dz} + \frac{dv_z}{dy} \right) + \\ + \tau_{zx} \left(\frac{dv_z}{dx} + \frac{dv_x}{dz} \right); \end{aligned} \quad (1.4)$$

В рівняннях 1.2 – 1.4: p – густина, C – питома теплоємність; k – коефіцієнт теплопровідності полімеру; v_x, v_y, v_z – компоненти вектору швидкості, T – температура, P – тиск, τ_{ij} – компоненти тензора напруги.

Для вирішення математичних моделей автори [3] використовують метод кінцевих елементів та різниць.

В ході математичних обчислень, автори роблять висновок, що на роботу зон екструдера впливає безліч факторів (технологічні, геометричні та теплофізичні). Розподіл температури корпусу екструдера також значною мірою впливає на перегрів чи гірше плавлення полімеру.

Не менш важливим фактором, що впливає на продуктивність екструдера, як зазначають автори [3], є радіальний отвір між шнеком і циліндром, збільшення якого призводить до виникнення потоків витоків, що призводять до зміни тиску по всій довжині каналу шнека.

Потужність, що витрачається шнеком, розподіляється по зонам екструдера – завантаження, плавлення, дозування, значною мірою залежить від фізико-хімічних властивостей полімеру. Так в залежності від рівня в'язкості полімеру змінюється кількість дисипативного тепла, тому в залежності від цього збільшується/зменшується навантаження на електродвигун екструдера, котрий обертає шнек та величина теплової енергії, що підводиться з боку нагрівачів.

Таким чином, розглянутий матеріал [3] дає уявлення про тонкощі комплексного моделювання процесу екструзії, основні закономірності процесів теплообміну, руху і плавлення полімерів в гвинтових каналах екструдерів та вплив технологічних і геометричних параметрів на роботу екструдера.

1.2 Екструзійна лінія як технологічний об'єкт

Екструзійна лінія представляє собою декілька, з'єднаних послідовно агрегатів [4, 5, 6], що злагоджено працюють для отримання задовільних фізико-хімічних властивостей кінцевого продукту – філаменту.

На рисунку 1.2 зображено лінію екструзії, що складається з екструдера, охолоджувальної зони та намотувача.



Рисунок 1.2 – Лінія екструзії

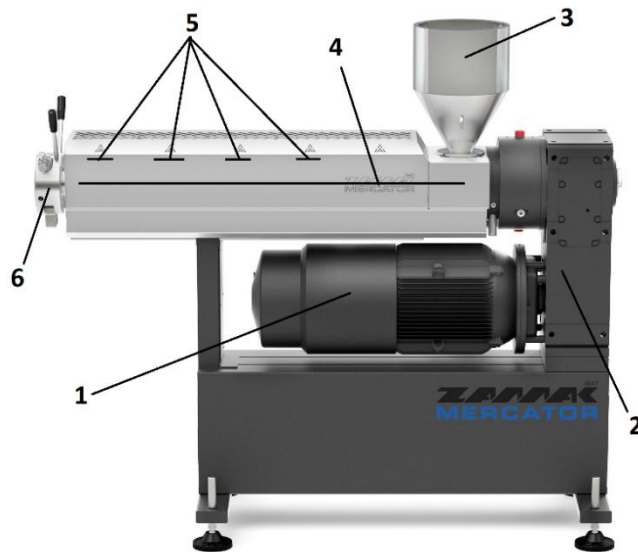
В екструдері гранули полімеру проходять процес екструзії при заданій температурі, відповідно до властивостей полімеру, і через формуюче сопло виходять у вигляді прутка, що охолоджується у ванні з водою або повітрям.

За допомогою намотувача і зворотного зв'язку з датчиком діаметру досягається номінальні діаметри філаменту – 1.75 / 3 мм.

1.2.1 Одношнековий екструдер

Одношнековий екструдер – головний агрегат екструзійної лінії, призначений для безперервної переробки полімерної сировини (гранул, полімерної стружки) в однорідний розплав та формування фізичної форми кінцевого продукту шляхом продавлювання через екструзійну головку (філь'єру).

Одношнековий екструдер зображено на рисунку 1.3.



1 – електродвигун; 2 – редуктор; 3 – завантажувальний бункер;

4 – циліндр з шнеком; 5 – кільцевий нагрівач під термо-бар'єром; 6 – фільєра

Рисунок 1.3 – Типовий одношнековий екструдер та його складові

Основні технологічні характеристики одношнекового екструдера:

- L та D – довжина та діаметр черв'якового шнека відповідно;
- відношення L/D ;
- максимальна швидкість обертання шнека, n – об/хв;
- геометричний профіль шнека (загальний діаметр, кут нарізки, глибина нарізки і т.д.) (рис. 1.4):

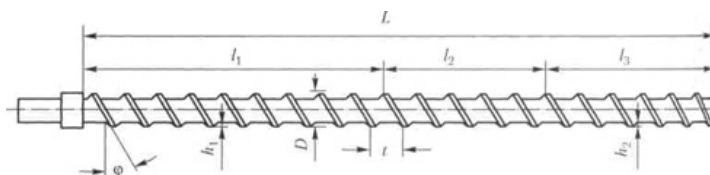


Рисунок 1.4 – Схема конструкції шнека

– ступінь стиснення (компресії) – відношення обсягу одного витка черв'яка в зоні завантаження до обсягу одного витка в зоні дозування.

В залежності від відношення L/D екструдери поділяються на:

- короткошнекові – $L/D = 12...18$;
- довгошнекові – $L/D > 30$.

Найбільш поширеними є екструдери з відношенням $L/D = 20-25$.

Показником роботи екструдера є його ефективність, що виражається формулою 1.5:

$$E = \frac{F}{P}, \quad (1.5)$$

де F – продуктивність екструдера, кг/год;

P – спожита потужність, кВт/год.

Підсистема управління роботою екструдера контролює наступні параметри:

- частоту обертання шнека (зворотній зв'язок з таходатчиком або датчиком Холла);
- температуру нагрівача/чів (зворотній зв'язок з термопарами);
- температуру розплаву в зоні дозування;
- рівень сировини в завантажувальному бункері (ультразвуковий перетворювач рівня);
- тиск розплаву (датчик тиску);
- температуру електродвигуна;
- діаметр прутка (цифровий мікрометр);
- швидкість обертання намотувача та його каретки.

1.2.2 Протягувач

Пристрій, котрий встановлюється на виході екструдера для протягування з відповідною швидкістю прутка філаменту з метою підтримання його необхідного натягу та попередження провисання.

Конструктивно, протягувач складається з крокового електродвигуна та прижимного ролику (рис. 1.5).

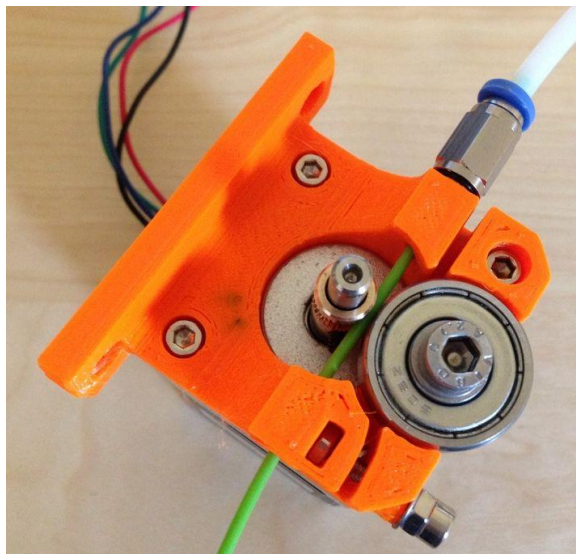


Рисунок 1.5 – Протягувач

Контрольований параметр з боку системи управління – швидкість протяжки (об/хв) протягувача. Швидкість протяжки безпосередньо впливає на діаметр прутка.

1.2.3 Намотувач

Пристрій, призначений для поступового намотування та акумуляції філаменту, що поступає з екструдера на бабіну.

Конструктивно, намотувач складається з двох основних частин – котушки та каретки (рис. 1.6).

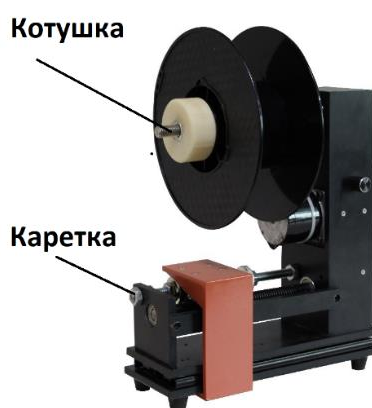


Рисунок 1.6 – Намотувач полімерного прутка

Каретка, що переміщується відносно ширини бабіни, забезпечує рівномірне укладання та натяг прутка.

Для контролю кількості філаменту, намотувач обладнаний вагами або лічильником довжини.

Контрольований параметр з боку системи управління – швидкість намотування (об/хв) та акумульована довжина філаменту (прутка).

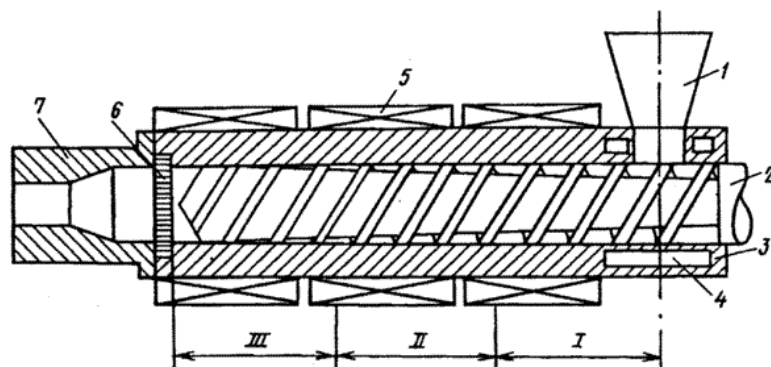
1.3 Особливості технологічного процесу екструзії

Екструзія – це спосіб переробки полімерних матеріалів безперервним продавлюванням їх розплаву через формуючу голівку (фільтеру/сопло), геометрична форма якої визначає профіль одержуваного виробу [2].

Основним обладнанням екструзійного процесу [4, 5, 6] є черв'ячний екструдер, оснащений формуючою голівкою. В процесі роботи агрегату, полімерний матеріал розплавляється, пластифікується і потім нагнітається в фільтеру.

1.3.1 Процеси, що відбуваються в момент екструзії

Технологічний процес екструзії [7, 8] складається з послідовного переміщення матеріалу в міжгвинтовому просторі шнека, який умовно поділений на три ділянки (рис. 1.7).



1 – бункер; 2 – шнек; 3 – циліндр; 4 – порожнина для циркуляції води;

5 – нагрівач; 6 – решітка з сіткою; 7 – фільтера

I – зона живлення; II – зона пластифікації; III – зона дозування розплаву

Рисунок 1.7 – Схема одношнекового екструдера

Поділ на зони I-III здійснюється за технологічною ознакою і вказує на те, яку операцію в основному виконує дана ділянка шнека. В залежності від природи полімеру, що переробляється, температурно-швидкісного режиму процесу та інших чинників, ці зони можуть змінюватися вздовж шнека.

Циліндр, в якому обертається шнек, також має певні довжини зон обігріву, які безпосередньо залежать від розташування нагрівачів на його поверхні і їх температури.

Межі зон шнека I-III і зон обігріву можуть не збігатися.

Вихідна сировина для екструзії, що подається у бункер, може бути у вигляді порошку, гранул, стрічок. Рівномірне дозування матеріалу в бункер забезпечує хорошу якість вихідного продукту.

Переробка полімеру у вигляді гранул – найкращий варіант живлення екструдера. Це пояснюється тим, що гранули менш схильні до утворення “склепінь” в бункері, ніж порошок, що виключає пульсації потоку на виході з екструдера.

Завантаження міжвіткового простору під воронкою бункера відбувається на відрізку довжини шнека, що дорівнює $(1 \div 1,5) D$. При утворенні “склепінь” на стінках бункера живлення шнека матеріалом припиняється. Для усунення цієї небажаної ситуації в бункер додатково встановлюють змішувачі.

Сипучість матеріалу залежить у великій мірі від вологості: чим більше вологість, тим менше сипучість. Тому матеріали повинні бути спочатку підсушені у сушильному бункері при температурі 80-90 °С.

Для збільшення продуктивності машини гранули зазвичай попередньо підігривають, а для збільшення продуктивності машини (в 3-4 рази) – застосування примусової подачі матеріалу з бункера на шнек.

При ущільненні матеріалу в міжвітковому просторі шнека, витіснене повітря виходить назад через бункер. Якщо видалення повітря буде неповним, то воно залишиться в розплаві і після формування утворює у виробі порожнини, що є браком виробів.

Зміна рівня заповнення бункера матеріалом по висоті також впливає на повноту заповнення шнека. Тому бункер забезпечений спеціальними автоматичними рівнемірами, по команді яких відбувається завантаження бункера матеріалом до потрібного рівня.

Завантаження бункера екструдера зазвичай здійснюється за допомогою пневмотранспорту.

При тривалій роботі екструдера можливий перегрів циліндра під воронкою бункера і самого бункера. В цьому випадку гранули почнуть злипатися і припиняться їх подача на шнек. Для запобігання перегріву цієї частини циліндра в ньому робляться порожнини для циркуляції охолоджуючої води.

В зоні живлення гранули, що надходять з бункера, заповнюють міжвітковий простір шнека зони I і ущільнюються. Ущільнення і стиснення гранул в зоні I відбувається, як правило, за рахунок зменшення глибини нарізки h шнека.

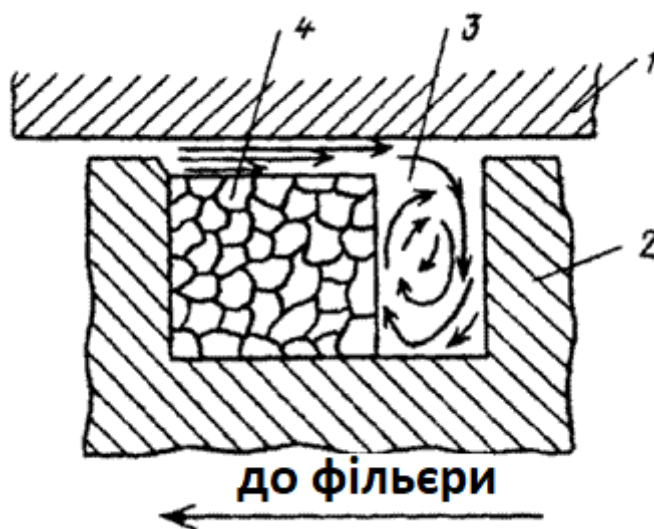
Просування гранул здійснюється внаслідок різниці значень сили тертя полімеру об внутрішню поверхню корпусу циліндра і об поверхню шнека.

Оскільки поверхня контакту полімеру з поверхнею шнека більше, ніж з поверхнею циліндра, необхідно зменшити коефіцієнт тертя полімеру об шнек, так як в іншому випадку матеріал перестане рухатися уздовж осі шнека, а почне обертатися разом з ним. Це досягається підвищенням температури стінки циліндра (нагріванням) і зниженням температури шнека (шнек охолоджується зсередини водою).

Нагрівання полімеру [9, 10] в зоні I відбувається за рахунок дисипативного тепла, що виділяється при терті матеріалу і за рахунок додаткового тепла від нагрівачів, розташованих по периметру циліндра.

Іноді кількість дисипативного тепла може бути достатньою для плавлення полімеру, і тоді нагрівачі відключають. На практиці таке відбувається рідко.

При оптимальній температурі процесу полімер спресований, ущільнений і утворює у міжвітковому просторі тверду пробку (рис. 1.8).



1-стінки циліндру; 2-гребінь шнека; 3-потоки розплаву полімеру; 4-спресований твердий полімер (пробка) в екструдері

Рисунок 1.8– Схема плавлення полімеру в зоні II в міжвітковому просторі шнека

Властивості пробки багато в чому визначають продуктивність машини, стабільність транспортування полімеру, величину максимального тиску і т. д.

На початку зони пластифікації II відбувається підплавлення полімеру, що примикає до поверхні циліндра. Розплав поступово накопичується і тисне по ширині пробки, що убуває. Оскільки глибина нарізки шнека зменшується в міру просування матеріалу від зони I до зони III, то виникає тиск, що змушує пробку щільно притискатися до гарячої стінки циліндра, таким чином відбувається плавлення полімеру.

В зоні пластифікації пробка плавиться також і під дією тепла, що виділяється внаслідок внутрішнього в'язкого тертя в матеріалі в тонкому шарі розплаву, де відбуваються інтенсивні зсувні деформації. Таким чином виникає змішувальний ефект: розплав інтенсивно гомогенізується, а складові композиційного матеріалу перемішуються.

Кінець зони II характеризується розпадом пробки на окремі фрагменти. Далі розплав полімеру із залишками твердих частинок потрапляє в зону дозування.

Збільшення тиску (P) розплаву відбувається на кордоні зон I і II. На цій межі утворюється пробка з спресованого матеріалу, яка “ковзає” по шнеку: в зоні I це твердий матеріал, а в зоні II-плавиться. Наявність цієї пробки і створює основний внесок у підвищення тиску розплаву.

Збільшення тиску можливе також за рахунок зменшення глибини нарізки шнека.

Накопичений на виході з циліндра тиск, витрачається на подолання опору сіток для фільтрації, течії розплаву в каналах головки і формування виробу.

Просування гетерогенного матеріалу (розплав, частинки твердого полімеру) продовжує супроводжуватися виділенням внутрішнього тепла, яке є результатом інтенсивних зсувних деформацій в полімері. Розплавлена маса продовжує гомогенізуватися, що проявляється в остаточному плавленні залишків твердого полімеру, усередненні в'язкості та температури розплавленої частини.

У міжвітковому просторі розплав має ряд потоків, основними з яких є продольний і циркуляційний.

Величина продольного (вздовж осі шнека) потоку визначає продуктивність екструдера Q , а циркуляційного — якість гомогенності полімеру або змішування компонентів.

Продольний потік, у свою чергу, складається з трьох потоків розплаву: прямого, зворотного і потоку витоків:

- прямий потік викликаний рухом шнека в напрямку формуючої головки;

- зворотний потік – це умовна течія, що викликається високим тиском з боку головки; в реальності не існує;

– потік витоку відбувається при перетіканні розплаву між циліндром і гребнем черв'яка.

Продуктивність Q екструдера з урахуванням розподілу швидкостей різних потоків становить:

$$Q = Q_{\text{пр}} - Q_{\text{зв}} - Q_{\text{вит}}, \quad (1.6)$$

де $Q_{\text{пр}}$ – продуктивності екструдера від прямого потоку;

$Q_{\text{зв}}$ – продуктивності екструдера від зворотнього потоку;

$Q_{\text{вит}}$ – продуктивності екструдера від витоку розплаву.

Загальна продуктивність шнекового екструдера виражена наступною залежністю:

$$Q = \alpha \cdot n - \beta \cdot \frac{\Delta P}{\mu \cdot L}, \quad (1.7)$$

де n — частота обертання шнека;

ΔP — тиск на виході із шнека (у кінці зони III);

μ — ефективна в'язкість розплаву;

L – довжина шнека;

α – константа швидкості прямого потоку;

β – константа швидкості зворотного потоку, які залежать від геометричних параметрів шнека.

1.3.2 Основні параметри екструзійного процесу

Головні технологічні параметри процесу екструзії включають:

- температуру полімеру в зонах пластифікації та дозування;
- тиск розплаву;
- швидкість обертання шнека;
- режим охолодження сформованого екструдата на виході з фільєри;
- поточна товщина філаменту;
- швидкість протягувача;
- швидкість намотувача та кількість акумульованого філаменту.

1.3.3 Матеріали, що використовуються для екструзії

Більшість термопластів і композицій на їх основі можуть перероблятися екструзією [11, 12]. Для цього достатньо, щоб час перебування розплаву в екструдері при даній температурі був менший часу термостабільності полімеру при тій же температурі.

При занадто високій в'язкості розплаву отримувати вироби методом екструзії важко через великого опору течії розплаву, виникнення нестійкого режиму руху потоку. Все це призводить до утворення дефектів виробів.

Підвищення температури переробки може призвести до термодеструкції розплаву, а збільшення тиску, шляхом збільшення потужності привода при низьких температурах — до механічної деструкції обладнання.

Таким чином, для екструзії розплавів повинні застосовуватися полімери з досить вузьким інтервалом коливання в'язкості.

Найбільш широко застосовується екструзія полімерів для 3D-друку [13, 14, 15] наступних типів (табл. 1.1).

Таблиця 1.1 – Найбільш поширені види полімеру для 3D-друку

Тип полімеру	Температура екструзії, °С	Теплоємність, $\frac{\text{кДж}}{\text{кг} \cdot \text{К}}$	В'язкість, Па·с
ABS	220-250	1935	1738
PLA	190-220	1500	1678
PETG	230-250	1200	1440
Nylon	220-270	1310	1921
Flex	225-245	2100	1556

В розділі розглянуто технологію виготовлення полімерного філаменту та особливості технологічного процесу екструзії.

Розглянуто складові екструзійної лінії – одношнековий екструдер, протягувач та намотувач.

Шляхом аналізу літературних джерел, вивчено особливості фізико-хімічних процесів, що протікають в екструдері під час процесу екструзії.

Виділено ключові технологічні параметри екструзії, такі як температура полімерної пробки в зонах пластифікації та дозування, швидкість обертання шнека, тиск розплаву, швидкість охолодження, поточна товщина філаменту, швидкість протягувача та намотувача.

Перелічено ключові полімерні матеріали та їх фізико-хімічні властивості, котрі використовуються для виготовлення філаменту та впливають на режим роботи екструзійної лінії.

РОЗДІЛ 2 ПОБУДОВА МАТЕМАТИЧНОЇ МОДЕЛІ СППР

2.1 Методи і технології синтезу математичних моделей процесів екструзії в гнучких виробництвах полімерних матеріалів

В [16] авторами запропоновані методи автоматизованого синтезу адаптованих математичних моделей одношнекових, двошнекових екструдерів в гнучких багатоасортиментних виробництвах монолітних і вспінених полімерних виробів різноманітних видів.

Автори підкреслюють проблему переобладнання виробничої екструзійної лінії на нову продуктивність та марку виробу та пропонують рішення цієї проблеми шляхом створення програмних комплексів, в основі яких лежать математичні деформаційні та теплові моделі. Ці програмні комплекси дають змогу в інтерактивному режимі визначити найкращі параметри функціонування агрегату відповідно до поточної конфігурації.

На рисунку 2.1 автори [16] приводять структуру математичного і інформаційного забезпечення програмного комплексу для динамічного синтезу моделей в залежності від вхідних параметрів.

Структура математичного і інформаційного забезпечення математичної моделі екструзії (рис. 2.1).

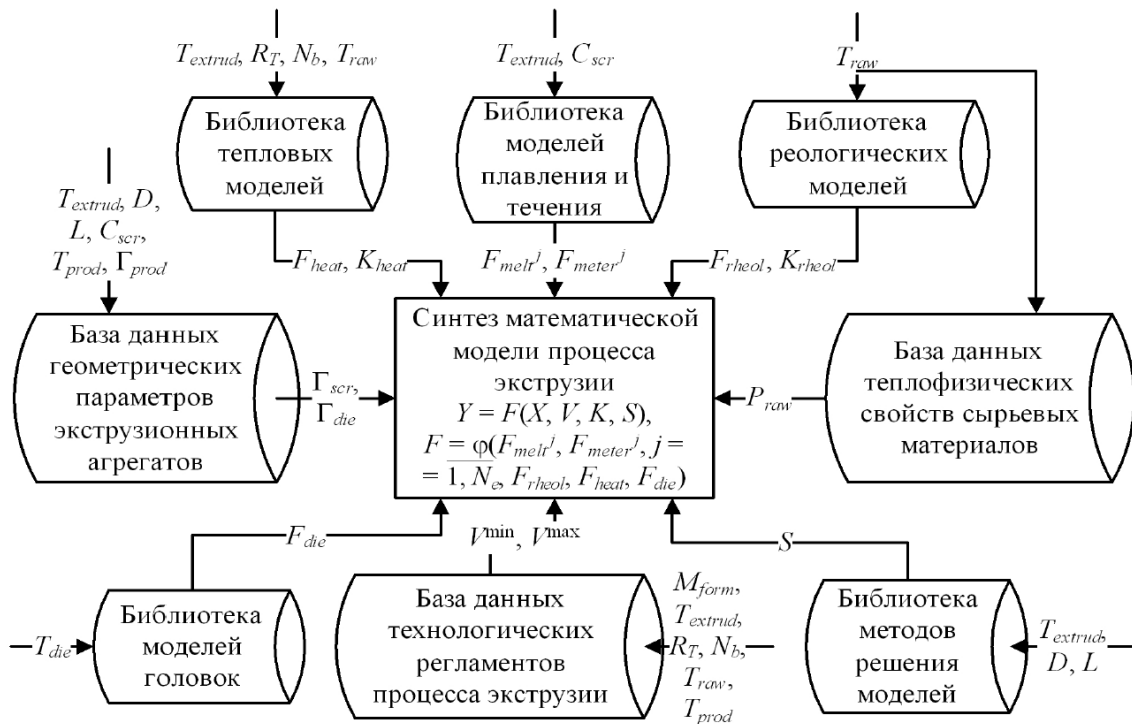


Рисунок 2.1 – Структура математичного і інформаційного забезпечення математичної моделі екструзії

Для більш детального уявлення про алгоритм роботи програмного комплексу, автори [16] приводять етапи синтезу математичних моделей процесів екструзії (рис. 2.2).

Етапи синтезу математичної моделі екструзії (рис. 2.2):

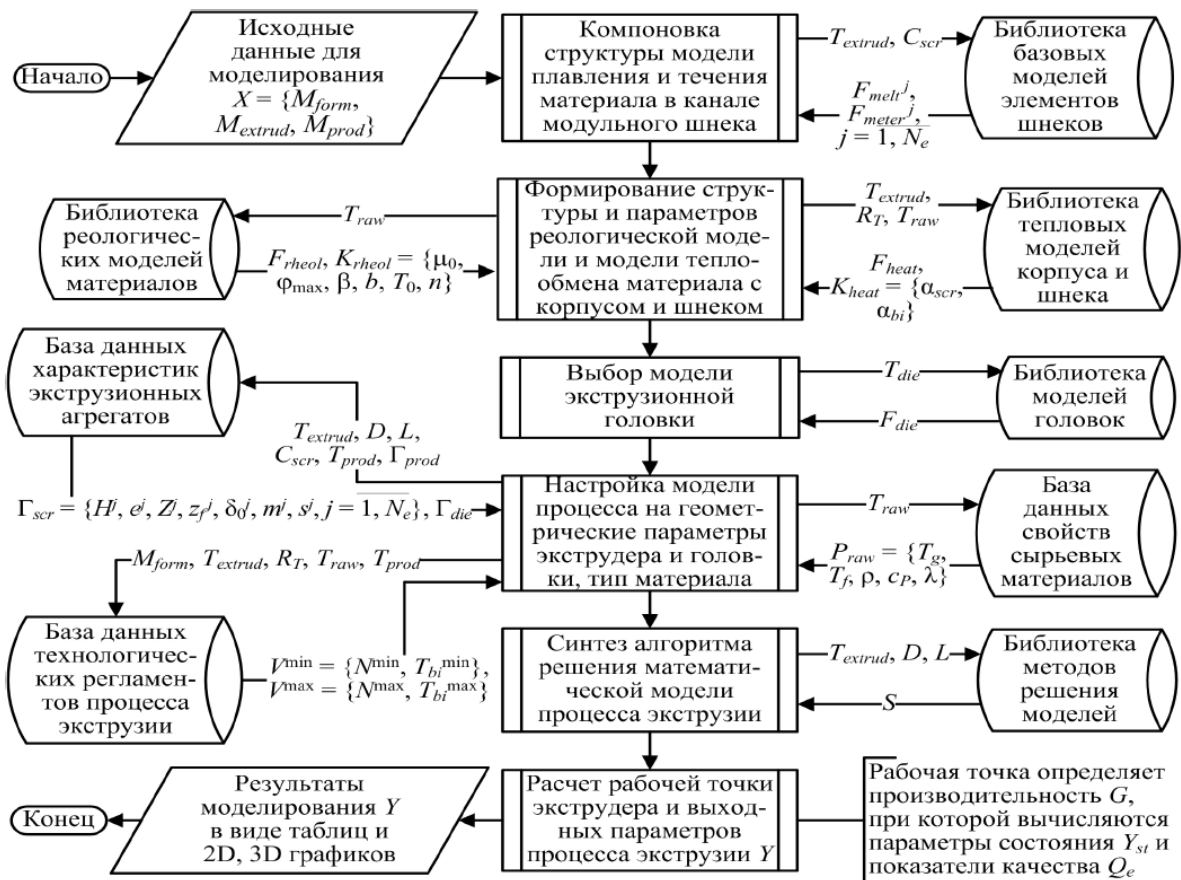


Рисунок 2.2 – Этапы синтезу математичних моделей екструзії

Результатами моделювання автори [16] підтверджують адекватність моделі (рис. 2.3):

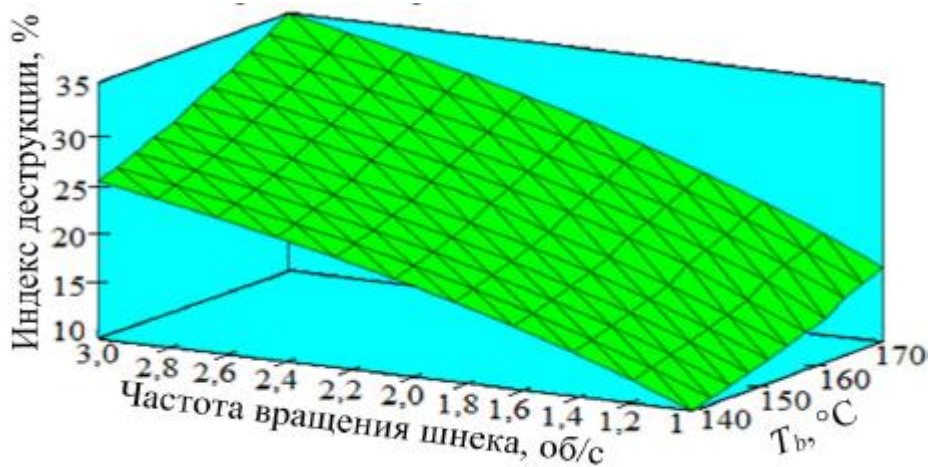


Рисунок 2.3 – Результати моделювання синтезованої математичної моделі екструзії

З підвищенням частоти обертання шнека зростає температура екструдата, що призводить до зростання індексу деструкції, котрий поступово сповільнюється внаслідок зменшення часу перебування полімеру в екструдері і швидкості дисипативного розігріву.

Точність моделі була підтвердження шляхом тестування на робочих технологічних екструдерах.

Таким чином, в статті [16] гнучкий проблемно-орієнтовний програмний комплекс, що дозволяє створювати і налаштовувати функціональні математичні моделі для проектування і управління екструзійними процесами на різноманітні модифікації їх апаратно-технологічної конфігурації, режиму експлуатації і типу полімерних матеріалів.

Концепції, математичний апарат та етапи розробки математичних моделей екструзії полімерних матеріалів будуть використані для досліджень та подальших розробок в кваліфікаційній роботі магістра.

2.2 Розрахунок теплового балансу

Правильне уявлення про тепловий баланс екструдера [16, 17, 18] при переробці полімеру, тобто рівність теплоти, що підводиться та відводиться від агрегату, дає змогу встановити найбільш оптимальні режими переробки, виявити вплив технологічних параметрів на продуктивність обладнання та якість кінцевої продукції, а також отримати данні для економічних розрахунків.

Для проведення розрахунків теплового балансу, необхідно встановити ключові теплофізичні властивості полімерів:

- коефіцієнт температуропровідності $\alpha, \frac{\text{м}^2}{\text{с}}$;
- коефіцієнт теплопровідності $\lambda, \frac{\text{кДж}}{\text{м} \cdot \text{ч} \cdot \text{К}}$;
- теплоємність $C, \frac{\text{кДж}}{\text{кг} \cdot \text{К}}$;
- густина розплаву полімеру $\rho, \frac{\text{кг}}{\text{м}^3}$.

Тепловий баланс екструдера визначається рівнянням [17]:

$$E_{\text{Н}} + E_{\text{Ш}} = E_{\text{М}} + E_{\text{О}} + E_{\text{З}}, \quad (2.1)$$

де $E_{\text{Н}}$ – теплота, що надходить від нагрівачів, кВт;

$E_{\text{Ш}}$ – теплота, що виділяється при роботі шнека (дисипативний нагрів – внаслідок тертя), кВт;

$E_{\text{М}}$ – теплота, що поглинається нагрітим матеріалом, кВт;

$E_{\text{О}}$ – теплота, що поглинається системою охолодження (вода), кВт;

$E_{\text{З}}$ – теплота, що витрачається у зовнішнє середовище, через теплоізоляцію екструдера, кВт.

Виходячи з рівняння теплового балансу, можна розрахувати кількість теплоти, котру необхідно підвести до екструдера з боку системи обігріву:

$$E_{\text{Н}} = E_{\text{М}} + E_{\text{О}} + E_{\text{З}} - E_{\text{Ш}} \quad (2.2)$$

Складові рівняння балансу розраховуються наступним чином:

1. E_M – теплота, що поглинається нагрітим матеріалом, кВт:

$$E_M = G_M \cdot c_M \cdot (t_K - t_{\Pi}) \cdot \frac{1}{3600}, \quad (2.3)$$

де G_M – кількість матеріалу, що переробляється, кг/год;

c_M – питома теплоємність полімеру, $\frac{\text{кДж}}{\text{кг} \cdot \text{К}}$;

t_K, t_{Π} – кінцева і початкова температури полімеру, С.

Вихідні дані для розрахунку: $G_M = 2$ кг/год, $c_M = 1935 \frac{\text{кДж}}{\text{кг} \cdot \text{К}}$ (для ABS пластику), $t_{\Pi} = 24$ С, $t_K = 180$ С.

$$E_M = 2 \cdot 1935 \cdot (180 - 24) \cdot \frac{1}{3600} = 168 \text{ Вт} = 0.168 \text{ кВт} \quad (2.4)$$

2. Теплота, що поглинається системою охолодження (вода), кВт:

$$E_O = G_B \cdot c_B \cdot (t_{BK} - t_{B\Pi}), \quad (2.5)$$

де G_B – кількість (витрата) води, що надходить на охолодження шнека, л/с;

c_B – питома теплоємність води, $\frac{\text{кДж}}{\text{кг} \cdot \text{К}}$;

$t_{BK}, t_{B\Pi}$ – кінцева і початкова температури води, С.

Кількість води G_B , що надходить на охолодження шнека, л/с розраховується за наступною формулою:

$$G_B = \rho \cdot F \cdot v, \quad (2.6)$$

де ρ – густина води, $\frac{\text{кг}}{\text{м}^3}$;

F – площа поперечного перерізу, м^2 ;

v – швидкість течії води, м/с.

Площа поперечного перерізу розраховується за наступною формулою:

$$F = \frac{\pi \cdot d^2}{4}, \quad (2.7)$$

де d – діаметр трубки в циліндрі шнека, м.

Вихідні дані для розрахунку: оскільки в розробленому зразку екструдера системи охолодження не передбачено, тому складова $E_0 = 0$ кВт.

3. Теплота, що витрачається у зовнішнє середовище, через теплоізоляцію екструдера, кВт:

$$\begin{aligned} E_B &= F_k \cdot \alpha \cdot (t_{\text{п}} - t_c), \\ F_k &= \pi \cdot d_k \cdot L_k, \\ \alpha &= 9,74 + 0,07 \cdot \Delta t, \\ \Delta t &= t_{\text{п}} - t_c, \end{aligned} \quad (2.8)$$

де F_k – площа зовнішньої поверхні корпусу екструдера;

α – коефіцієнт теплопередачі, $\frac{\text{кВт}}{\text{м}^2 \cdot \text{К}}$;

$t_{\text{п}}$ – температура зовнішньої поверхні ізольованого корпусу, С;

t_c – температура зовнішнього середовища, С;

d_k – діаметр корпусу з урахуванням ізоляції, м;

L_k – довжина корпусу, м.

Вихідні дані для розрахунку: $d_k = 0.2$ м, $L_k = 0.5$ м, $t_{\text{п}} = 40^\circ\text{С}$, $t_c = 24^\circ\text{С}$.

$$\begin{aligned} F_k &= 3.14 \cdot 0.2 \cdot 0.5 = 0.314 \text{ м}^2, \\ \alpha &= 9,74 + 0,07 \cdot (40 - 24) = 10.86 \frac{\text{кВт}}{\text{м}^2 \cdot \text{К}} \end{aligned} \quad (2.9)$$

$$E_3 = 0.314 \cdot 10.86 \cdot (40 - 24) = 54 \text{ Вт} = 0.054 \text{ кВт}$$

4. Теплота, що виділяється при роботі шнека (дисипативний нагрів – внаслідок тертя), кВт:

$$E_{\text{ш}} = \left(\frac{\pi^3 \cdot D^3 \cdot N^2 \cdot \eta \cdot L_{\text{н}}}{h_{\text{ср}}} + \frac{Q \cdot P}{\cos^2 \phi} + \frac{\pi^2 \cdot D^2 \cdot N^2 \cdot \eta \cdot L_{\text{н}}}{\delta \cdot \text{tg} \phi} \right) \cdot 9,8 \cdot 10^{-7}, \quad (2.10)$$

де D – діаметр шнека, см;

N – частота обертання шнека, об/с;

P – тиск в циліндрі, Па;

η – в'язкість розплаву полімеру, Па · с;

ϕ – кут підйому гвинтової нарізки шнека, град.;

$h_{\text{ср}}$ – середня глибина нарізки шнека, см;

$L_{\text{н}}$ – довжина напірної частини шнека, см;

Q – об'ємна витрата екструдера, $\frac{\text{см}^3}{\text{с}}$;

δ – величина зазору між гребнем шнека і циліндром, см.

$$\phi = \text{arctg} \left(\frac{t}{\pi \cdot (D - h_{\text{ср}})} \right), \quad (2.11)$$

де t – шаг нарізки, см.

Вихідні дані для розрахунку: $D = 2$ см, $N = 0.5$ об/с, $P = 8000000$ Па, $\eta = 1748$ Па · с (для ABS пластику), $t = 1.5$ см, $h_{\text{ср}} = 1$ см, $L_{\text{н}} = 20$ см, $Q = 0.5$ см³/с, $\delta = 0.02$ см.

$$\phi = \text{arctg} \left(\frac{1.5}{\pi \cdot (2 - 1)} \right) = 25^\circ \quad (2.12)$$

$$E_{\text{ш}} = \left(\frac{\pi^3 \cdot 2^3 \cdot 0.5^2 \cdot 1748 \cdot 20}{1} + \frac{0.5 \cdot 8000000}{\cos^2 25} + \frac{\pi^2 \cdot 2^2 \cdot 0.5^2 \cdot 1748 \cdot 20}{0.02 \cdot \text{tg } 25} \right) \cdot 9,8 \cdot 10^{-7} = \quad (2.13)$$

$$\begin{aligned} &= (2167958 + 4869771 + 36997180) \cdot 9,8 \cdot 10^{-7} = \\ &= 43.15 \text{ Вт} = 0.04 \text{ кВт} \end{aligned}$$

Таким чином, кількість теплоти, котру необхідно підвести до екструдера з боку системи обігріву:

$$E = 0.168 + 0 + 0.054 - 0.04 = 0.182 \text{ кВт} \quad (2.14)$$

Виходячи з того, що на розробленому малогабаритному екструдері встановлено 2 кільцевих нагрівача, потужністю 140 Вт кожний, загальної потужності 280 Вт достатньо, щоб покрити витрати теплоти (приблизно 180 Вт) на матеріал, охолодження та у навколишнє середовище.

Отримана залежність дає змогу оцінити вплив наступних параметрів на кількість теплоти, яку необхідно підвести з боку нагрівачів для забезпечення ефективного прогріву полімеру:

- вид полімеру (теплоємність, в'язкість);
- температура сировини на вході у бункер екструдера (попередній підігрів);
- ізоляція корпусу;
- величина зазору між шнеком і циліндром (знос обладнання);
- частота обертання шнека (зростання/зменшення дисипативного тепла).

2.2.1 Аналіз впливу попереднього підігріву полімеру при подачі в екструдер

Розрахований тепловий баланс дає змогу оцінити вплив початкової температури сировини та загальну кількість тепла, котру необхідно підвести з боку нагрівачів.

В таблицю 2.1 зведено експериментальні дані розрахунків кількості теплоти при зміні температури полімеру на вході в екструдер.

Таблиця 2.1 – Результат розрахунків кількості теплоти в залежності від температури полімеру на вході в екструдер

Температура матеріалу, С	Кількість теплоти, Вт
20	187
40	165
60	144
80	122
100	101

На основі отриманих даних побудовано графік залежності кількості теплоти від температури матеріалу на вході в екструдер (рис. 2.4).

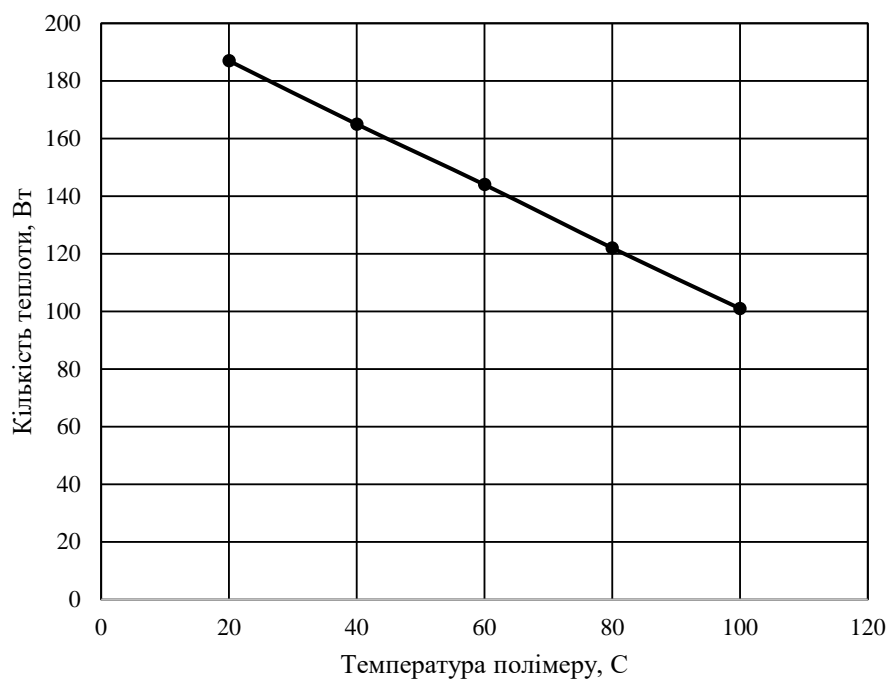


Рисунок 2.4 – Залежність кількості теплоти від початкової температури полімеру

Отримана лінійна залежність, що свідчить про те, що попередньо підігрітий полімер значною мірою впливає на кількість тепла, котре необхідно підвести з боку нагрівачів. Але потрібно враховувати фізико-хімічно властивості протікання екструзійного процесу і конструктивні особливості екструдера, адже занадто підігрітий полімер може призвести до погіршення його просування в циліндрі й перегріву.

2.2.2 Аналіз впливу частоти обертання шнека

Дисипативне тепло, що виділяється в процесі обертання шнека, при просуванні матеріалу, створює додаткове джерело теплової енергії, тим самим зменшуючи необхідність підведення додаткового тепла з боку нагрівачів. Але отримання теплоти внаслідок роботи одного тільки шнека значною мірою збільшує навантаження на електродвигун, тим самим створюючи прискорений зніс обладнання.

В табл. 2.2 зведено результати обчислень кількості теплоти, в залежності від частоти обертання шнека.

Таблиця 2.2 – Результат розрахунків кількості теплоти в залежності від частоти обертання шнека

Частота обертання шнека, об/с	Кількість теплоти, Вт
0.5	182
1	44
1.5	-172
2	-467
2.5	-856

На основі отриманих даних (табл. 2.2) побудовано графік залежності кількості теплоти від частоти обертання шнека (рис. 2.5):

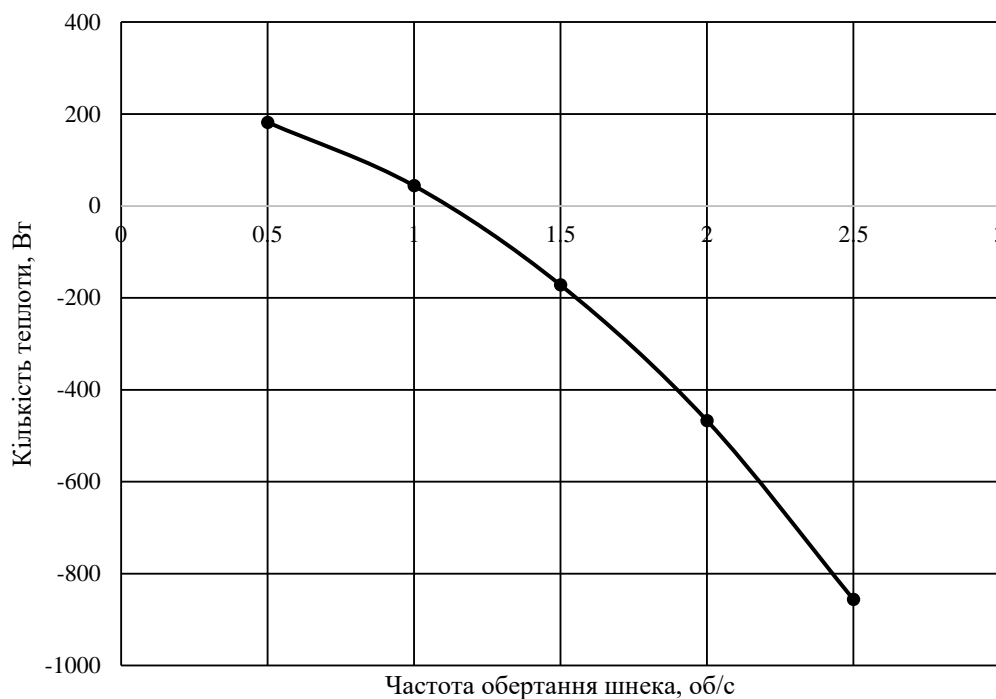


Рисунок 2.5 – Залежність кількості теплоти від частоти обертання шнека

Отриманий результат свідчить, що існує лінійна залежність кількості тепла, яку треба підвести до екструдера від частоти обертання шнека. Зі збільшенням частоти обертання шнека, зростає кількість дисипативного тепла, внаслідок його роботи та збільшується навантаження на електродвигун. Тому для більш ефективної та довгострокової роботи екструзійної установки найбільш ефективним буде додаткове підведення теплоти з боку нагрівачів.

2.2.3 Аналіз впливу теплоємності полімеру

Режим роботи екструдера значною мірою залежить від типу полімеру, що переробляється. На тепловий баланс впливає теплоємність матеріалу, яка зазвичай варіюється в діапазоні від 1200-2400 $\frac{\text{кДж}}{\text{кг} \cdot \text{К}}$ в залежності від виду полімеру.

В табл. 2.3 приведено розподіл теплоємностей полімеру та кількість теплоти, необхідної для його прогріву з урахуванням всіх технологічних чинників.

Таблиця 2.3 – Розподіл теплоємностей полімеру та відповідної кількості теплоти

Теплоємність полімеру, $\frac{\text{кДж}}{\text{кг}\cdot\text{К}}$	Кількість теплоти, Вт
1200	159
1400	176
1600	193
1800	211
2000	228
2200	245
2400	263

На основі отриманих даних (табл. 2.3) побудовано графік залежності (рис. 2.6).

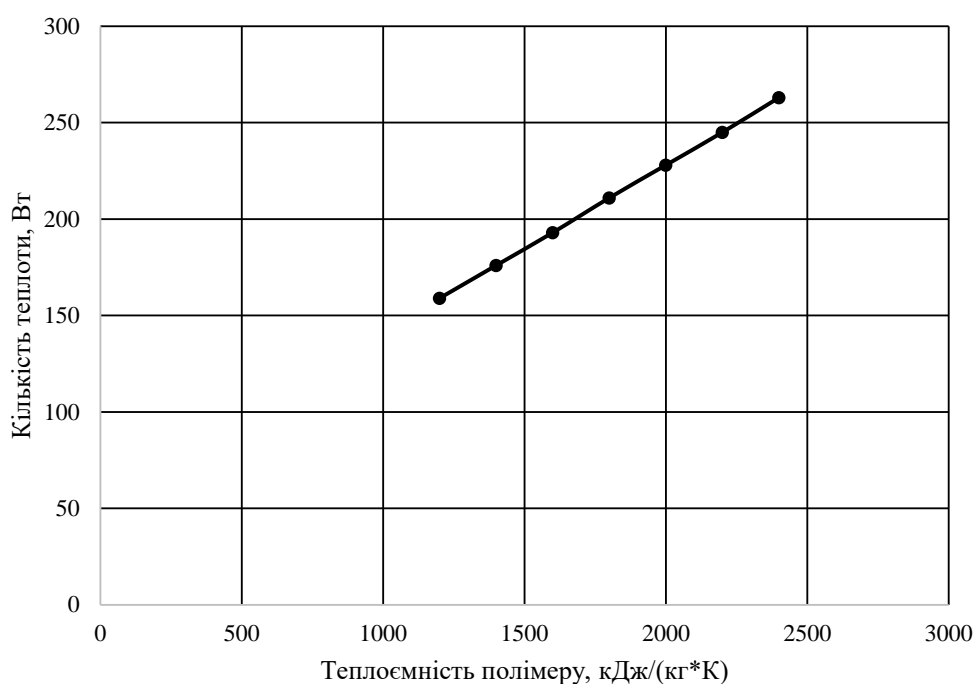


Рисунок 2.6 – Залежність кількості теплоти від теплоємності полімеру

Отримано лінійну залежність. Таким чином, чим більше теплоємність полімеру, тим більше теплоти необхідно підвести з боку нагрівачів для його прогріву.

2.3 Розрахунок потужності електродвигуна екструдера

Потужність, що витрачається екструдером на переміщення полімерної пробки вздовж шнека, відіграє важливу роль для ефективного протікання технологічного процесу екструзії та забезпечення довгострокової строку служби обладнання, а саме – електродвигуна.

Розрахувати потужність, що витрачається екструдером на переміщення полімерної пробки вздовж спірального каналу шнека до зони дозування можливо за формулою [17]:

$$N_1 = 9,8 \cdot 10^{-7} \cdot \left[\frac{\pi^3 \cdot (t - e) \cdot L \cdot J \cdot \eta}{t^2} \cdot n^2 + A \cdot \Delta P \cdot n \right], \quad (2.15)$$

де t – крок нарізки, см;

e – ширина гребня, см;

L – довжина шнека, см;

η – в'язкість розплаву полімеру, Па · с;

n – частота обертання шнека, об/хв;

A – постійна прямого потоку екструдера, см³;

ΔP – перепад тиску в екструдері, Па;

J – коефіцієнт, що розраховується за формулою:

$$J = \frac{\pi^2 \cdot D^2 - 4 \cdot t}{\pi^2} + \frac{(D + d_3)^3 - \left(D + \frac{(d_1 + d_2)}{2}\right)^3}{3 \cdot \left(d_3 + \frac{(d_1 + d_2)}{2}\right)} + \frac{2,3 \cdot \pi^2 \cdot D^5 \cdot \ln \frac{(h_1 + h_2)/2}{h_3}}{(t^2 + \pi^2 + D^2) \left(\frac{(h_1 + h_2)}{2} - h_3\right)}, \quad (2.16)$$

де D – діаметр шнека, см;

d_1 – діаметр серцевини шнека в зоні плавлення, см;

d_2 – діаметр серцевини шнека в зоні пластифікації, см;

d_3 – діаметр серцевини шнека в зоні дозування, см;

h_1 – глибина нарізки шнека в зоні плавлення, см;

h_2 – глибина нарізки шнека в зоні пластифікація, см;

h_3 – глибина нарізки шнека в зоні дозування, см.

Вихідні дані для розрахунку:

$t = 1.5$ см, $e = 1$ см, $L = 48$ см, $\eta = 1748$ Па · с, $n = 0.5$ об/хв, $A = 10$ см³,
 $\Delta P = 8000000$ Па, $D = 2$ см, $d_1 = 1$ см, $d_2 = 1$ см, $d_3 = 1.2$ см, $h_1 = 0.5$ см, $h_2 = 0.5$
 см, $h_3 = 0.4$ см.

$$\begin{aligned}
 J &= \frac{3,14 \cdot 2^2 - 4 \cdot 1,5}{3,14^2} + \\
 &+ \frac{(2 + 1,2)^3 - \left(2 + \frac{(1+1)}{2}\right)^3}{3 \cdot \left(1,2 + \frac{(1+1)}{2}\right)} \\
 &+ \frac{2,3 \cdot 3,14^2 \cdot 2^5 \cdot \ln \frac{(0,5+0,5)}{0,4}}{(1,5^2 + 3,14^2 + 2^2) \left(\frac{(0,5+0,5)}{2} - 0,4\right)} = 113,56
 \end{aligned} \tag{2.17}$$

$$\begin{aligned}
 N_1 &= 9,8 \cdot 10^{-7} \cdot \left[\frac{3,14^3 \cdot (1,5 - 1) \cdot 48 \cdot 113,56 \cdot 1748}{1,5^2} \cdot 0,5^2 + \right. \\
 &\quad \left. + 10 \cdot 8000000 \cdot 0,5 \right] \\
 &= 55.28 \text{ Вт} = 0,05 \text{ кВт}
 \end{aligned} \tag{2.18}$$

Потужність, що витрачається на зріз матеріалу в зазорі між шнеком і стінкою циліндра, розраховується за формулою [1]:

$$N_2 = 9,8 \cdot 10^{-7} \cdot \frac{\pi^3 \cdot D^3 \cdot e \cdot L \cdot \eta}{\delta \cdot t} \cdot n^2, \tag{2.19}$$

де δ – величина зазору між шнеком і стінкою циліндра, см

Вихідні дані для розрахунків: $\delta = 0.02$ см, $t = 1.5$ см, $e = 1$ см, $L = 48$ см,
 $\eta = 1748$ Па · с, $n = 0.5$ об/хв, $D = 2$ см.

$$N_2 = 9,8 \cdot 10^{-7} \cdot \frac{3,14^3 \cdot 2^3 \cdot 1 \cdot 48 \cdot 1748}{0,02 \cdot 1,5} \cdot 0,5^2 = 169,96 \text{ Вт} \quad (2.20)$$

$$= 0.169 \text{ кВт}$$

Загальна потужність, що витрачається на обертання шнека розраховується за наступною формулою:

$$N = N_1 + N_2 \quad (2.21)$$

$$N = 0,05 + 0,169 = 0,219 \text{ кВт}$$

Таким чином, щоб покрити всі інші невраховані витрати потужності електродвигуна (механічні витрати), необхідно до загальної потужності, додати ще 20%.

$$N_{\text{заг}} = N + N \cdot 20\% \quad (2.22)$$

$$N_{\text{заг}} = 0,219 + 0,219 \cdot 20\% \approx 0.3 \text{ кВт}$$

2.3.1 Аналіз впливу зазору між шнеком і стінкою циліндра на загальну потужність електродвигуна екструдера

В табл. 2.4 зведені розрахункові дані потужності в залежності від зміни зазору між шнеком і стінкою циліндра.

Таблиця 2.4 – Розрахункові дані потужності в залежності від зміни зазору між шнеком і стінкою циліндра

Величина зазору, см	Потужність, Вт
0.02	225
0.06	111
0.1	89
0.14	79
0.18	74

На основі отриманих даних (табл. 2.4) побудовано графік залежності (рис. 2.7).

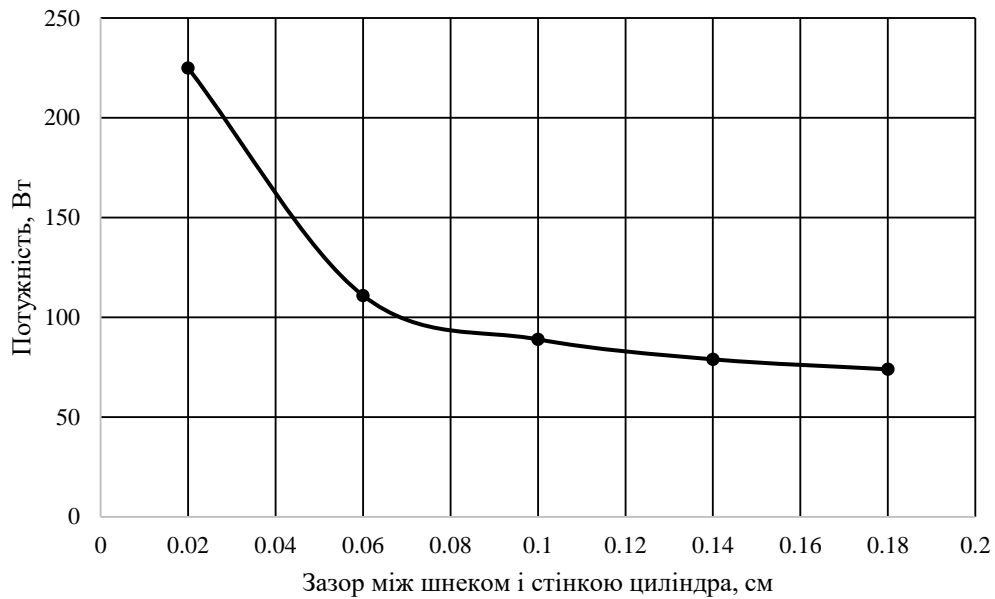


Рисунок 2.7 – Залежність загальної потужності від величини зазору між шнеком і стінкою циліндра

Спостерігається експоненціальна залежність. Очевидно, що при збільшенні зазору, величина навантаження на електродвигун зменшується, але це призводить до того, що полімерна пробка перестає ефективно просуватися по виткам шнека й починає провертатись на місці. Тому для максимально ефективного просування пробки, необхідно, щоб зазор був як найменший й потужності електродвигуна було достатньо, щоб її провертати.

2.4 Структура математичної моделі СППР

В основу математичної моделі системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів покладено математичні розрахунки ключових екструзійних параметрів – теплового балансу та потужності [16, 17].

На рисунку 2.8 приведено структуру математичної моделі системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів.



Рисунок 2.8 – Структура математичної моделі системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів

На вхід системи прийняття рішення по управлінню процесом екструзії полімерних матеріалів надходять поточні параметри обладнання – екструдера (параметри шнека, електродвигуна, системи охолодження, системи нагрівання) та фізико-хімічні властивості полімерного матеріалу, що обробляється.

На базі отриманих вхідних даних система прийняття рішень виконує розрахунок необхідного кількості теплоти та потужності електродвигуна для ефективного протікання технологічного процесу екструзії.

В розділі досліджено енергетичний баланс та потужність екструдера.

Досліджено тепловий баланс екструдера, що складається з витрат теплоти на матеріал, систему охолодження, навколишнє середовище та дисипативного тепла, що виділяється в процесі роботи шнека.

Виконано математичний розрахунок теплового балансу екструдера, звідки виведено залежність необхідної кількості теплоти для прогріву полімерної пробки в міжшнековому просторі при поточних технологічних параметрах екструдера та фізичних властивостей полімерного матеріалу, що переробляється.

Досліджено вплив попереднього підігріву полімеру, частоти обертання електродвигуна та теплоємності полімеру на кінцеву розрахункову кількість теплоти – отримано лінійну залежність в усіх випадках.

Виконано розрахунок необхідної потужності електродвигуна, щоб забезпечити максимальну ефективність протікання екструзійного процесу та збільшити його довготривалий строк служби, шляхом попередження надлишкових навантажень.

Досліджено вплив зазору між шнеком і циліндром на кінцеву розрахункову потужність електродвигуна, отримано експоненціальну залежність.

Розроблено структуру, вхідні та вихідні параметри математичної моделі системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів, котра складається з розглянутих математичних залежностей – теплового балансу та розрахункової потужності екструдера.

РОЗДІЛ 3 РОЗРОБКА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Схема інформаційних та матеріальних потоків

На схемі інформаційних та матеріальних потоків, що приведена на рисунку 3.1, зображено види матеріальних потоків, що супроводжують технологічний процес екструзії.

Оскільки лінія екструзії складається з декількох ключових агрегатів (екструдер, протягувач, намотувач), синхронна робота кожного з них є першочерговою задачею.

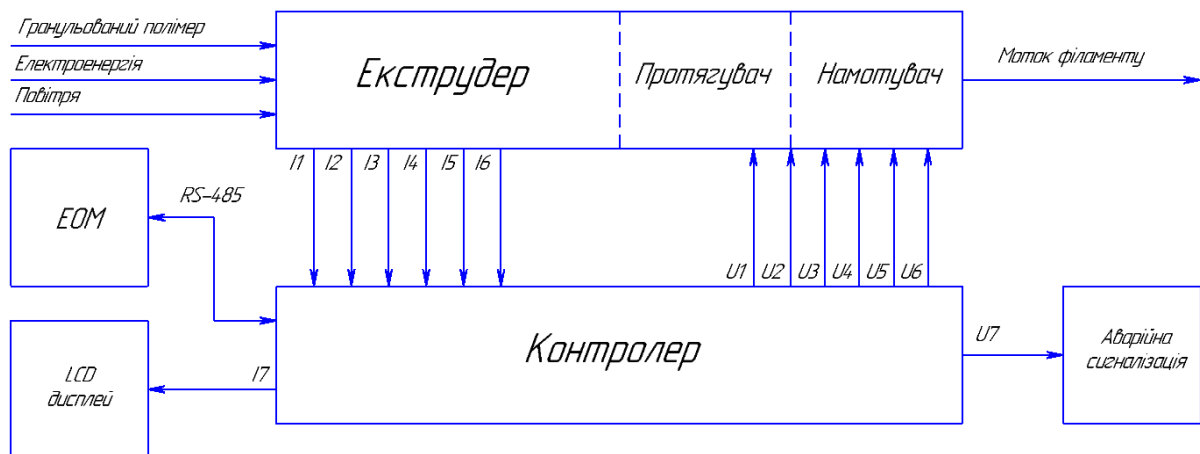


Рисунок 3.1 – Схема інформаційних та матеріальних потоків

Контрольованими параметрами є:

- рівень гранульованого полімеру в завантажувальному бункері (I1);
- частота обертання електродвигуна (I2), що безпосередньо впливає на продуктивність екструдера;
- температури кільцевих нагрівачів (I3, I4), що повинні забезпечувати відповідний температурний режим;
- температура розплаву полімеру (I5) в зоні дозування;
- діаметр філаменту (I6).

Керуючими впливами є:

- частота обертання електродвигуна (U1);
- температура нагрівачів (U2, U3);
- частота обертання протягувача (U4);
- частота обертання намотувача (U5);
- частота обертання охолоджувального вентилятора (U6).

Вся актуальна інформація про хід технологічного процесу передається на операторський LCD дисплей (I7) та ЕОМ і у разі виникнення аварійної ситуації спрацьовує звукова сигналізація (U7).

На рисунку 3.2 приведено розроблений малогабаритний екструдер:



Рисунок 3.2 – Розроблений малогабаритний екструдер

3.2 Перелік застосованих технічних засобів автоматизації

3.2.1 Первинні перетворювачі

Термоелектричний перетворювач

Для контролю температури нагрівачів та розплаву полімеру, температура якого максимально може досягати 300 °С, використано термопару ХА к-типу, що виконана в захисному кожусі (рис. 3.3).

Для нормалізації сигналу з термопари застосовується нормуючий пристрій на базі мікросхеми МАХ6675, що компенсує температуру холодного спаю та видає на мікроконтролер вже оцифрований сигнал.



Рисунок 3.3 – Термоелектричний перетворювач ХА та нормуючий перетворювач МАХ6675

Ключові характеристики термопари в парі з нормуючим пристроєм приведено в табл. 3.1.

Таблиця 3.1 – Ключові параметри термоелектричного перетворювача МАХ6675

Параметр	Значення
Діапазон вимірювальних Т, °С	0 ... 800
Діапазон робочих Т, °С	0...600
Точність, °С	±1.5
Температурне розрішення	0.25
Вихідний сигнал	цифровий, SPI
Живлення, Vcc, В	3...5.5
Струм споживання модулем, мА	до 1,5

Ультразвуковий перетворювач відстані

Для вимірювання рівня грануляту в бункері екструдера, що за своєї внутрішньою глибиною не перевищують 30 см, використано ультразвуковий датчик (принцип ехолокації) HC-SR04 (рис. 3.4), що має досить велику точність (± 0.3 см) та вимірює відстань, залежно від тривалості імпульсу, в діапазоні від 2 до 400 см.



Рисунок 3.4 – Ультразвуковий перетворювач рівня HC-SR04

Ключові характеристики цього датчика приведені в табл. 3.2.

Таблиця 3.2 – Ключові параметри ультразвукового перетворювача HC0SR04

Параметр	Значення
Діапазон відстані, см	2...400
Оглядний кут, °	15
Точність, см	± 0.3
Вимірювальний кут	30
Вихідний сигнал	цифровий
Живлення, Vcc, В	3...5.5
Струм споживання модулем, мА	до 15

Для контролю обертів колекторного електродвигуна використано датчик Холла 49Е (рис. 3.5).

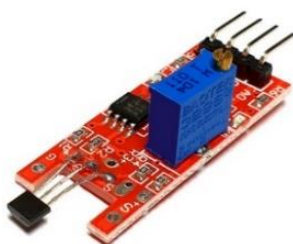


Рисунок 3.5 – Датчик Холла

Ключові параметри датчика Холла зведено в таблицю 3.3.

Таблиця 3.3 – Ключові параметри датчика Холла

Параметр	Значення
Чутливість, мВ/с	1.4
Час відгуку, мкс	3
Тип чутливості до магнітного поля	біполярний
Живлення, V_{cc} , В	3...5.5
Вихідний сигнал:	цифровий та аналоговий

3.2.2 Вторинні прилади

РК-дисплей 2004А

Для відображення інформації про хід технологічного процесу, а також інтерактивної взаємодії з мікроконтролером Arduino використано 4-х строковий рідкокристалічний дисплей 2004А (рис. 3.6), що має на борту додатково розпаяний контролер HD44780 для реалізації спрощеного зв'язку на базі послідовної шини I2C.



Рисунок 3.6 – РК-дисплей 2004А

Ключові характеристики дисплею 2004А приведені на таблиці 3.4.

Таблиця 3.4 – Ключові параметри РК-дисплею 2004А

Параметр	Значення
Відображення	20 символів * 4 рядки
Колір	Синій
Наявність підсвічування	Так
Вихідний сигнал	Цифровий, I2C
Живлення, V_{cc} , В	3...5.5
Струм споживання модулем, мА	до 15

3.2.3 Контролер Arduino MEGA

Для управління роботою екструдера використано платформу Arduino MEGA (рис. 3.7) на базі мікроконтролера ATmega328 [19, 20, 21].

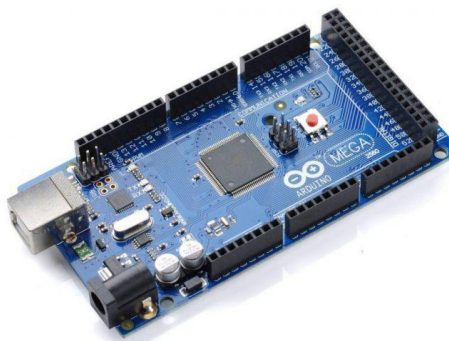


Рисунок 3.7 – Arduino MEGA

Ключові характеристики Arduino MEGA наведені на таблиці 3.5.

Таблиця 3.5 – Характеристики Arduino MEGA

Параметр	Значення
Мікроконтролер	ATmega2560
Кількість цифрових входів	54 (15 підтримують ШІМ)
Кількість аналогових входів	16
Flash-пам'ять, Кб	256 (8 на bootloader)
SRAM, Кб	8
EEPROM, Кб	4
Тактова частота, мГц	16
Живлення, В	7-12
Струм споживання модулем, мА	до 50

Програмування контролера відбувається в середовищі Arduino Studio на C-подібній мові програмування у вигляді скетчу.

Основні критерії вибору мікроконтролеру:

- невисока вартість;
- широкий спектр сумісних технічних засобів;
- оптимальна кількість каналів I/O;
- високорівнева мова програмування с з широким вибором бібліотек.

3.2.4 Виконавчі механізми

Кільцевий нагрівач

Для нагрівання полімеру в зоні дозування використано кільцевий нагрівач з нержавіючої сталі (рис. 3.8).



Рисунок 3.8 – Кільцевий нагрівач

Параметри нагрівача приведені в таблиці 3.6.

Таблиця 3.6 – Ключові параметри кільцевого нагрівача з слюди

Параметр	Значення
Діапазон температур, °С	0...400
Розміри, мм	30*50
Внутрішній діаметр, мм	30
Напруга, В	220
Потужність, Вт	140

Колекторний електродвигун Welling UML5202.01

Для передачі обертового моменту на шнек, через редуктор, використано колекторний електродвигун Welling UML5202.01 (рис. 3.9) потужністю 500Вт.



Рисунок 3.9 – Колекторний електродвигун

Головною перевагою колекторних електродвигунів є простота в регулюванні частоти обертання.

Недоліком даних двигунів є те, що колекторні електродвигуни не призначені для довготривалої роботи і схильні до перегріву, що потребує встановлення додаткового охолодження.

Іншим недоліком цього двигуна є те, що він відносить до високообертальних електродвигунів, частота обертання якого може досягати 11 тис./хв. З чого слідує, що у двигуна номінальний обертовий момент не є постійним на всьому діапазоні регулювання.

Мінімальне число обертів, яке здатен тримати цей двигун, становить 600 об/хв – це значить, що мінімальне число обертання шнека буде 20 об/хв, а максимальне 50 об/хв (на що розрахований редуктор).

Кроковий двигун типорозміру NEMA 17

В якості електродвигунів для протягувача і намотувача було обрано кроковий двигун NEMA 17 (рис. 3.10), модель – SY42STH38.



Рисунок 3.10 – Кроковий двигун NEMA 17

Характеристики NEMA 17 зведено в таблицю 3.7.

Таблиця 3.7 – Ключові параметри крокового двигуна NEMA 17

Параметр	Значення
Кут обертання за один крок, °	1.8
Діаметр валу, мм	5
Довжина валу, мм	24
Струм на обмотку, А	1.7
Робоча напруга, В	5-24
Обертний момент утримання, кг/см	3.5

3.2.5 Інші технічні засоби

Енкодер

Для інтерактивної взаємодії оператора з контролером за допомогою меню використано позиційний енкодер з додатковою функцією кнопки (рис. 3.11).



Рисунок 3.11 – Поворотний енкодер

Твердотільне реле SSR-25DA

Для реалізації двох-позиційного управління роботою кільцевого нагрівача використано твердотільне реле SSR-25DA, що розраховано на комутацію АС навантаження з током до 25 А (рис. 3.12).



Рисунок 3.12 – Твердотільне реле SSR-25DA

Технічні характеристики реле SSR-25DA наведені в таблиці 3.8.

Таблиця 3.8 – Технічні характеристики SSR-25DA

Параметр	Значення
Керуюча напруга, В	3,5...32
Струм спрацювання, мА	7.5
Струм навантаження, А	25
Напруга навантаження, В	АС, 24...220
Швидкість спрацювання, мс	10

Головною перевагою цього реле є наявність гальванічної розв'язки з силовою лінією.

Регулятор обертів TDA1085C

Для управління роботою електродвигуна використано регулятор обертів на базі контролера TDA1085C (рис. 3.13).



Рисунок 3.13 – Регулятор обертів TDA1085C

Дана плата дозволяє регулювати оберти електродвигуна без втрати потужності з зворотним зв'язком від таходатчика або датчика Холла.

Плата дає можливість налаштувати: плавний пуск, максимальні оберти та чуттєвість до навантаження.

Максимальне навантаження – 40А. Плата підтримує до 4 кВт потужності електродвигуни.

Драйвер крокового електродвигуна L298N

Для управління роботою крокового електродвигуна NEMA 17 було використано драйвер L298N (рис. 3.14).



Рисунок 3.14 – Драйвер крокового електродвигуна L298N

Параметри драйвера крокового електродвигуна L298N зведено в таблицю 3.9.

Таблиця 3.9 – Параметри драйвера крокового електродвигуна L298N

Параметр	Значення
Напруга живлення логіки, В	5
Напруга живлення електродвигунів, В	5-35
Максимальна потужність, Вт	20

3.3 Принципова електрична схема управління

Принципова електрична схема (ПЕС) управління є проектним документом, що визначає повний склад електричної частини розроблених САУ/САК і зв'язок між її елементами, даючи детальне уявлення про принцип роботи системи управління екструзією в цілому.

Принципальна схема управління приведені у додатку А.

В розділі розглянуто розроблений малогабаритний екструдер та його технічне забезпечення.

Описано склад, характеристики та призначення технічних засобів автоматизації (первинні та вторинні перетворювачі, мікроконтролер, виконавчі механізми та інші технічні засоби).

Для детального уявлення принципу роботи системи автоматизації екструзійного процесу та комунікаційного зв'язку технічних засобів автоматизації приведено принципові електричні схеми управління та живлення.

РОЗДІЛ 4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Сучасні програмні засоби для розробки Web-додатків

Для розробки системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів використано комплекс технологій, котрі є актуальними для ефективної побудови прогресивного Web-додатку станом на 2021 рік.

Web-додаток складається з двох частин: клієнтської та серверної.

Для клієнтської частини застосовано наступні технології: React, Redux, WebSocket, Recharts, Material-UI.

Для серверної частини використано: Express, Postgress, WebSocket, SerialPort.

4.2.1 Застосування React як платформи для СППР

React – відкрита JavaScript бібліотека для створення Web-інтерфейсів користувача [22], яка покликана вирішувати проблеми часткового оновлення вмісту Web-сторінки, з якими стикаються в розробці односторінкових Web-додатків. Розробляється Facebook, Instagram і спільнотою індивідуальних розробників.

React дозволяє розробникам створювати великі Web-застосунки, які використовують дані, котрі змінюються з часом, без перезавантаження сторінки. Його мета полягає в тому, щоб бути швидким, простим, масштабованим. React обробляє тільки користувацький інтерфейс у Web-додатках. Це відповідає View у шаблоні модель-вид-контролер (Model-View-Controller) та може бути використаний у поєднанні з іншими JavaScript бібліотеками.

Як бібліотеку інтерфейсу користувача React найчастіше використовують разом з іншими бібліотеками, такими як Redux.

Особливості React бібліотеки:

а) одностороння передача даних – властивості передаються в рендерер компоненту, як властивості html тегу. Компонент не може напряму змінювати властивості, що йому передані, але може їх змінювати через callback функції. Такий механізм називають «властивості донизу, події нагору»;

б) віртуальний DOM – React підтримує віртуальний DOM, а не покладається виключно на DOM браузера. Це дозволяє бібліотеці визначити, які частини DOM змінилися, порівняно зі збереженою версією віртуального DOM, і таким чином визначити, як найефективніше оновити DOM браузера. Таким чином програміст працює зі сторінкою, вважаючи, що вона оновлюється вся, але бібліотека самостійно вирішує, які компоненти сторінки треба оновити;

в) JSX – компоненти React зазвичай написані на JSX. Код написаний на JSX компілюється у виклики методів бібліотеки React. Розробники можуть так само зручно писати програмний код Web-додатку на чистому JavaScript;

г) не лише рендеринг HTML в браузері – React використовують не лише для рендерингу HTML в браузері. Наприклад, Facebook має динамічні графіки які рендеряться в теги <canvas>, Netflix та PayPal використовують ізоморфне завантаження для рендерингу ідентичного HTML на сервері та клієнті;

д) методи життєвого циклу — це різні методи, які вбудовуються за допомогою ReactJS. Вони дозволяють розробнику обробляти дані в різних точках життєвого циклу програми React:

1. `shouldComponentUpdate` — це метод життєвого циклу, який каже Javascript оновити компонент, використовуючи логічні змінні;

2. `componentWillMount` — це метод життєвого циклу, який каже Javascript налаштувати певні дані перед монтуванням компонентів (вставлення у віртуальний DOM);

3. `componentDidMount` — це метод життєвого циклу, подібний до компонента `WillMount`, за винятком того, що він працює після методу `render`, і може використовуватися для додавання JSON-даних, а також для визначення властивостей та станів;

4. `render` є найважливішим методом життєвого циклу, необхідним у будь-якому компоненті. Метод `render` — це те, що з'єднується з JSX і відображати власний JSX;

е) вкладені елементи – кілька елементів на одному рівні повинні бути загорнутими в один елемент контейнера, наприклад елемент `<div>`;

є) атрибути – JSX надає ряд атрибутів елементів, призначених для відображення тих, що надаються у форматі HTML. Користувацькі атрибути також можуть бути передані компоненту. Всі атрибути будуть отримані компонентом як реквізит;

ж) вирази JavaScript – це типові вирази JavaScript, котрі можна використовувати в JSX з фігурними дужками `{}`:

```
<h1>{10 + 1}</h1>
```

Приклад, наведений вище, відобразатиметься так:

```
<h1>11</h1>
```

з) умовні вирази: вираз типу `if-else` не можуть бути використані всередині JSX, але замість них можуть використовуватися умовні вирази. У прикладі нижче буде показано як рядок `'true'`, оскільки і дорівнює 1:

```
class App extends React.Component {
  render() {
    const i = 1;
    return (
      <div>
        <h1>{ i === 1 ? 'true' : 'false' }</h1>
      </div>
    );
  }
}
```

и) функції та JSX можна використовувати в умовних виразах:

```
class App extends React.Component {
  render() {
    const sections = [1, 2, 3];
    return (
      <div>
        {
          sections.length > 0
            ? sections.map(n => <div>Section {n}</div>)
            : null
        }
      </div>
    );
  }
}
```

Програмний код, написаний у JSX, потребує перетворення за допомогою такого інструменту, як Babel, для того, щоб його могли зрозуміти Web-браузери. Ця обробка, як правило, виконується під час процесу збірки, перш ніж програма буде запущена.

4.2.2 Централізоване сховище для зберігання даних Redux

Redux – відкрита JS бібліотека призначена для управління станом Web-додатків [23, 24]. Найчастіше використовується разом з React для побудови інтерфейсів користувача.

Redux дотримується концепції “єдиного джерела істини” – зберігає стан всього Web-додатку в дереві об’єктів в одному сховищі.

Одне дерево станів полегшує налагодження або перевірку програми, а також дозволяє зберігати стан вашого застосунку для прискорення циклу розробки.

Життєвий цикл Redux приведено на рисунку 4.1.

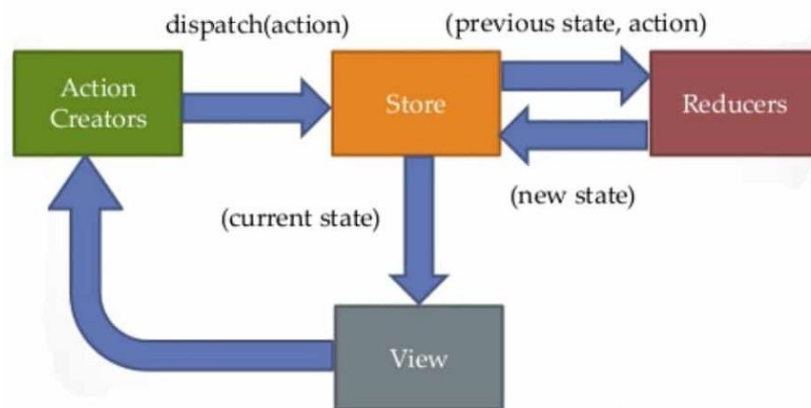


Рисунок 4.1 – Життєвий цикл Redux

Store (Сховище) – це об'єкт, який з'єднує екшени, які представляють факт того, що "щось сталося" і редюсери, які оновлюють стан (state) відповідно до цих екшенів разом.

Store містить стан додатку (application state), надаючи доступ до стану за допомогою `getState()` функції, випускає оновлення стану за допомогою `dispatch(action)` екшенів, обробляє скасування реєстрації слухачів за допомогою функції, що повертається `subscribe(listener)`.

State (Стан) призначений тільки для зчитування. Єдиний спосіб змінити стан — це виокремити дію, об'єкт, що описує те, що сталося. Це гарантує, що ні перегляди, ні зворотні виклики мережі ніколи не будуть змінювати стан. Натомість вони виражають тільки намір це зробити. Всі зміни — централізовані і відбуваються одна за іншою у чіткій послідовності. Оскільки дії є простими об'єктами, вони можуть бути зареєстровані, серіалізовані, збережені та пізніше відтворені для налагодження або тестування.

Action (Екшен) - це структура, яка передає дані з застосуванням в стан. Він є єдиними джерелами інформації для стану. Екшенів може бути безліч й вони можуть бути в обгорнуті в чисті функції (Action Creators) задля зручності та зменшення дублювання коду. Redux відправляє їх в Store, використовуючи метод `store.dispatch()`.

Типовий action creator в системі управління екструзією полімерних матеріалів, що відповідає за збереження стану поточної заданої температури (завдання) для PID-регулятора:

```
function setTemperature(value) {
  return {
    type: 'PID/SetTemperature',
    payload: value
  }
}
```

Редьюсери (reducers) — це лише чисті функції, які приймають попередній стан і дію тобто state, і повертають наступний стан. Редьюсери визначають, як стан додатки змінюється у відповідь на екшени (action), які відправлені в стан.

Екшени тільки описують, що сталося, але не описують, як змінюється стан додатка. Нижче приведено лістинг PID-редьюсера, котрий призначений для зміни стану заданої температури (завдання) PID-регулятора:

```
const initialState = {
  temperature: 0,
  point: 160
}

function PIDReducer(state = initialState, action) {
  switch (action.type) {
    case 'PID/SetTemperature':
      return {...state, temperature: action.payload}
    default:
      return state
  }
}
```

В процесі розробки редьюсери можуть бути розділені на дрібніші редьюсери, які управляють певними частинами дерева станів. Оскільки редьюсери – це лише функції, можливо контролювати порядок їх надсилання, передавати додаткові дані або навіть створювати повторювані редьюсери для звичайних завдань, таких як розбиття на сторінки.

4.2.3 Забезпечення нерозривного з'єднання між клієнтом і сервером за допомогою протоколу WebSocket

WebSocket – протокол зв'язку поверх TCP-з'єднання, призначений для обміну повідомленнями між браузером і WEB-сервером в режимі реального часу [24, 25, 26].

Технологія WebSocket розроблена для використання в Web-браузерах і Web-серверах, але вона може бути використана для будь-якого клієнтського або серверного додатку.

Протокол WebSocket – це незалежний протокол, заснований на протоколі TCP. Він робить можливим більш тісну взаємодію між браузером і Web-додатком (рис. 4.2), сприяючи поширенню інтерактивного вмісту і створення додатків реального часу.

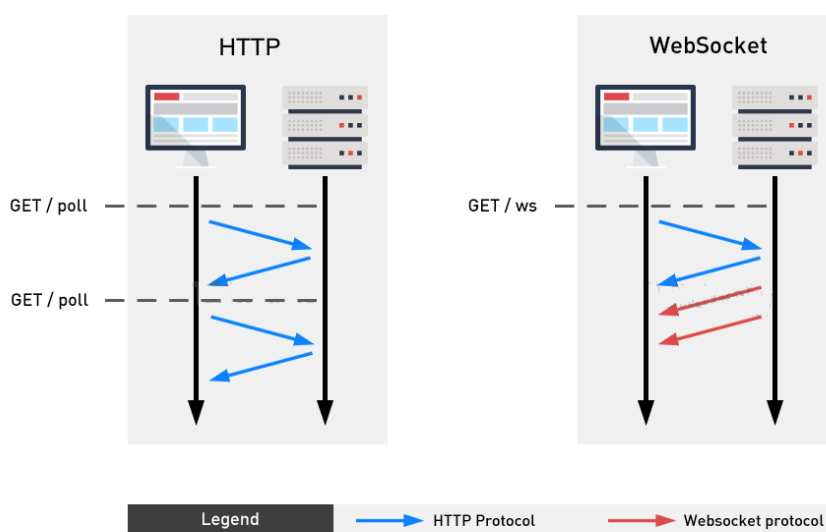
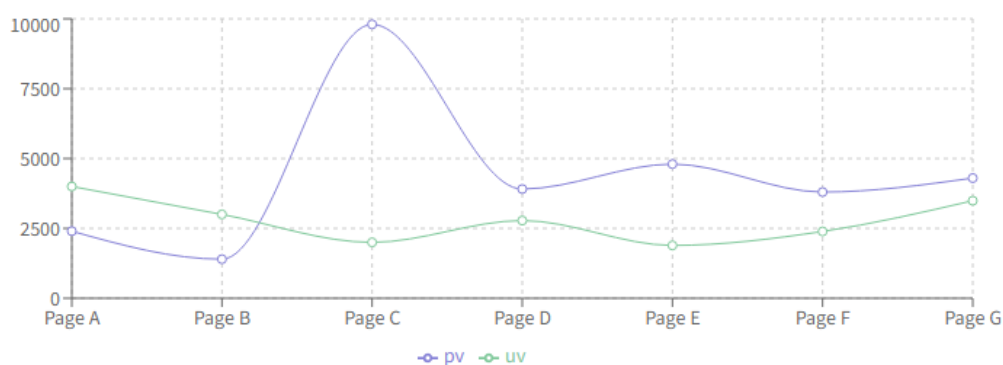


Рисунок 4.2 – WebSocket в порівнянні зі звичайним HTTP з'єднанням

4.2.4 Інтеграція динамічних графіків за допомогою Recharts

Recharts – бібліотека [28], що призначена для зручної інтеграції графіків різного типу в Web-додатках написаних на React [22]. Бібліотека містить великий перелік графіків (лінійний, круговий, радіальний) у вигляді react-компонентів (рис. 4.3).

LineChart



```

<LineChart width={730} height={250} data={data}
  margin={{ top: 5, right: 30, left: 20, bottom: 5 }}>
  <CartesianGrid strokeDasharray="3 3" />
  <XAxis dataKey="name" />
  <YAxis />
  <Tooltip />
  <Legend />
  <Line type="monotone" dataKey="pv" stroke="#8884d8" />
  <Line type="monotone" dataKey="uv" stroke="#82ca9d" />
</LineChart>

```

Рисунок 4.3 – Лінійний графік бібліотеки Recharts

4.2.5 Застосування графічних компонентів бібліотеки Material UI

Бібліотека [29] стилізованих та уніфікованих графічних компонентів (рис. 4.4), призначення для React [22]: кнопки, тумблери, модальні вікна, панелі, індикатори і т.д. Бібліотека дає змогу налаштовувати будь-який компонент під потреби розробки та швидко інтегрувати його у Web-додаток.

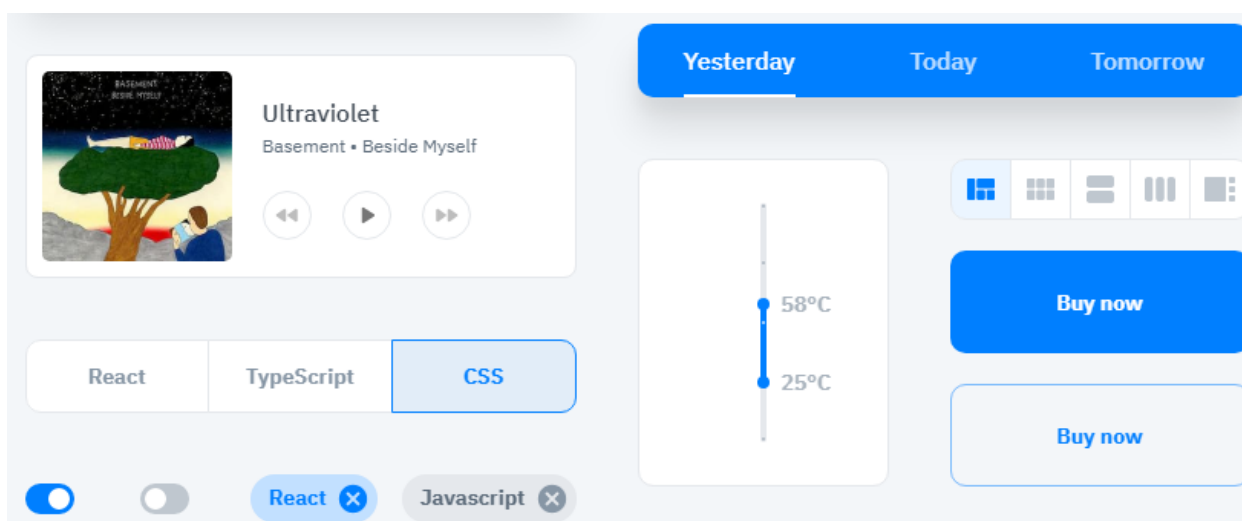


Рисунок 4.4 – Графічні компоненти бібліотеки Material UI

4.2.6 Розгортання Web-сервера за допомогою Express.js

Це мінімалістичний та гнучкий Web-фреймворк [30, 31] для побудови серверної частини Web-додатків, мовою JavaScript. Фреймворк дає змогу швидко розгорнути Web-сервер та обробляти вхідні запити.

Архітектура Web-сервера на базі Express.js приведена на рисунку 4.5.

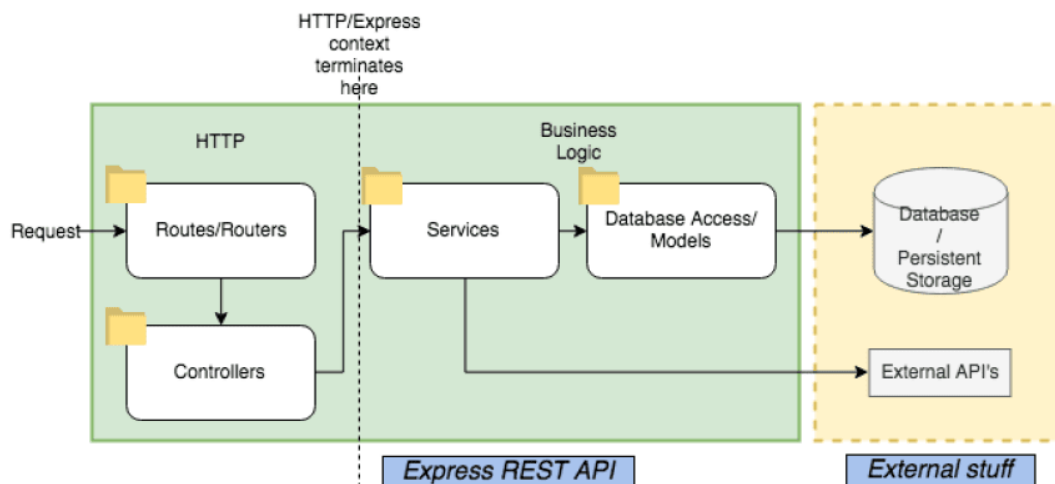


Рисунок 4.5 – Архітектура Web-сервера на базі Express.js

Згідно рис. 4.5 HTTP-запити, що надходять з боку клієнта на сервер Express.js обробляються контролерами (Controllers), котрі взаємодіють з сервісами (Services), що безпосередньо виконують відповідні дії на запит (збереження, оновлення, видалення даних у базі даних (Database) чи звернення до зовнішніх сервісів(External API's)). Виконавши необхідні дії, сервіс повертає результат обробки запиту назад до контролеру, а той, в свою чергу, формує відповідь клієнту.

4.2.7 Реляційна база даних PostgreSQL

PostgreSQL – Об'єктно-орієнтована система управління базами даних мовою SQL [32, 33].

SQL (Structured Query Language) – структурна мова запитів, що дає змогу зберігати, маніпулювати і отримувати дані з реляційних баз даних.

На рисунку 4.6 приведено структуру мови SQL.

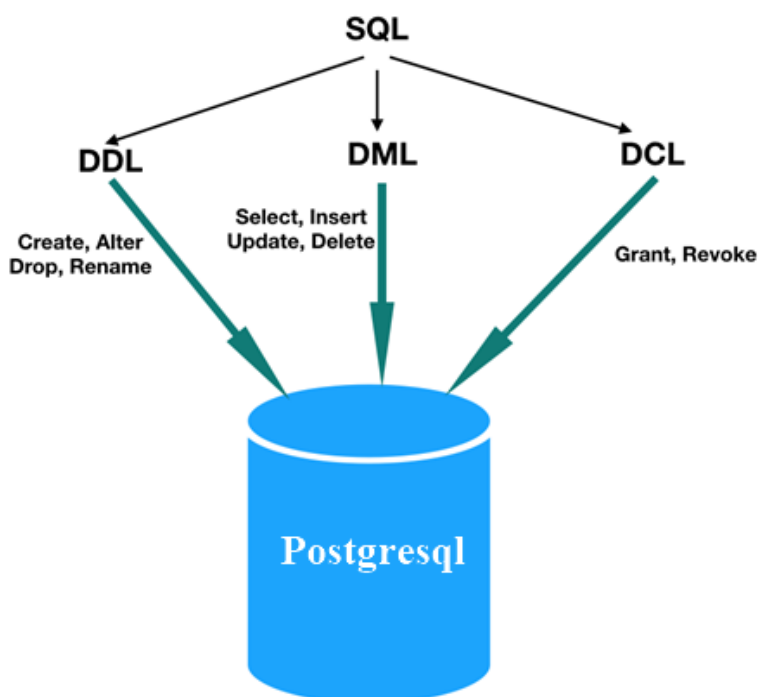


Рисунок 4.6 – Мова SQL та її структура в розрізі PostgreSQL

DDL (Data definition language) – частина мови SQL призначена для виконання запитів з маніпуляцією структурою бази даних (створення бази даних, створення, перейменування, видалення, очищення таблиць).

DML (Data manipulation language) – особливість мови SQL, відповідальна за операції над даними таблиці – вибірка (Select), вставка (Insert), видалення (Delete) та оновлення (Update).

DCL (Data control language) – складова мови SQL призначена для встановлення контролю доступу до даних, що зберігаються в базі даних, тобто встановлення прав користувачів.

В розробленому Web-додатку база даних PostgreSQL використовується для збереження даних авторизованих користувачів (логіни, паролі, права доступу), хронологію зміни технологічних параметрів екструзії, поточні параметри агрегатів екструзійної лінії та вихідні дані для системи підтримки прийняття рішення по управлінню екструзією полімерних матеріалів.

4.3 Архітектура розробленого Web-додатку

На рисунку 4.7 представлено архітектуру Web-додатку системи управління екструзією полімерних матеріалів, куди буде інтегровано систему підтримки прийняття рішень.

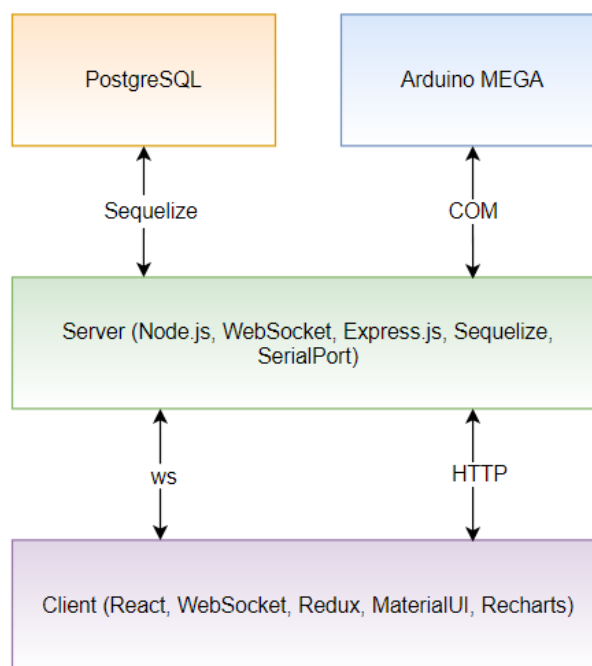


Рисунок 4.7 – Архітектура WEB-додатку

Web-додаток складається з клієнтської (Client) та серверної (Server) частини.

Server виконує наступний перелік функцій:

- обробляє дані з мікроконтролера Arduino та дає на нього команди, обмін відбувається через COM-з'єднання [34];
- зберігає дані, що надійшли з мікроконтролера, в базу даних PostgreSQL через sequelize інтерфейс;
- обробляє HTTP запити, що надходять з боку клієнта (авторизація);
- відправляє поточні технологічні параметри екструзії на клієнт через WebSocket в режимі реального часу.

Client виконує наступний перелік функцій:

- обробляє поточні технологічні параметри екструзії, що надходять з боку сервера через WebSocket;
- відображає мнемосхему екструдера з поточним технологічним станом;
- відображає вікно задання технологічних параметрів екструзії (контролери, регулятори, реле і т.д.);
- відображає вікно оптимізації.

4.4 Розробка серверної частини

Код серверної частини складається з наступних частин:

1. Реєстрація Web-серверу за допомогою Express.js:

server/index.js – головний файл сервера

```
require('dotenv').config()

const SerialPort = require('serialport')
const Readline = require('@serialport/parser-readline')
const express = require('express')
const sequelize = require('./db')
const cors = require('cors')
const router = require('./routes')
const ws = require('ws')

const PORT = process.env.PORT || 5000

sequelize.authenticate()
sequelize.sync()

const app = express()
app.use(cors())
app.use(express.json())
app.use('/api', router)

const server = app.listen(PORT, () => {
  console.log(`Server started on port ${PORT}`)
})
```

2. Реєстрація Web-сокет з'єднання:

server/index.js – головний файл сервера

```
const wsServer = new ws.Server({
  server
}, () => {
  console.log('wsServer started on port: ' + PORT)
})
```

```

function setInitParams(ws) {
  ws.send(JSON.stringify({
    type: 'PIDRegulator1/setInitialParams',
    payload: {
      status: 'OK',
      active: false,
      point: 150,
      Kp: 4.2,
      Ki: 0.6,
      Kd: 0
    }
  })))
}

//.....//
}

const broadcastConditionallyMessage = (broadcastMessage, parsedMessage, condition, delay = 2000) => {
  if (parsedMessage.type === condition) {
    setTimeout(() => {
      broadcastMessage(parsedMessage)
    }, delay)
  }
}

wsServer.on('connection', function connection(ws) {
  setInitParams(ws)

  ws.on('message', function (message) {
    const parsedMessage = JSON.parse(message)

    //.....//

    if (parsedMessage.type === 'PIDRegulator1/setParams') {
      const serialMessage = SerialProtocol.PIDRegulatorSetParams(parsedMessage.type,
        parsedMessage.payload)
      port.write(serialMessage, function (err) {
        if (err) {
          return broadcastMessage({type: 'PIDRegulator1/setStatus', payload: 'ERROR'})
        }
        broadcastMessage(parsedMessage)
      })
    }
  })

  //.....//

})

function broadcastMessage(message) {
  wsServer.clients.forEach(client => {
    client.send(JSON.stringify(message))
  })
}

```

3. Створення контролерів:

`server/controllers/authController.js` – контролер авторизації користувача

```
const User = require('../models/User')

module.exports = {
  login: async (req, res) => {
    try {
      const {email, password} = req.body
      const user = await User.findOne({where: {email}})

      if (user && password === user.password) {
        res.json({user})
      }

      res.status(401).json({message: 'Bad credentials'})
    } catch (e) {
      res.sendStatus(500)
    }
  }
}
```

4. Створення та реєстрація HTTP-маршрутів, прив'язка їх до відповідних контролерів:

`server/routes/authRouter.js` – реєстрація маршруту авторизації

```
const express = require('express')
const authController = require('../controllers/authController')
const router = new express()

router.post('/login', authController.login)

module.exports = router
```

`server/routes/index.js` – глобальна реєстрація маршрутів

```
const express = require('express')
const router = new express()

const authRouter = require('./authRouter')

router.use('/auth', authRouter)

module.exports = router
```

5. Створення сервісу для роботи з базою даних PostgreSQL:

`services/db.js` – сервіс для роботи з базою даних PostgreSQL

```
const {Sequelize} = require('sequelize')

module.exports = new Sequelize(
  process.env.DB_NAME,
  process.env.DB_USER,
  process.env.DB_PASSWORD,
  {
    dialect: 'postgres',
    host: process.env.DB_HOST,
```

```

    port: process.env.DB_PORT
  }
)

```

server/models/User.js – модель користувача в базі даних

```

const db = require('./db')
const { DataTypes } = require('sequelize')

module.exports = db.define('user', {
  id: { type: DataTypes.INTEGER, primaryKey: true, autoIncrement: true },
  email: { type: DataTypes.STRING, unique: true },
  password: { type: DataTypes.STRING },
  role: { type: DataTypes.STRING, defaultValue: 'USER' }
})

```

6. Інтеграція обробки COM-з'єднання з мікроконтролером Arduino:

```

const port = new SerialPort('COM3')

const parser = port.pipe(new Readline({ delimiter: '\r\n'}))

const reduxActionParse = (type, payload) => {
  return {
    type, payload
  }
}

parser.on('data', function (data) {
  try {
    console.log({ from: 'PLC', data })

    const [address, action, value] = data.split('/')

    switch (address) {
      case 'sensors':
        broadcastMessage(reduxActionParse(`${address}/${action}`, +value))
        break
      case 'imitHeatObj1':
        broadcastMessage(reduxActionParse(`${address}/${action}`, +value))
        break
      case 'PIDRegulator1':
        if (action === 'setSignal') {
          broadcastMessage(reduxActionParse(`${address}/${action}`, +value))
        }
        break
      case 'imitHeatObj2':
        broadcastMessage(reduxActionParse(`${address}/${action}`, +value))
        break
      case 'PIDRegulator2':
        if (action === 'setSignal') {
          broadcastMessage(reduxActionParse(`${address}/${action}`, +value))
        }
        break
      default:
        break
    }
  } catch (e) {
    console.log(e)
  }
})

```



```
const SerialProtocol = {
  PIDRegulatorSetParams(type, payload) {
    // [PIDRegulator1/setParams/active;point;Kp;Ki;Kd]
    const { active, point, Kp, Ki, Kd } = payload
    return `|${type}/${+active};${point};${Kp};${Ki};${Kd}|`
  }
}
```

4.5 Розробка клієнтської частини

Код клієнтської частини складається з наступних частин:

1. Ініціалізація Redux сховища

`client/src/redux/slices/PIDRegulator1.js` – об'єкт, що відповідає за зберігання та управління станом PID-регулятора

```
import { createSlice } from '@reduxjs/toolkit'

const initialState = {
  status: 'UNKNOWN', // 'OK', 'SENDING', 'ERROR'
  active: false,
  point: null,
  Kp: null,
  Ki: null,
  Kd: null,
  signal: null
}

const PIDRegulator1Slice = createSlice({
  name: 'PIDRegulator1',
  initialState,
  reducers: {
    setStatus: (state, action) => {
      state.status = action.payload
    },
    setSignal: (state, action) => {
      state.signal = action.payload
    },
    setParams: (state, action) => {
      const { active, point, Kp, Ki, Kd } = action.payload
      state.status = 'OK'
      state.active = active
      state.point = point
      state.Kp = Kp
      state.Ki = Ki
      state.Kd = Kd
    },
    setInitialParams: (state, action) => {
      return action.payload
    }
  }
})

export default PIDRegulator1Slice.reducer

export const PIDRegulator1Selectors = {
  params: ({PIDRegulator1: { active, point, Kp, Ki, Kd, status }}) => ({ active, status, point, Kp, Ki, Kd}),
```

```

    signal: ({PIDRegulator1}) => PIDRegulator1.signal
  }

export const {setStatus, setParams} = PIDRegulator1Slice.actions

```

`client/src/redux/slices/wsSlice.js` – Об’єкт, що відповідає за зберігання стану `WebSocket` з’єднання

```

import {createSlice} from '@reduxjs/toolkit'

export const WebSocketConnect = host => ({type: 'WS_CONNECT', host})
export const WebSocketDisconnect = () => ({type: 'WS_DISCONNECT'})
export const wsMessage = (payload) => ({type: 'WS_MESSAGE', payload})

const initialState = {
  connection: true,
  connected: false
}

const wsSlice = createSlice({
  name: 'ws',
  initialState,
  reducers: {
    setConnection: (state, action) => {
      state.connection = action.payload
    },
    setConnected: (state, action) => {
      state.connected = action.payload
    }
  }
})

export default wsSlice.reducer

export const wsSelectors = {
  connection: ({ws}) => ws.connection,
  connected: ({ws}) => ws.connected
}

export const {setConnected, setConnection} = wsSlice.actions

```

`client/src/redux/reducers/authReducer.js` – об’єкт, що відповідає за збереження та зміну стану авторизації користувача

```

import {createReducer, createAsyncThunk} from "@reduxjs/toolkit";
import * as API from '../api/axiosAPI'

export const loginAsync = createAsyncThunk(
  'auth/login',
  async (credentials, {rejectWithValue, fulfillWithValue}) => {
    try {
      const {data: {user}} = await API.auth.login(credentials)
      return fulfillWithValue(user)
    } catch (e) {
      rejectWithValue(e.response.data)
    }
  }
)

```

```

const initialState = {
  authenticated: false,
  user: null,
  ui: {
    fetching: false,
    error: null
  }
}

const authReducer = createReducer(initialState, (builder) => {
  builder
    .addCase(loginAsync.pending, (state) => {
      state.ui.fetching = true
      state.ui.error = null
    })
    .addCase(loginAsync.fulfilled, (state, action) => {
      state.user = action.payload
      state.authenticated = true
      state.ui.fetching = false
    })
    .addCase(loginAsync.rejected, (state, action) => {
      state.authenticated = false
      state.ui.fetching = false
      state.ui.error = action.payload
    })
})

export default authReducer

```

`client/src/redux/store.js` – створення Redux-сховища та реєстрація всіх його компонентів

```

import {configureStore} from '@reduxjs/toolkit'
import authReducer from './reducers/authReducer'
import sensorsSlice from './slices/sensorsSlice'
import wsReducer from './slices/wsSlice'
import WebSocketMiddleware from './middlewares/WebSocketMiddleware'
import PIDRegulator1Reducer from './slices/PIDRegulator1Slice'
import PIDRegulator2Reducer from './slices/PIDRegulator2Slice'
import engineControllerReducer from './slices/engineController'
import stepperController1Reducer from './slices/stepperController1'
import stepperController2Reducer from './slices/stepperController2'
import coolingControllerReducer from './slices/coolingController'
import relay1Reducer from './slices/relay1Slice'
import relay2Reducer from './slices/relay2Slice'
import extruderReducer from './slices/extruderSlice'

export default configureStore({
  reducer: {
    auth: authReducer,
    ws: wsReducer,
    sensors: sensorsSlice,
    PIDRegulator1: PIDRegulator1Reducer,
    PIDRegulator2: PIDRegulator2Reducer,
    engineController: engineControllerReducer,
    stepperController1: stepperController1Reducer,
    stepperController2: stepperController2Reducer,
    coolingController: coolingControllerReducer,
    relay1: relay1Reducer,
    relay2: relay2Reducer,

```

```

    extruder: extruderReducer
  },
  middleware: (getDefaultMiddleware) => getDefaultMiddleware().concat(WebSocketMiddleware)
})

```

2. Створення WebSocket middleware (посередника)

client/src/middlewares/WebSocketMiddleware.js – посередник для обробки WebSocket з'єднання

```

import { setConnected, setConnection } from '../redux/slices/wsSlice'
const WebSocketMiddleware = () => {
  let socket = null

  const onOpen = store => event => {
    store.dispatch(setConnection(false))
    store.dispatch(setConnected(true))
  }

  const onClose = store => event => {
    store.dispatch(setConnection(true))
    store.dispatch(setConnected(false))
  }

  const onMessage = store => event => {
    const payload = JSON.parse(event.data)
    store.dispatch(payload)
  }

  const onError = store => event => {
    console.log('Socket error!')
  }

  return store => next => action => {
    switch (action.type) {
      case 'WS_CONNECT':
        if (socket !== null) {
          socket.close()
        }
        socket = new WebSocket(action.host)
        socket.onopen = onOpen(store)
        socket.onclose = onClose(store)
        socket.onmessage = onMessage(store)
        socket.onerror = onError(store)
        break
      case 'WS_DISCONNECT':
        if (socket !== null) {
          socket.close()
        }
        socket = null
        break
      case 'WS_MESSAGE': {
        socket.send(JSON.stringify(action.payload))
        break
      }
      default:
        return next(action)
    }
  }
}

```

```

    }
  }

export default WebSocketMiddleware()

```

3. Інтеграція Redux-сховища у React-додаток

client/src/index.js – кореневий файл React-додатку

```

import React from 'react'
import ReactDOM from 'react-dom'
import '@fontsource/roboto'
import './index.css'
import App from './App'
import {Provider} from 'react-redux'
import store from './redux/store'

ReactDOM.render(
  <React.StrictMode>
    <Provider store={store}>
      <App/>
    </Provider>
  </React.StrictMode>,
  document.getElementById('root')
)

```

4. Реєстрація маршрутів React-додатку

client/src/App.js – опис маршрутів React-додатку

```

import {BrowserRouter as Router, Route, Switch} from 'react-router-dom'
import './App.css'
import Login from './pages/Login'
import MainPage from './pages/MainPage'

function App({user, handleLoginAction}) {
  return (
    <Router>
      <Switch>
        <Route path={'/login'}>
          <Login user={user} handleLogin={handleLoginAction}/>
        </Route>
        <Route path={'/'}>
          <MainPage/>
        </Route>
      </Switch>
    </Router>
  )
}

export default App

```

5. Створення сторінки авторизації користувачів

client/src/pages/Login.js – сторінка авторизації користувачів

```

import {Button, CircularProgress, Divider, Paper, TextField, Typography} from '@material-ui/core'
import {useFormik} from "formik";
import loginSchema from "../validation/loginSchema";

```

```

import withShowPasswordIcon from "../hoc/withShowPasswordIcon";
import {useHistory} from 'react-router-dom'
import {useDispatch, useSelector} from "react-redux";
import {loginAsync} from "../redux/reducers/authReducer";

const PasswordField = withShowPasswordIcon(TextField)

const Login = () => {

  const history = useHistory()
  const {authenticated, error} = useSelector(state => state.auth)
  const dispatch = useDispatch()

  useEffect(() => {
    if (authenticated) {
      history.push('/')
    }
  }, [authenticated, history])

  const formik = useFormik({
    initialValues: {
      email: "",
      password: "",
    },
    validationSchema: loginSchema,
    onSubmit: async (values) => {

      await dispatch(loginAsync(values))

    }
  })

  return (
    <div style={{
      height: '100vh', display: 'flex',
      background: 'linear-gradient(to right, #8360c3, #2ebf91)',
      justifyContent: 'center', alignItems: 'center'
    }}>
      <form onSubmit={formik.handleSubmit}>
        <Paper variant={'outlined'}
          style={{
            display: 'flex',
            height: 'max-content',
            flexDirection: 'column',
            padding: '10px',
            width: '300px'
          }}>
          <Typography align={'center'} variant={'h5'}>Login</Typography>
          <Divider/>
          <div style={{
            display: 'flex',
            flexDirection: 'column'
          }}>
            {error && <div style={{'background': '#f44336', borderRadius: '4px', padding: '4px', marginTop:
'12px'}}>
              <Typography style={{color: 'white'}} variant={'caption'}>{error}</Typography>
            </div>
            <TextField label='Email'
              style={{marginTop: '18px'}}
              variant='outlined'
              type='email'
            </TextField>
          </div>
        </form>
      </div>
    </div>
  )
}

```

```

      autoComplete={'email'}
      size={'small'}
      name={'email'}
      value={formik.values.email}
      onChange={formik.handleChange}
      error={formik.touched.email && Boolean(formik.errors.email)}
      helperText={formik.touched.email && formik.errors.email}
    />
    <PasswordField label='Password'
      style={{marginTop: '16px'}}
      variant='outlined'
      type={'password'}
      autoComplete={'current-password'}
      size={'small'}
      name={'password'}
      value={formik.values.password}
      onChange={formik.handleChange}
      error={formik.touched.password && Boolean(formik.errors.password)}
      helperText={formik.touched.password && formik.errors.password}/>
    <Button variant={'contained'}
      size={'small'}
      style={{marginTop: '18px'}}
      color={'primary'}
      type={'submit'}>
      Sign in {formik.isSubmitting &&
    <CircularProgress style={{color: 'white', marginLeft: '10px'}} size={20}/>
    </Button>
  </div>
</Paper>
</form>
</div>
)
}

export default Login

```

6. Створення розмітки головної сторінки

client/src/pages/MainPage.js – розмітка головної сторінки Web-додатку

```

import {CircularProgress, Typography} from '@material-ui/core'
import {useEffect} from 'react'
import {useDispatch, useSelector} from 'react-redux'
import {Route, Switch} from 'react-router-dom'
import Header from '../components/Header'
import {WebSocketConnect, wsSelectors} from '../redux/slices/wsSlice'
import OptimizationTab from './OptimizationTab'
import PreviewTab from './PreviewTab'
import RTMTab from './RTMTab'

function MainPage() {
  const dispatch = useDispatch()
  const connection = useSelector(wsSelectors.connection)

  useEffect(() => {
    setTimeout(() => {
      dispatch(WebSocketConnect('ws://localhost:5000'))
    }, 3000)
  }, [dispatch])

  const onLogout = () => {

```

```

    }

    if (connection) {
      return (
        <div style={{
          height: '100vh', display: 'flex',
          background: 'linear-gradient(to right, #8360c3, #2ebf91)',
          justifyContent: 'center',
          alignItems: 'center',
          flexDirection: 'column'
        }}>
          <CircularProgress size={40} style={{ color: 'white' }}/>
          <Typography variant={'h6'} style={{ color: 'lightgray', marginTop: '16px' }}>
            Web-socket connection
          </Typography>
        </div>
      )
    }

    return (
      <>
        <Header onLogout={onLogout}/>
        <main>
          <Switch>
            <Route path={'/'} exact>
              <RTMTab/>
            </Route>
            <Route path={'/preview'}>
              <PreviewTab/>
            </Route>
            <Route path={'/optimization'}>
              <OptimizationTab/>
            </Route>
          </Switch>
        </main>
      </>
    )
  }

  export default MainPage

```

7. Розроблена вкладки мнемосхеми екструдера

client/src/pages/PreviewTab.js – вкладка мнемосхеми екструдера

```

import HeatOptimizationCard from '../components/MiniCards/Optimization/HeatOptimizationCard'
import extruderSwg from '../extruder.svg'
import Engine from '../components/Extruder/Engine'
import EngineOptimizationCard from '../components/MiniCards/Optimization/EngineOptimizationCard'

import BunkerMiniCard from '../components/MiniCards/BunkerMiniCard'
import CoolingMiniCard from '../components/MiniCards/CoolingMiniCard'
import EngineMiniCard from '../components/MiniCards/EngineMiniCard'
import FanMiniCard from '../components/MiniCards/FanMiniCard'
import Heater1MiniCard from '../components/MiniCards/Heater1MiniCard'
import Heater2MiniCard from '../components/MiniCards/Heater2MiniCard'
import MeasurerMiniCard from '../components/MiniCards/MeasurerMiniCard'
import RewinderMiniCard from '../components/MiniCards/RewinderMiniCard'
import StretcherMiniCard from '../components/MiniCards/StretcherMiniCard'
import InnerExtruderSensorsMiniCard from '../components/MiniCards/InnerExtruderSensorsMiniCard'

```



```

import Cooling from '../components/Extruder/Cooling'
import Corpus from '../components/Extruder/Corpus'
import Material from '../components/Extruder/Material'
import Screw from '../components/Extruder/Screw'

function PreviewTab() {
  const commonMiniCardsTop = 40
  return (
    <div style={{
      height: '700px',
      display: 'flex',
      alignItems: 'center',
      justifyContent: 'center',
      position: 'relative'
    }}>
      <img src={extruderSwg} alt={'extruder-swg'}/>

      <EngineMiniCard style={{left: 270, top: commonMiniCardsTop}}/>
      <BunkerMiniCard style={{left: 465, top: commonMiniCardsTop}}/>
      <Heater1MiniCard style={{left: 700, top: commonMiniCardsTop}}/>
      <Heater2MiniCard style={{left: 850, top: commonMiniCardsTop}}/>
      <FanMiniCard style={{left: 1000, top: commonMiniCardsTop}}/>
      <MeasurerMiniCard style={{left: 1160, top: commonMiniCardsTop}}/>
      <StretcherMiniCard style={{left: 1320, top: commonMiniCardsTop}}/>
      <RewinderMiniCard style={{left: 1525, top: commonMiniCardsTop}}/>

      <InnerExtruderSensorsMiniCard style={{left: 850, top: 540}}/>
      <CoolingMiniCard style={{left: 390, top: 560}}/>

      <Engine style={{left: 320, top: 400}}/>
      <Screw style={{left: 650, top: 400}}/>
      <Corpus style={{left: 510, top: 400}}/>
      <Material style={{left: 480, top: 210}}/>
      <Cooling style={{left: 434, top: 470}}/>

      <HeatOptimizationCard style={{left: 700, top: 230}}/>
      <EngineOptimizationCard style={{left: 70, top: 350}}/>
    </div>
  )
}

export default PreviewTab

```

8. Розроблена вкладка монітору реального часу

client/src/pages/RTMTab.js – компонент вкладки “Монітор реального часу”

```

import Bunker from '../components/Cards/Bunker'
import Cooling from '../components/Cards/Cooling'
import Engine from '../components/Cards/Engine'
import Fan from '../components/Cards/Fan'
import Heater1 from '../components/Cards/Heater1'
import Heater2 from '../components/Cards/Heater2'
import Stretcher from '../components/Cards/Stretcher'
import Rewinder from '../components/Cards/Rewinder'
import Sensors from '../components/Cards/Sensors'

const RTMTab = () => {

```

```

return (
  <div style={{ margin: '16px', display: 'flex', flexWrap: 'wrap', justifyContent: 'space-between'}}>
    <Engine/>
    <Heater1/>
    <Heater2/>
    <div>
      <Sensors/>
      <Bunker style={{ marginTop: 8 }}/>
      <Fan style={{ marginTop: 8 }}/>
    </div>
    <div>
      <Stretcher/>
      <Cooling style={{ marginTop: 8 }}/>
    </div>
    <Rewinder/>
  </div>
)
}

export default RTMTab

```

9. Розроблена вкладка “Оптимізація”

client/src/pages/OptimizationTab.js – вкладка “Оптимізація”

```

import { Paper, Tab, Tabs } from '@material-ui/core'
import { useState } from 'react'
import HeatBalance from '../components/Optimization/HeatBalance'
import PowerBalance from '../components/Optimization/PowerBalance'

function TabPanel(props) {
  const { children, value, index, ...other } = props

  return (
    <div
      hidden={value !== index}
      aria-labelledby={`simple-tab-${index}`}
      {...other}
    >
      {value === index && (
        <div>
          {children}
        </div>
      )}
    </div>
  )
}

function OptimizationTab() {
  const [value, setValue] = useState(0)

  const handleChange = (event, newValue) => {
    setValue(newValue)
  }

  const tabStyle = { background: '#c7ecee' }

  return (
    <Paper style={{ margin: 8 }} variant={'outlined'}>
      <Tabs value={value} onChange={handleChange} centered indicatorColor={'primary'}>
        <Tab label="Тепловий баланс" style={tabStyle}/>
      </Tabs>
    </Paper>
  )
}

```

```

    <Tab label="Баланс потужності" style={tabStyle}/>
  </Tabs>
  <TabPanel value={value} index={0}>
    <HeatBalance style={{padding: 8}}/>
  </TabPanel>
  <TabPanel value={value} index={1}>
    <PowerBalance style={{padding: 8}}/>
  </TabPanel>
</Paper>
)
}

export default OptimizationTab

```

client/src/components/Optimization/HeatBalance.js – блок “Тепловий баланс”

```

import React, {useEffect, useState} from 'react'
import {useSelector} from 'react-redux'
import {Button, TextField} from '@material-ui/core'
import {useFormik} from 'formik'

import {heatBalanceParams} from '../../redux/selectors/calc-heat.selectors'
import {calcCoolingHeat, calcEnvHeat, calcMaterialHeat, calcScrewHeat} from '../../calc/HeatBalance'
import CardContainer from './Common/CardContainer'
import MaterialsSelect from './Common/MaterialsSelect'
import Summary from './Common/Summary'

function HeatBalance({style}) {
  const _heatBalanceParams = useSelector(heatBalanceParams)

  const [materialHeat, setMaterialHeat] = useState(0)
  const [coolingHeat, setCoolingHeat] = useState(0)
  const [envHeat, setEnvHeat] = useState(0)
  const [screwHeat, setScrewHeat] = useState(0)
  const [totalHeat, setTotalHeat] = useState(0)

  const formik = useFormik({
    initialValues: _heatBalanceParams,
    enableReinitialize: true
  })

  useEffect(() => {
    const {
      materialParams,
      coolingParams,
      envParams,
      screwParams,
      facilities
    } = formik.values

    const material = facilities.materials.find(m => m.id === +materialParams.current)

    setMaterialHeat(calcMaterialHeat(
      +materialParams.consumption,
      material?.heatCapacity,
      +materialParams.temperatureInput,
      +materialParams.temperatureOutput
    ))
    setCoolingHeat(calcCoolingHeat(

```

```

+coolingParams.channelArea,
+coolingParams.speed,
+coolingParams.temperatureInput,
+coolingParams.temperatureOutput
))
setEnvHeat(calcEnvHeat(
+envParams.diameter,
+envParams.length,
+envParams.corpusTemp,
+envParams.envTemp
))
setScrewHeat(calcScrewHeat(
+screwParams.diameter,
+screwParams.speed,
material?.viscosity,
+screwParams.pressurePartLength,
+screwParams.avgCuttingDepth,
+screwParams.cuttingStep,
+screwParams.volumetricFlow,
+screwParams.pressure,
+screwParams.clearance
))
setTotalHeat(envHeat + coolingHeat + materialHeat - screwHeat)
}, [formik.values, envHeat, coolingHeat, materialHeat, screwHeat])

const defaultTextFieldProps = {
  size: 'small',
  variant: 'outlined',
  InputLabelProps: {shrink: true},
  fullWidth: true
}

return (
<form onSubmit={formik.handleSubmit} onReset={formik.handleReset} style={{...style}}>
<div style={{display: 'flex', overflowX: 'auto', flexWrap: 'nowrap'}}>
<CardContainer title={'Тепло, що витрачається на матеріал, Вт'}>
  <MaterialsSelect onChange={id => formik.setFieldValue('materialParams.current', id)}
    currentMaterialId={formik.values.materialParams.current}
    style={{marginTop: 12}}
    materials={formik.values.facilities.materials}/>
  <TextField style={{marginTop: 12}}
    {...defaultTextFieldProps}
    name={'materialParams.consumption'}
    value={formik.values.materialParams.consumption}
    onChange={formik.handleChange}
    label={'Витрата матеріалу, кг/год'}/>
  <TextField style={{marginTop: 12}}
    {...defaultTextFieldProps}
    name={'materialParams.temperatureInput'}
    value={formik.values.materialParams.temperatureInput}
    onChange={formik.handleChange}
    label={'Т полімеру на вході, C'}/>
  <TextField style={{marginTop: 12}}
    {...defaultTextFieldProps}
    name={'materialParams.temperatureOutput'}
    value={formik.values.materialParams.temperatureOutput}
    onChange={formik.handleChange}
    label={'Т полімеру на виході, C'}/>
  <Summary value={materialHeat} unit={'Вт'}/>
</CardContainer>
</div>
</div>

```

```

    <Summary value={totalHeat} unit={'Вт'} title={'Кількість теплоти'}/>
    <Button onClick={() => formik.resetForm()}
      variant={'contained'}
      size={'small'}
      disabled={!formik.dirty}
      style={{marginTop: 8}}
      color={'primary'}
      fullWidth>
      Зкинути зміни
    </Button>
  </div>
</form>
)
}

```

```
export default HeatBalance
```

client/src/components/Optimization/PowerBalance.js – блок “Потужність”

```

import { Button, TextField } from '@material-ui/core'
import { useFormik } from 'formik'
import React, { useEffect, useState } from 'react'
import { useSelector } from 'react-redux'
import { calc_N1, calc_N2 } from '../../calc/PowerBalance'
import CardContainer from './Common/CardContainer'
import { powerBalanceParams } from '../../redux/selectors/cals-engine-power.selectors'
import Summary from './Common/Summary'

function PowerBalance({ style }) {

  const _powerBalanceParams = useSelector(powerBalanceParams)

  const [N1, setN1] = useState(0)
  const [N2, setN2] = useState(0)
  const [totalPower, setTotalPower] = useState(0)

  const formik = useFormik({
    initialValues: _powerBalanceParams,
    validationSchema: null,
    enableReinitialize: true
  })

  useEffect(() => {
    const {
      diameter,
      cuttingStep,
      coreDiameters,
      cuttingDepth,
      e,
      length,
      viscosity,
      speed,
      a,
      pressure,
      clearance
    } = formik.values

    setN1(calc_N1(
      +diameter,
      +cuttingStep,
      {

```

```

    d1: +coreDiameters.d1,
    d2: +coreDiameters.d2,
    d3: +coreDiameters.d3
  },
  {
    h1: +cuttingDepth.h1,
    h2: +cuttingDepth.h2,
    h3: +cuttingDepth.h3
  },
  +e,
  +length,
  +viscosity,
  +speed,
  +a,
  +pressure
))

setN2(calc_N2(
  +diameter,
  +cuttingStep,
  +e,
  +length,
  +speed,
  +viscosity,
  +clearance
))

setTotalPower(N1 + N2)
}, [formik.values, N1, N2])

const defaultTextFieldProps = {
  size: 'small',
  variant: 'outlined',
  InputLabelProps: { shrink: true },
  fullWidth: true
}

return (
  <form onSubmit={formik.handleSubmit} onReset={formik.handleReset} style={{...style}}>
    <div style={{ display: 'flex', overflowX: 'auto', flexWrap: 'nowrap', justifyContent: 'center'}}>
      <CardContainer title={'Потужність, що витрачається на обертання шнека, Вт'}>
        <TextField style={{ marginТop: '12px' }}
          {...defaultTextFieldProps}
          name={'diameter'}
          value={formik.values.diameter}
          onChange={formik.handleChange}
          label={'Діаметр шнека, см'}/>
        <TextField style={{ marginТop: '12px' }}
          {...defaultTextFieldProps}
          name={'length'}
          value={formik.values.length}
          onChange={formik.handleChange}
          label={'Довжина шнека, см'}/>
        <TextField style={{ marginТop: '12px' }}
          {...defaultTextFieldProps}
          name={'cuttingStep'}
          value={formik.values.cuttingStep}
          onChange={formik.handleChange}
          label={'Крок нарізки шнека, см'}/>
        { //.....// }
      </CardContainer>
    </div>
  </form>
)

```

```

<Summary value={totalPower} unit='Вт'/>
<Button onClick={() => formik.resetForm()}
  variant={'contained'}
  size={'small'}
  disabled={!formik.dirty}
  style={{marginTop: '8px'}}
  color={'primary'}
  fullWidth>
  Зкинути зміни
</Button>
</form>
)
}

export default PowerBalance

```

4.6 Розробка функціонального модуля системи підтримки прийняття рішень

Система підтримки прийняття рішення (СППР) – комп’ютеризована система, яка через збирання та аналіз інформації може впливати на процес ухвалення рішення з управління технологічним процесом й у розрізі екструзійного виробництва, підвищувати його ефективність.

На базі попередньо розрахованих математичних залежностей – енергетичного балансу та потужності екструдера мовою JavaScript розроблено функціональний модуль системи підтримки прийняття рішення по управлінню екструзією полімерних матеріалів, який було інтегровано в розроблений Web-додаток системи управління екструзійної лінії.

Програмний код функціонального модуля СППР приведено нижче:

client/src/calc/HeatBalance.js – модуль для розрахунку енергетичного балансу екструдера

```

import {
  cross_square,
  cylinder_square,
  fi_calc, rad_to_deg,
  transfer_coefficient,
  water_waste
} from './Common'

/**
 * Рассчитать тепло, уносимое материалом, Вт
 * @param Gm Расход материала, кг/час
 * @param Cm Теплоемкость материала, Дж/(кг*К)
 * @param t_enter Т полимера на входе в экструдер, С
 * @param t_exit Т полимера на выходе из экструдера, С
 * @return {number} Тепло, уносимое материалом, Вт

```

```

*/
export function calcMaterialHeat(Gm = 0, Cm = 0, t_enter = 0, t_exit = 0) {
  return Math.floor((Gm * Cm * (t_exit - t_enter)) / 3600)
}

/**
 * Рассчитать тепло, поглощаемое системой охлаждения, Вт
 * @param channelArea Диаметр канала охлаждения, м
 * @param speed Скорость циркуляции воды в охлаждающей системе
 * @param t_enter Т воды на входе в экструдер, С
 * @param t_exit Т воды на выходе из экструдера, С
 * @returns {number} Тепло, поглощаемое системой охлаждения, Вт
 */
export function calcCoolingHeat(channelArea = 0, speed = 0, t_enter = 0, t_exit = 0) {
  const water_density = 1000 // Плотность воды, кг/м3
  const water_heat_capacity = 4190 // Теплоемкость воды, Дж / (кг * К)
  const Gw = water_waste(water_density, cross_square(channelArea), speed)
  return Math.floor(Gw * water_heat_capacity * (t_exit - t_enter))
}

/**
 * Рассчитать тепло, выделяемое при работе шнека, Вт
 * @param D Диаметр шнека, см
 * @param N Частота вращения шнека, об/с
 * @param viscosity Вязкость расплава в условиях переработки, Па * С
 * @param Ln Длина напорной части шнека, см
 * @param h_mid Средняя глубина нарезки шнека, см
 * @param t Шаг нарезки шнека, см
 * @param Q Объемный расход экструдера, см3/с
 * @param P Давление в экструдере в режиме переработки, Па
 * @param gap Величина зазора между гребнем и шнеком, см
 * @returns {number}
 */
export function calcScrewHeat(D= 0, N= 0, viscosity= 0, Ln= 0, h_mid= 0, t= 0, Q = 0, P= 0, gap= 0) {
  const fi = rad_to_deg(fi_calc(t, D, h_mid))
  const p1 = (Math.PI ** 3 * D ** 3 * N ** 2 * viscosity * Ln) / h_mid
  const p2 = (Q * P) / Math.cos(fi) ** 2
  const p3 = (Math.PI ** 2 * D ** 2 * N ** 2 * viscosity * Ln) / (gap * Math.tan(fi))
  return Math.floor((p1 + p2 + p3) * 9.8 * 10 ** -7)
}

/**
 * Рассчитать тепло, уходящее в окружающую среду
 * @param diameter Диаметр корпуса экструдера, м
 * @param length Длина корпуса экструдера, м
 * @param t_corpus Т поверхности корпуса экструдера, С
 * @param t_env Т окружающей среды, С
 * @returns {number} Тепло, уходящее в окружающую среду, Вт
 */
export function calcEnvHeat(diameter = 0, length = 0, t_corpus = 0, t_env = 0) {
  const F = cylinder_square(1.7, 0.4)
  const alpha = transfer_coefficient(t_corpus, t_env)
  return Math.floor(F * alpha * (t_corpus - t_env))
}

```


client/src/calc/PowerBalance.js – математичний модуль розрахунку

потужності

```

/**
 * Рассчитать коэффициент J
 * @param D Диаметр шнека, см
 * @param t Шаг нарезки шнека, см
 * @param d1 Диаметр сердцевины вала в зоне плавления, см
 * @param d2 Диаметр сердцевины вала в зоне пластификации, см
 * @param d3 Диаметр сердцевины вала в зоне дозирования, см
 * @param h1 Глубина нарезки в зоне плавления, см
 * @param h2 Глубина нарезки в зоне пластификации, см
 * @param h3 Глубина нарезки в зоне плавления, см
 * @returns {number} Значение коэффициента
 */
function calc_J(D, t, d1, d2, d3, h1, h2, h3) {
  const d1_d2 = (d1 + d2) / 2
  const h1_h2 = (h1 + h2) / 2
  const arg1 = (Math.pow(Math.PI, 2) * Math.pow(D, 2) - 4 * t) / Math.pow(Math.PI, 2)
  const arg2 = (Math.pow(D + d3, 3) - Math.pow(D + d1_d2, 3)) / 3 * (d3 - d1_d2)
  const arg3 = (2.3 * Math.pow(Math.PI, 2) * Math.pow(D, 5) * Math.log10(h1_h2 / h3)) /
    (Math.pow(t, 2 + Math.pow(Math.PI, 2)) + Math.pow(D, 2)) + (h1_h2 - h3)

  return arg1 + arg2 + arg3
}
/**
 * Рассчитать мощность, потребляемую экструдером на передвижение массы вдоль спирального
канала к головке
 * @param D Диаметр шнека, см
 * @param t Шаг нарезки шнека, см
 * @param coreDiameters Диаметр сердцевины шнека по зонам, см {d1, d2, d3}
 * @param cuttingDepth Глубина нарезки шнека по зонам, см {h1, h2, h3}
 * @param e Ширина гребня, см
 * @param L Длина шнека, см
 * @param viscosity Вязкость полимера в условиях переработки, Па * с
 * @param n Частота вращения шнека, 1/с
 * @param A Постоянная прямого потока экструдера, см3
 * @param delta_P Перепад давления в головке, Па
 * @returns {number}
 */
export function calc_N1(D = 0, t = 0,
  coreDiameters = {
    d1: 0,
    d2: 0,
    d3: 0
  },
  cuttingDepth = {
    h1: 0,
    h2: 0,
    h3: 0
  }, e = 0, L = 0, viscosity = 0, n = 0, A = 0, delta_P = 0) {
  const {d1, d2, d3} = coreDiameters
  const {h1, h2, h3} = cuttingDepth
  const J = calc_J(D, t, d1, d2, d3, h1, h2, h3)
  const arg1 = (Math.pow(Math.PI, 3) * (t - e) * L * J * viscosity) / Math.pow(t, 2) * Math.pow(n, 2)
  const arg2 = A * delta_P * n
  return Math.floor(9.8 * Math.pow(10, -7) * (arg1 + arg2))
}

```

```

/**
 * Мощность, затрачиваемая на срез материала в зазоре между шнеком и корпусом цилиндра
 * @param D Диаметр шнека, см
 * @param t Шаг нарезки, см
 * @param e Ширина гребня, см
 * @param L Длина шнека, см
 * @param n Частота вращения шнека, 1/с
 * @param viscosity Вязкость расплава в условия переработки, Па * с
 * @param clearance Зазор между гребнем шнека и стенкой цилиндра
 * @returns {number}
 */
export function calc_N2(D = 0, t = 0, e = 0, L = 0, n = 0, viscosity = 0, clearance = 0) {
  const arg1 = (Math.pow(Math.PI, 3) * Math.pow(D, 3) * e * L * viscosity) / (clearance * t)
  return Math.floor(9.8 * Math.pow(10, -7) * arg1 * Math.pow(n, 2))
}

```

В розділі розглянуто архітектуру розробленого Web-додатку, котрий складається з серверної та клієнтської частин, та функціональний модуль СППР по управлінню процесом екструзії полімерних матеріалів.

Описано ключові складові серверної частини Web-додатку та їх взаємодія в цілому: ініціалізація Express-сервера, створення WebSocket-з'єднання, реєстрація маршрутів HTTP-запитів, імплементація сервісів для роботи з базою даних PostgreSQL та обробки з COM-з'єднання.

Розроблена клієнтська частина, складається з односторінкового Web-додатку на React. В цей додаток було інтегровано Redux-сховище для зберігання поточного стану змінних додатку та їх централізованого оновлення.

В розроблений Web-додаток інтегровано WebSocket-посередника, який відповідає за відкриття нерозривного з'єднання з сервером для передачі даних в режимі реального часу.

Для відображення інтерактивних компонентів (кнопок, текстових полів, модальних вікон і т.д.) використано сучасну бібліотеку графічних компонентів для React-додатків – MaterialUI.

Для відображення лінійних графіків динаміки зміни температури нагрівачів використано бібліотеку Recharts, що містить в своєму арсеналі різні типи графіків (лінійні, радіальні, стовпчикові і т.д.), котрі легко

інтегруються у Web-додаток та гнучко налаштовуються під відповідні потреби розробника.

Розробка Web-додатку виконувалася з використанням компонентного підходу, який дає змогу знизити дублювання коду та підвищити продуктивність виконання Web-додатку.

Розроблений Web-додаток складається з двох сторінок: авторизації користувачів та головного екрану. Головний екран, в свою чергу, складається з трьох вкладок: монітор реального часу, мнемосхема екструдера та вікно оптимізації.

Математичне забезпечення системи підтримки прийняття рішення реалізовано за допомогою створення функціонального модуля, в котрому описані функції для розрахунку відповідних функціональних залежностей – теплового балансу та потужності електродвигуна.

РОЗДІЛ 5 ПРАКТИЧНА РЕАЛІЗАЦІЯ

В цьому розділі розглядається візуальна складова та функціональні можливості розробленого Web-додатку системи прийняття рішення по управлінню екструзією.

5.1 Мнемосхема екструзійної лінії

На рисунку 5.1 представлено розроблену мнемосхему екструзійної лінії.

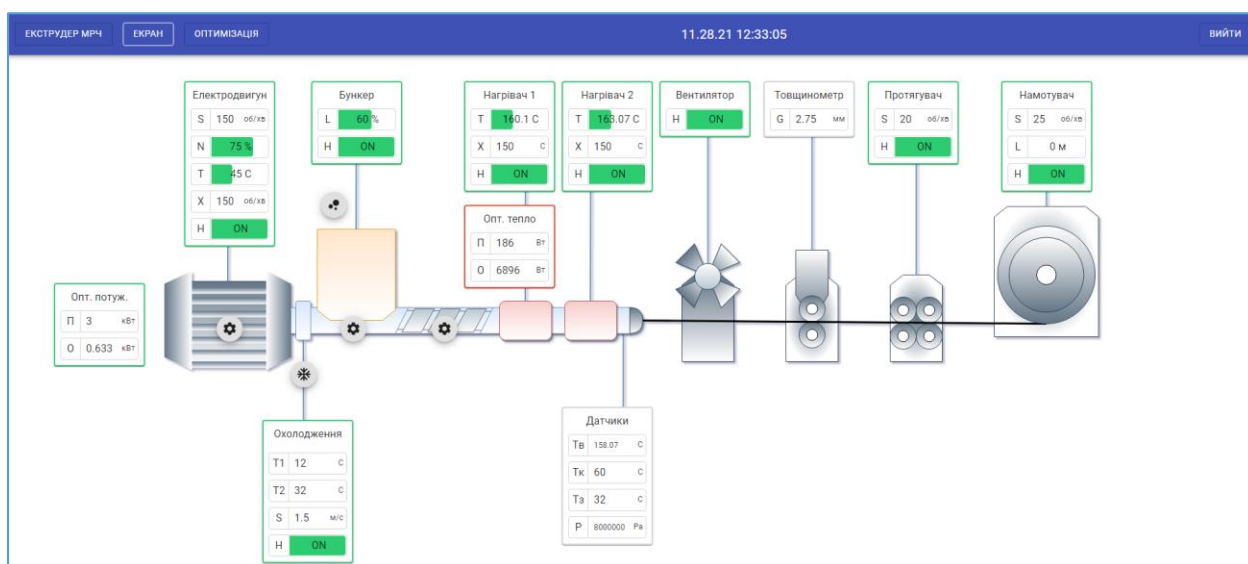


Рисунок 5.1 – Розроблена мнемосхема екструзійної лінії

На мнемосхемі (рис. 5.1) зображено всі агрегати екструзійної лінії:

- екструдер;
- вентилятор;
- товщинометр;
- протягувач;
- намотувач.

Розгорнуте зображення мнемосхеми екструдера зображено у додатку Б.

На рисунку 5.2 на мнемосхемі зображено екструдер:

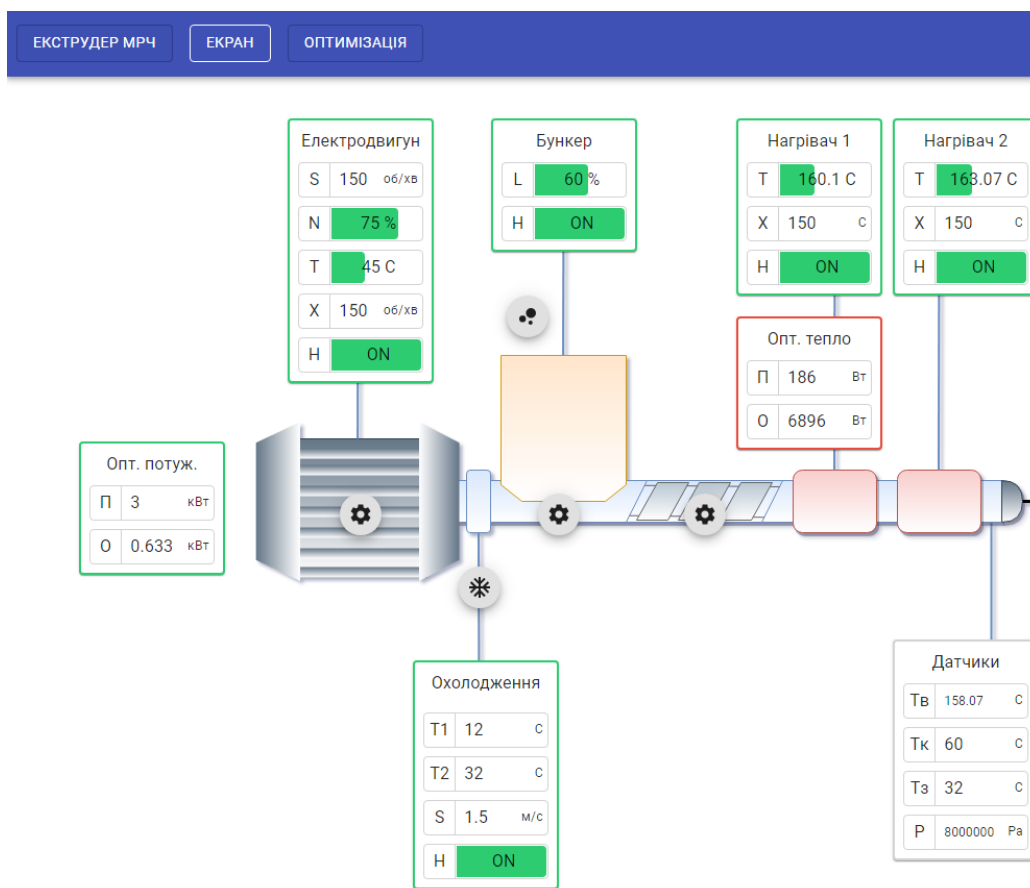


Рисунок 5.2 – Екструдер на мнемосхемі

На мнемосхемі екструдера (рис. 5.2) графічно, у вигляді сгрупованих у блоки, винесено всі поточні технологічні параметри його складових: електродвигуна, бункера, нагрівачів, системи охолодження та інші датчики.

Підхід розташування всіх агрегатів екструзійної лінії на одній мнемосхемі дає змогу оператору-технологу швидко проаналізувати поточний стан та режим роботи виробництва.

Електродвигун екструдера має наступні параметри (табл. 5.1):

Таблиця 5.1 – Технологічні параметри електродвигуна

Параметр	Характеристика	Од. вим.	Значення
S	Поточна швидкість обертання	об/хв	0-300
N	Навантаження	%	0-100%
T	Температура	С	0-200
X	Задана швидкість обертання	об/хв	0-300
H	Стан		ON/OFF

Натискаючи на іконку електродвигуна, відкривається модальне вікно (рис. 5.3), де технолог задає ключову технологічну характеристику електродвигуна – потужність. Це значення в подальшому враховується в системі прийняття рішення, порівнюючись з розрахунковим оптимальним значенням потужності електродвигуна і за допомогою індикації блоку “Опт. потужність” дає операторові зрозуміти чи достатньо (підсвічується зеленим) потужності електродвигуна для ефективного виконання екструзійного процесу чи ні (підсвічується червоним).

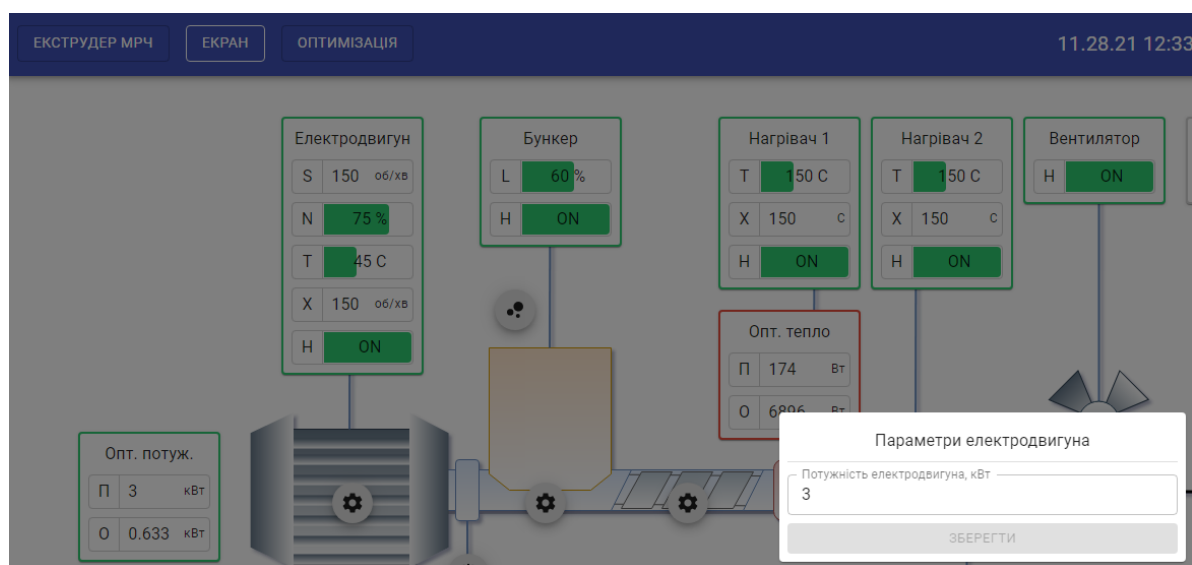


Рисунок 5.3 – Параметри електродвигуна

Система охолодження екструдера має наступні параметри (табл. 5.2)

Таблиця 5.2 – Технологічні параметри системи охолодження

Параметр	Характеристика	Од. вим.	Значення
T1	Температура води на вході	С	0-100
T2	Температура води на виході	С	0-100
S	Швидкість циркуляції води	м/с	0-3
H	Стан		ON/OFF

Натискаючи на іконку охолодження, відкривається модальне вікно з ключовою характеристикою системи охолодження (рис. 5.4) – площа поперечного перетину. Цей параметр враховується для розрахунку оптимальної кількості теплоти з рівняння теплового балансу.

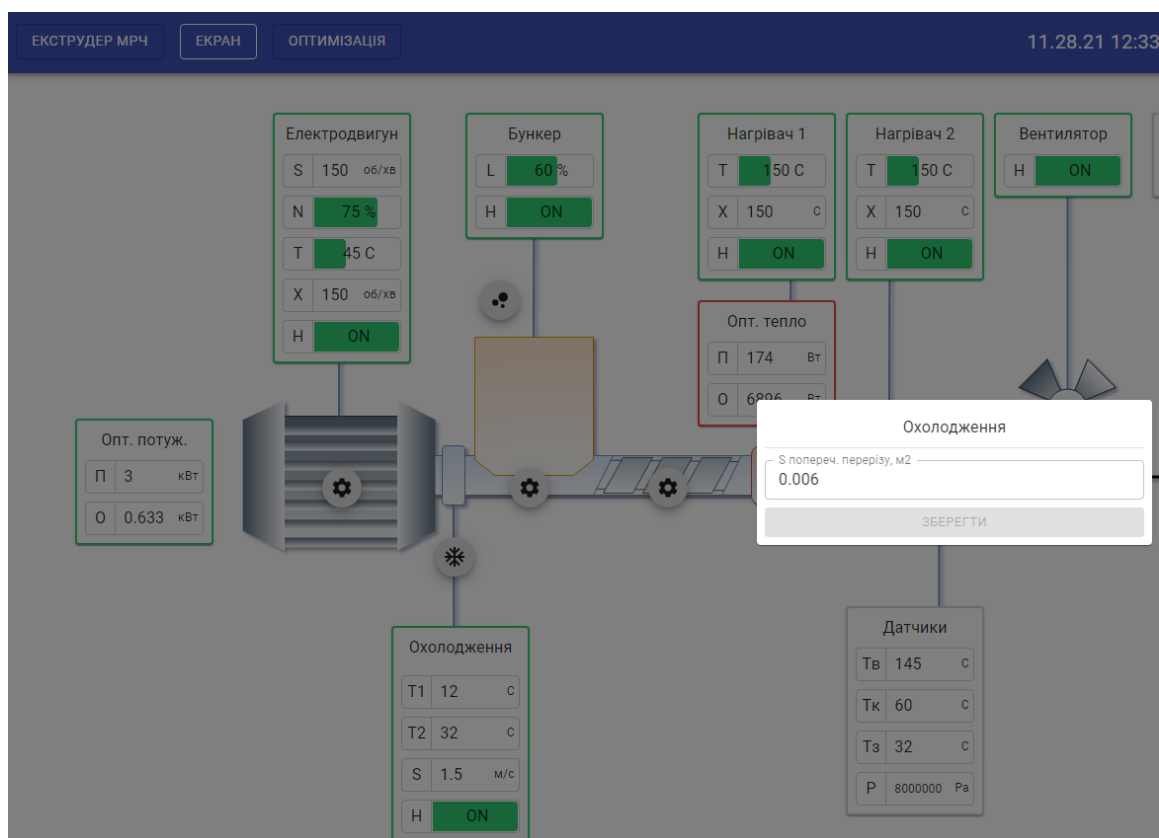


Рисунок 5.4 – Параметри охолодження

Бункер екструдера на мнемосхемі має наступні параметри (табл. 5.3).

Таблиця 5.3 – Параметри бункера

Параметр	Характеристика	Од. вим.	Значення
L	Рівень	%	0-100
H	Стан (Автозаповнення)		ON/OFF

При натисканні на іконку матеріал, відкривається модальне вікно з параметрами матеріалу (рис. 5.5), який наразі задіяний у виробництві та інші додаткові параметри. Всі ці значення враховуються як у розрахунку оптимальної кількості теплоти, так і потужності електродвигуна.

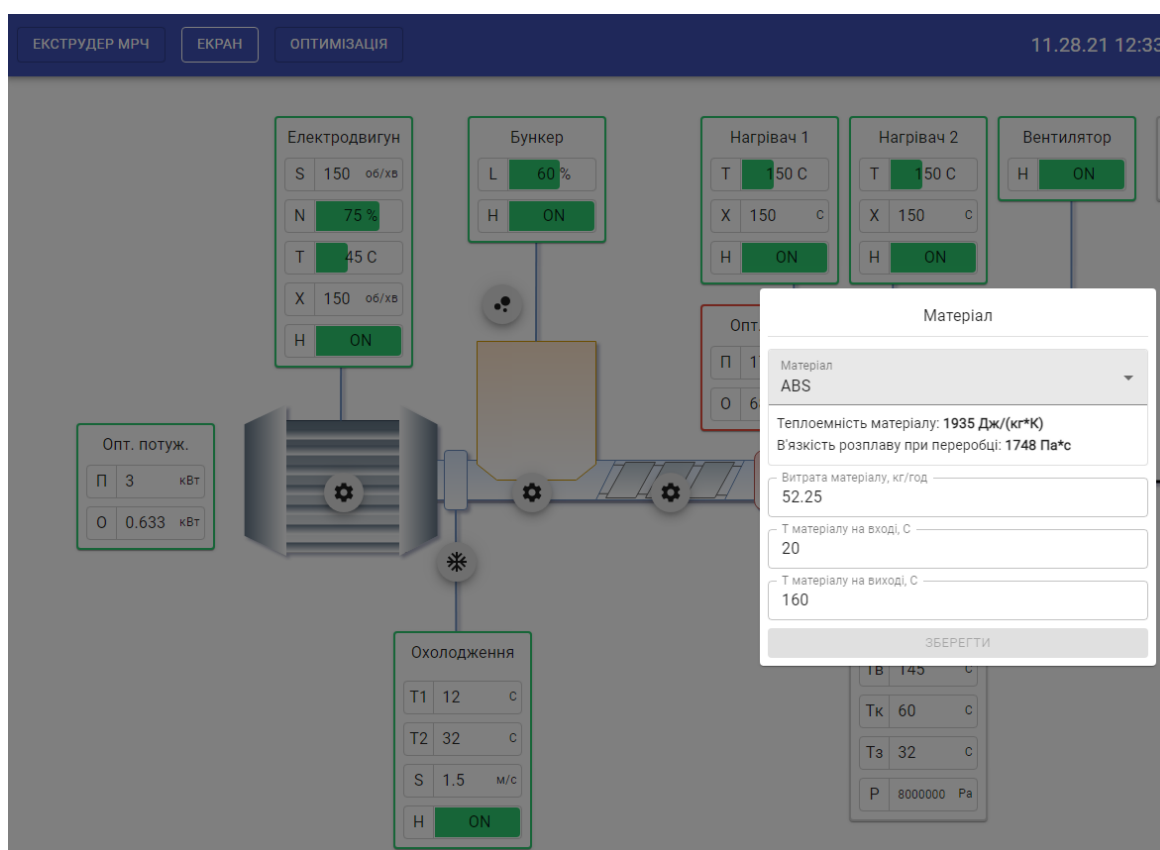


Рисунок 5.5 – Параметри матеріалу

Задаються параметри корпусу, при натисканні відповідної іконки (рис. 5.6). Параметри корпусу враховуються для розрахунку теплоти, котра витрачається у навколишнє середовище через його поверхню за допомогою конвекційного теплообміну.

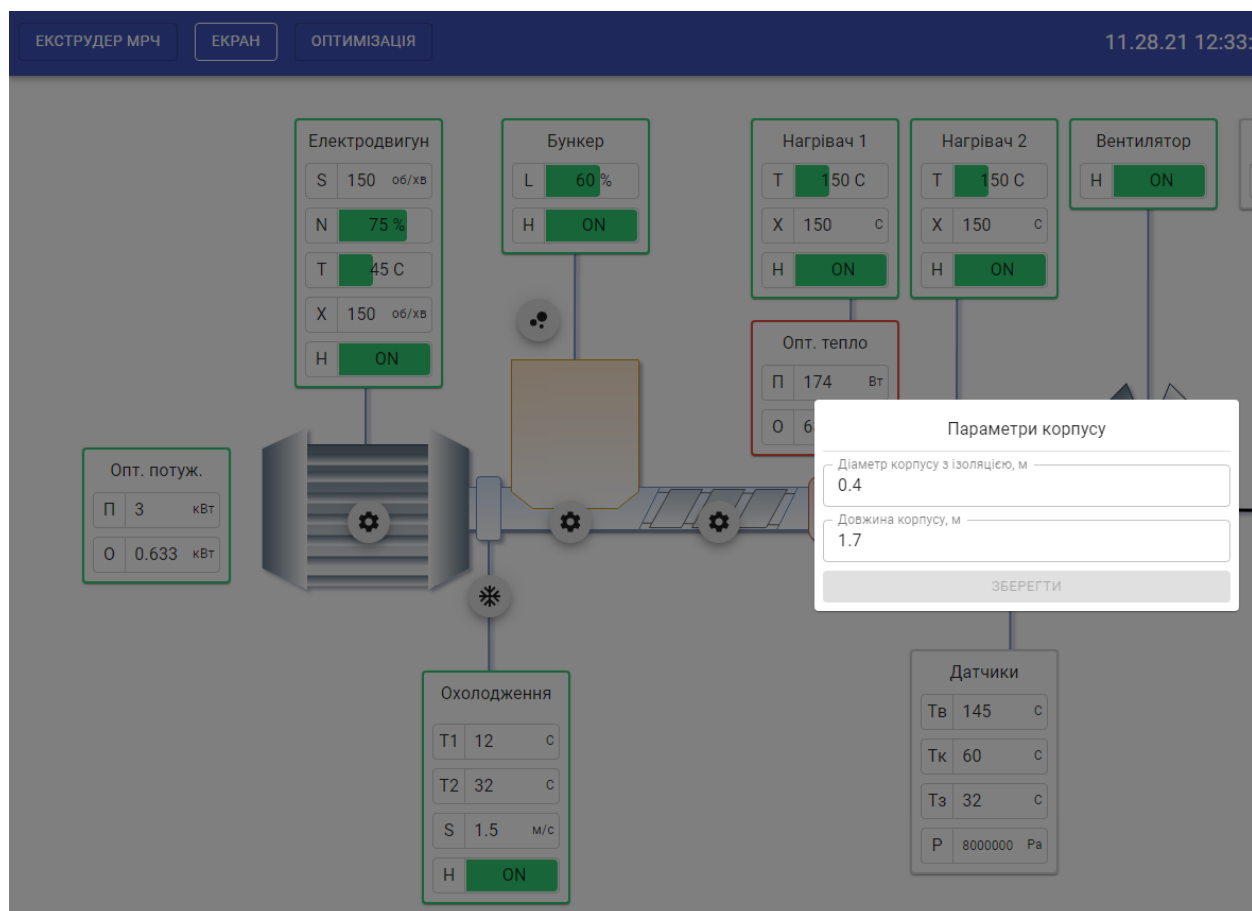


Рисунок 5.6 – Параметри корпусу

Фізичні параметри шнека (рис. 5.7) відіграють важливу роль у розрахунках оптимальних параметрів у системі прийняття рішення.

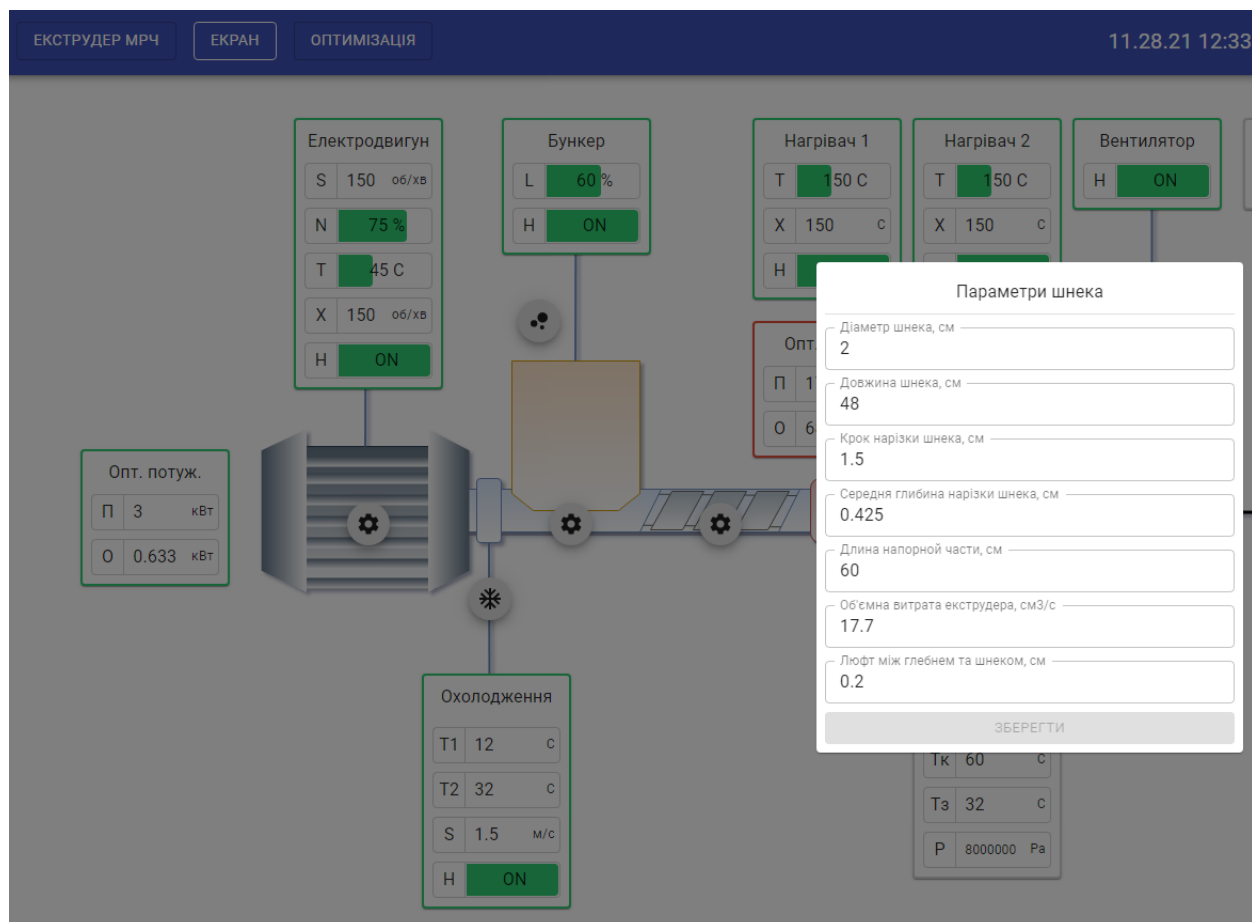


Рисунок 5.7 – Параметри шнека

На екструдері встановлено два нагрівача (рис. 5.8).

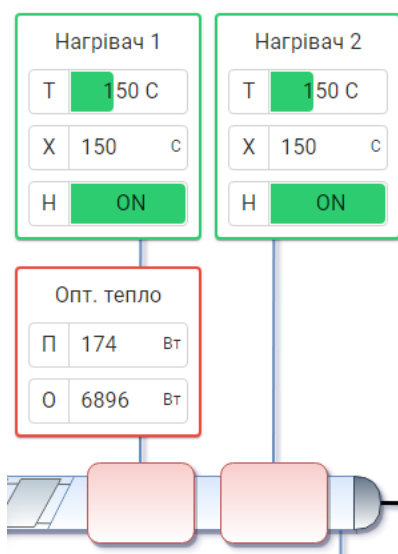


Рисунок 5.8 – Нагрівачі

Технологічні параметри кільцевих нагрівачів зведено в таблицю 5.4.

Таблиця 5.4 – Технологічні параметри кільцевих нагрівачів

Параметр	Х-ка	Од. вим.	Значення
Т	Поточна температура	С	0-400
Х	Завдання	С	0-400
Н	Стан		ON/OFF

Поряд з нагрівачами знаходиться блок оптимізації (рис. 5.9), котрий виводить оптимальну та поточну кількість теплоти у Вт. І в залежності чи задовольняє поточна кількість теплоти, яка виділяється в процесі роботи агрегату оптимальну розрахункову, блок оптимізації підсвічується відповідним кольором (зелений – задовольняє, червоний – не задовольняє)

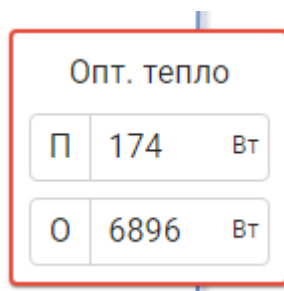


Рисунок 5.9 – Блок системи підтримки прийняття рішення (оптимальна кількість теплоти (О) в порівнянні з поточною (П))

В зоні фільтери екструдера виведено наступні датчики (рис. 5.10):

- внутрішня температура (Тв);
- температура корпусу екструдера (Тк);
- температура зовнішнього середовища (Тз);
- тиск в зоні фільтери (Р).

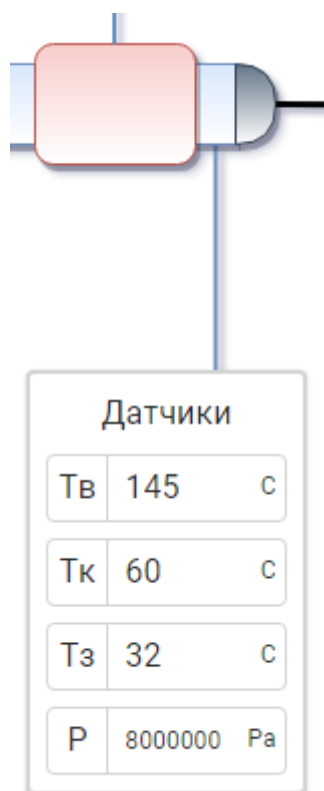


Рисунок 5.10– Датчики в зоні філь'єри

Вентилятор – компонент екструзійної лінії (рис. 5.11), який потрібен задля більш швидкого охолодження філаменту на виході з екструдера.

Оператор задає поточний стан (вкл/викл) вентилятора на вкладці монітору реального часу.

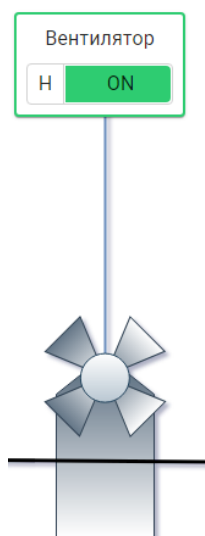


Рисунок 5.11– Вентилятор

Товщинометр призначений для визначення поточної товщини філаменту (рис. 5.12) та саме він є показником якості кінцевого продукту.

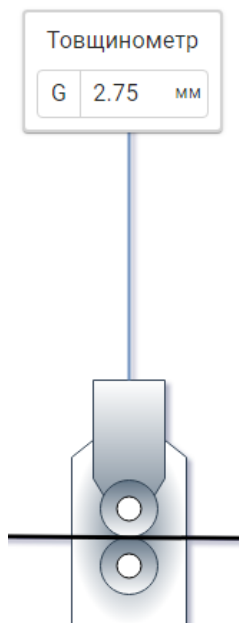


Рисунок 5.12 – Товщинометр

Протягувач (рис. 5.13) призначений для створення постійного натягнення філаменту в процесі виробництва, мінімізуючи провисання філаменту.



Рисунок 5.13 – Протягувач

Параметри протягувача зведено в таблицю 5.5.

Таблиця 5.5 – Параметри протягувача

Параметр	Характеристика	Од. вим.	Значення
S	Задана кількість обертів	об/хв	0-200
H	Стан		ON/OFF

Намотувач (рис. 5.14) призначений для акумуляції виготовленого філаменту.

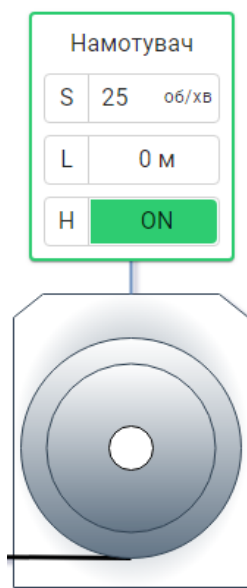


Рисунок 5.14 – Намотувач

Параметри намотувача зведено в таблицю 5.6.

Таблиця 5.6 – Параметри намотувача

Параметр	Характеристика	Од. вим.	Значення
S	Задана кількість обертів	об/хв	0-200
L	Акумульована довжина	м	0-200
H	Стан		ON/OFF

5.2 Екран монітору реального часу екструзійної лінії

На екран монітору реального часу (рис. 5.15) виведено параметри всіх вузлів екструзійної лінії:

- екструдер (електродвигун, нагрівач 1, нагрівач 2, охолодження, бункер, датчики);
- вентилятор;
- протягувач;
- намотувач.

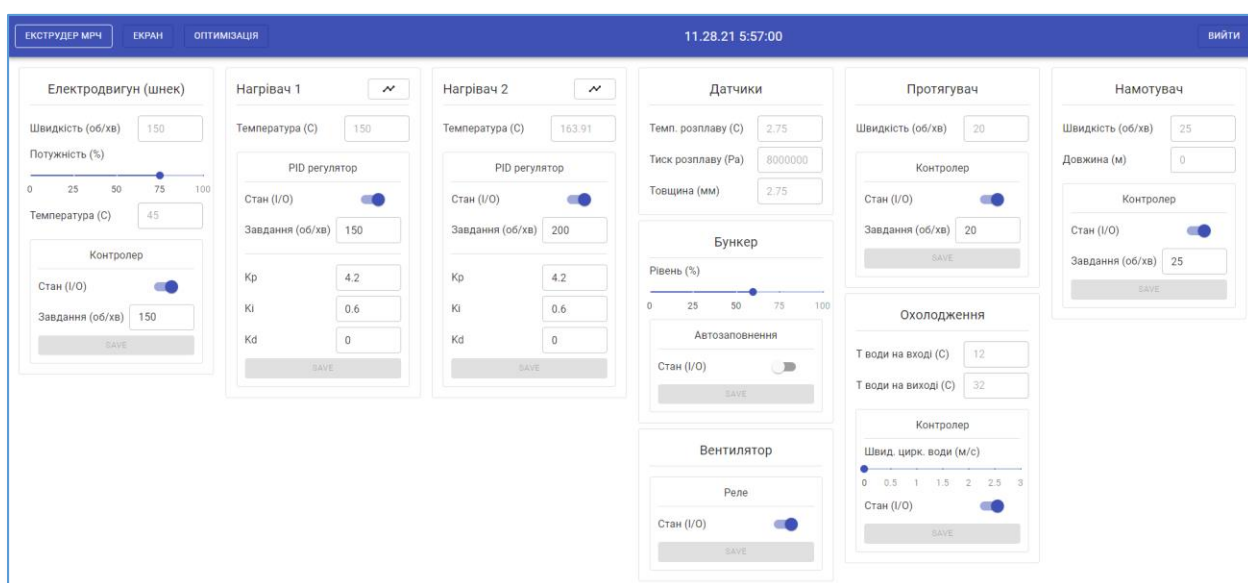


Рисунок 5.15 – Екран монітору реального часу екструдера

Розгорнуте зображення монітору реального часу екструдера зображено у додатку В.

До екструдера відносяться наступні блоки: електродвигун (шнек), нагрівач 1, нагрівач 2, датчики, бункер, охолодження.

Блок “Електродвигун 1” (рис. 5.16) містить наступні елементи:

- поточна швидкість, об/хв;
- потужність, %;
- температура, С;
- контролер: стан, I/O;
- контролер: завдання, С.

Електродвигун (шнек)

Швидкість (об/хв) 150

Потужність (%)

0 25 50 75 100

Температура (C) 45

Контролер

Стан (I/O)

Завдання (об/хв) 150

SAVE

Рисунок 5.16– Блок електродвигуна

Блок “Нагрівач 1” (рис. 5.17) містить наступні параметри:

- поточна температура, C;
- PID-регулятор: стан, I/O;
- PID-регулятор: завдання, об/хв;
- PID-регулятор: K_p ;
- PID-регулятор: K_i ;
- PID-регулятор: K_d .

Нагрівач 1 📈

Температура (C)

PID регулятор

Стан (I/O)

Завдання (об/хв)

Kp

Ki

Kd

SAVE

Рисунок 5.17 – Нагрівач 1

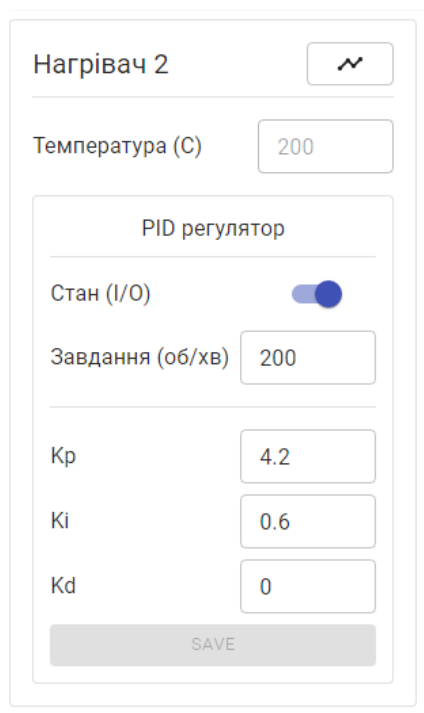
При натисканні на кнопку “Графік” відкривається модальне вікно з графіком зміни температури нагрівача відносно завдання (5.18).



Рисунок 5.18 – Графік зміни температури нагрівача №1

Блок “Нагрівач 2” (рис. 5.19) містить наступні параметри:

- поточна температура, С;
- PID-регулятор: стан, I/O;
- PID-регулятор: завдання, об/хв;
- PID-регулятор: Кр;
- PID-регулятор: Кі;
- PID-регулятор: Кd.



Нагрівач 2

Температура (С) 200

PID регулятор

Стан (I/O)

Завдання (об/хв) 200

Кр 4.2

Кі 0.6

Кd 0

SAVE

Рисунок 5.19 – Блок “Нагрівач 2”

При натисканні на кнопку “Графік” відкривається модальне вікно з графіком зміни температури нагрівача відносно завдання (рис. 5.20).

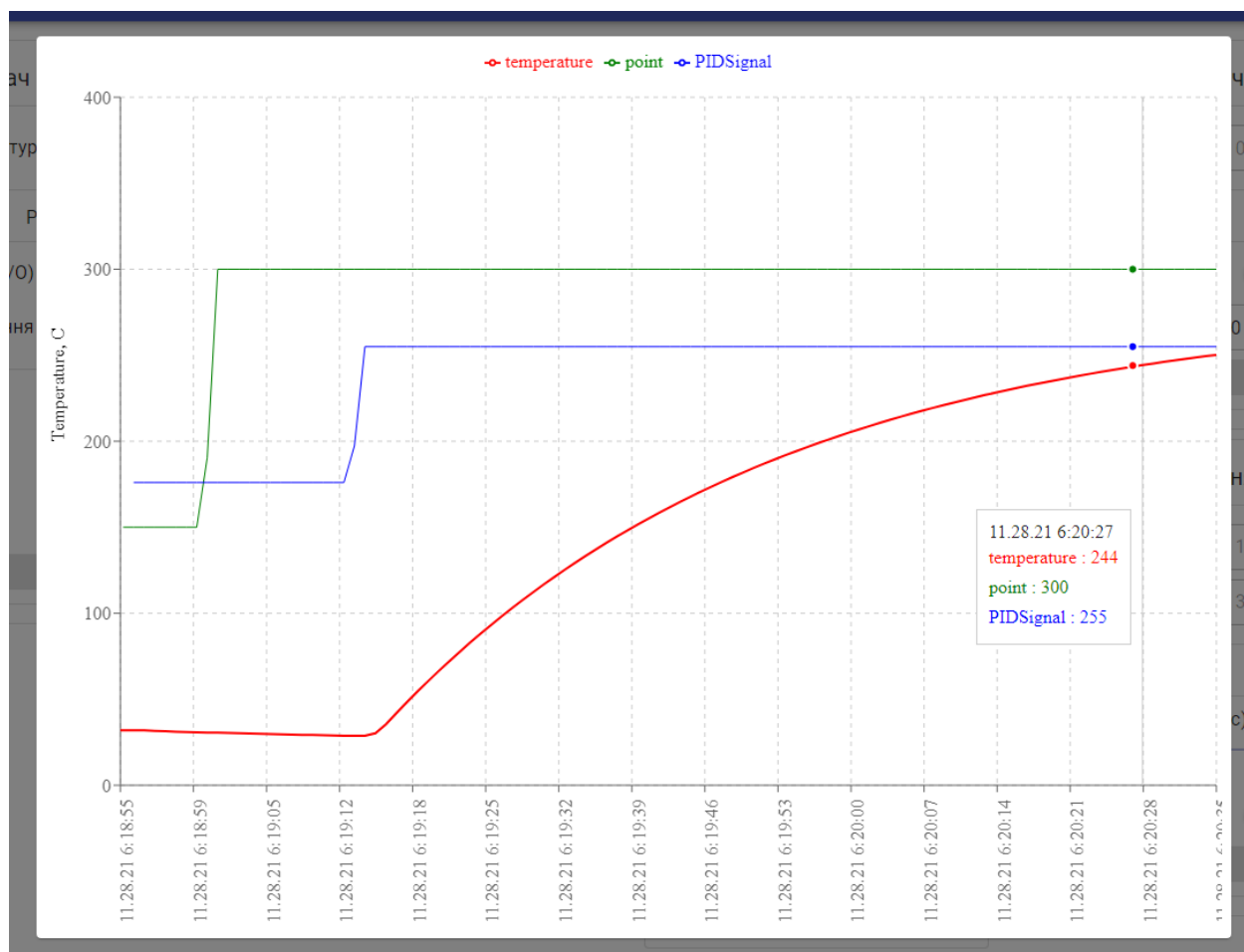


Рисунок 5.20 – Графік зміни температури нагрівача №2

Блок “Бункер” містить наступні параметри (рис. 5.21):

- рівень, %;
- автозаповнення: стан, I/O.

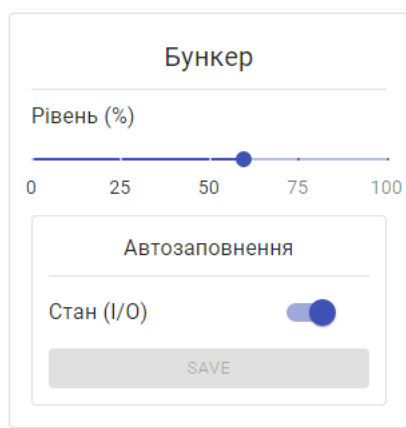


Рисунок 5.21 – Блок “Бункер”

Блок “Охолодження” містить наступні параметри (рис. 5.22):

- температура води на вході, С;
- температура води на виході, С;
- контролер: швидкість циркуляції води, м/с.
- контролер: стан, I/O.

Охолодження

Т води на вході (С) 12

Т води на виході (С) 32

Контролер

Швид. цирк. води (м/с)

0 0.5 1 1.5 2 2.5 3

Стан (I/O)

SAVE

Рисунок 5.22 – Блок “Охолодження”

Блок “Датчики” містить наступні параметри (рис. 5.23):

- температура розплаву в зоні фільєри, С;
- тиск розплаву в зоні фільєри, Па;
- поточна товщина філаменту, мм.

Датчики

Темп. розплаву (С) 150

Тиск розплаву (Па) 8000000

Товщина (мм) 2.75

Рисунок 5.23– Блок “Датчики”

Блок “Вентилятор” (рис. 5.24) містить тумблер для зміни поточного стану охолоджувального вентилятора.

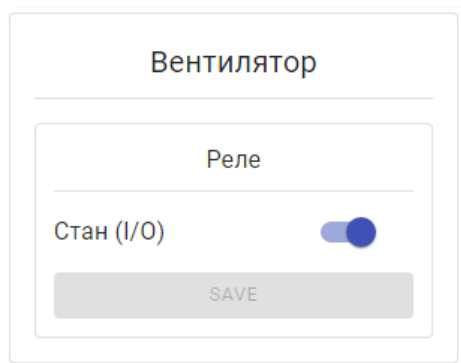


Рисунок 5.24 – Блок “Вентилятор”

Блок “Протягувач” містить наступні параметри (рис. 5.25):

- швидкість обертання, об/хв;
- контролер: швидкість обертання, об/хв;
- контролер: стан, I/O.

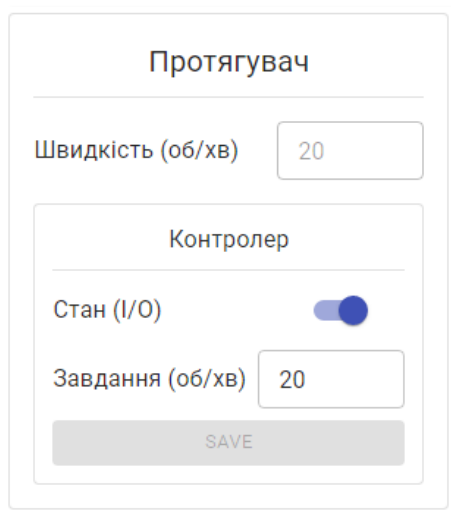


Рисунок 5.25 – Блок “Протягувач”

Блок “Намотувач” містить наступні параметри (рис. 5.26):

- швидкість обертання, об/хв;
- акумульована довжина, м;
- контролер: швидкість обертання, об/хв;
- контролер: стан, I/O.

Намотувач

Швидкість (об/хв) 25

Довжина (м) 0

Контролер

Стан (I/O)

Завдання (об/хв) 25

SAVE

Рисунок 5.26 – Блок “Намотувач”

5.3 Вікно оптимізації СППР

Вікно оптимізації реалізує взаємодію з системою підтримки прийняття рішення по управлінню екструзією полімерних матеріалів. В цьому вікно, на відповідних вкладках, встановлюються вхідні дані для СППР та виконується автоматичний розрахунок значень кількості теплоти та потужності екструдера.

Вікно складається з двох вкладок: тепловий баланс та потужність.

5.3.1 Вкладка “Кількість теплоти”

На рисунку 5.27 приведено вкладку системи підтримки прийняття рішення – “Кількість теплоти”.

Тепло, що витрачається на матеріал, Вт	Тепло, що поглинається системою охолодження, Вт	Тепло, що витрачається у навколишнє середовище, Вт	Тепло, що виділяється при роботі шнека, Вт
Матеріал: ABS Теплоємність матеріалу: 1935 Дж/(кг·К) В'язкість розплаву при переробі: 1748 Па·с Витрата матеріалу, м³/год: 2 Т ооди на виході, С: 24 Т ооди на виході, С: 160 Результати розрахунків: 146 Вт	Площа поперечного перетину, м²: 0.006 Т ооди на виході, С: 12 Т ооди на виході, С: 32 Швидкість циркуляції води, м/с: 1.5 Результати розрахунків: 3554 Вт	Діаметр корпусу з ізоляцією, мм: 0.4 Довжина корпусу, м: 1.7 Температура поверхні корпусу, С: 60 Температура навколишнього середовища, С: 24 Результати розрахунків: 942 Вт	Крок нарізки шнека, см: 1.5 Діаметр шнека, см: 2 Середня глибина нарізки шнека, см: 0.425 Величина зазору між гребнем і шнеком, см: 0.2 Довжина валорної частини шнека, см: 60 Об'ємна витрата екструдера, см³/с: 17.7 Тиск в екструдері, Па: 8000000 Частота обертання шнека, об/с: 2.5 Результати розрахунків: 1288 Вт
Кількість теплоти: 3354 Вт			

Рисунок 5.27 – Графічний інтерфейс СППР (кількість теплоти)

Розгорнуте зображення графічного інтерфейсу СППР (кількість теплоти) приведено у додатку Г.

На вкладці “Тепловий баланс”, оператору-технологу відображається поточна розрахункова кількість теплоти.

Оператор-технолог має можливість змінювати значення технологічних параметрів з метою оцінки їх впливу на технологічний процес екструзії.

Блок “Тепло, що витрачається на матеріал, Вт” (рис. 5.28).

The screenshot shows a web-based calculation tool. At the top, the title is "Тепло, що витрачається на матеріал, Вт". Below the title is a dropdown menu for "Матеріал" with "ABS" selected. Underneath, two material properties are listed: "Теплоємність матеріалу: 1935 Дж/(кг*К)" and "В'язкість розплаву при переробці: 1748 Па*с". There are three input fields: "Витрата матеріалу, кг/год" with the value "2", "Т полімеру на вході, С" with "24", and "Т полімеру на виході, С" with "160". At the bottom, a blue button displays the result: "Результати розрахунків: 146 Вт".

Рисунок 5.28 – Блок “Тепло, що витрачається на матеріал, Вт”

Блок “Тепло, що поглинається системою охолодження, Вт” (рис. 5.29).

The screenshot shows a web-based calculation tool. The title is "Тепло, що поглинається системою охолодження, Вт". Below the title are four input fields: "Площа поперечного перетину, м2" with "0.006", "Т води на вході, С" with "12", "Т води на виході, С" with "32", and "Швидкість циркуляції води, м/с" with "1.5". At the bottom, a blue button displays the result: "Результати розрахунків: 3554 Вт".

Рисунок 5.29 – Блок “Тепло, що поглинається системою охолодження,
Вт”

Блок “Тепло, що витрачається у навколишнє середовище, Вт” (рис. 5.30).

The screenshot shows a web-based calculation tool titled "Тепло, що витрачається у навколишнє середовище, Вт". It contains four input fields with the following values: "Діаметр корпусу з ізоляцією, мм" (0.4), "Довжина корпусу, м" (1.7), "Температура поверхні корпусу, С" (60), and "Температура навколишнього середовища, С" (24). A blue button at the bottom displays the result: "Результати розрахунків: 942 Вт".

Parameter	Value
Діаметр корпусу з ізоляцією, мм	0.4
Довжина корпусу, м	1.7
Температура поверхні корпусу, С	60
Температура навколишнього середовища, С	24
Результати розрахунків	942 Вт

Рисунок 5.30 – Блок “Тепло, що витрачається у навколишнє середовище, Вт”

Блок “Тепло, що виділяється при роботі шнека, Вт” (рис. 5.31).

The screenshot shows a web-based calculation tool titled "Тепло, що виділяється при роботі шнека, Вт". It contains eight input fields with the following values: "Крок нарізки шнека, см" (1.5), "Діаметр шнека, см" (2), "Середня глибина нарізки шнека, см" (0.425), "Величина зазору між гребнем і шнеком, см" (0.2), "Довжина напорної частини шнека, см" (60), "Об'ємна витрата екструдера, см3/с" (17.7), "Тиск в екструдері, Па" (8000000), and "Частота обертання шнека, об/с" (2.5). A blue button at the bottom displays the result: "Результати розрахунків: 1288 Вт".

Parameter	Value
Крок нарізки шнека, см	1.5
Діаметр шнека, см	2
Середня глибина нарізки шнека, см	0.425
Величина зазору між гребнем і шнеком, см	0.2
Довжина напорної частини шнека, см	60
Об'ємна витрата екструдера, см3/с	17.7
Тиск в екструдері, Па	8000000
Частота обертання шнека, об/с	2.5
Результати розрахунків	1288 Вт

Рисунок 5.31 – Блок “Тепло, що виділяється при роботі шнека, Вт”

5.3.2 Вкладка “Потужність”

На рисунку 5.32 приведено вкладку “Потужність” системи підтримки прийняття рішення.

The screenshot displays the 'Потужність' (Power) tab within the SPPR system interface. The interface includes a navigation bar at the top with 'ЕКСТРУДЕР МРЧ', 'ЕКРАН', and 'ОПТИМІЗАЦІЯ' buttons, along with a timestamp '12.13.21 9:09:45' and a 'ВИЙТИ' button. The main content area is titled 'Потужність, що витрачається на обертання шнека, Вт' (Power consumed for screw rotation, W). Below this title, there is a list of parameters with their current values:

- Діаметр шнека, см: 2
- Довжина шнека, см: 48
- Крок нарізки шнека, см: 1.5
- Величина зазору між гребенями і шнеком, см: 0.2
- Тиск в екструдері, Па: 8000000
- Частота обертання шнека, об/с: 2.5
- Діаметр серцевини шнека в зоні плавлення, см: 1
- Діаметр серцевини шнека в зоні пластифікації, см: 1
- Діаметр серцевини шнека в зоні дозування, см: 1.2
- Глибина нарізки шнека в зоні плавлення, см: 0.5
- Глибина нарізки шнека в зоні пластифікації, см: 0.5
- Глибина нарізки шнека в зоні дозування, см: 0.4

At the bottom of the interface, a blue bar displays the result: 'Результати розрахунків: 633 Вт'. A 'ЗМІНИТИ ЗМІНИ' button is located at the bottom center.

Рисунок 5.32 – Графічний інтерфейс СППР (потужність)

Розгорнуте зображення графічного інтерфейсу СППР (потужність) приведено у додатку Г.

На цій вкладці (рис. 5.32) оператор може оцінити вплив поточних технологічних параметрів на розрахункову потужність електродвигуна екструдера. Змінюючи значення технологічних параметрів в той чи інший бік, система буде розраховувати нову величину необхідної потужності електродвигуна.

Таким чином, оператор має можливість щодо прийняття рішення про зміну реальних технологічних параметрів для отримання більш ефективного протікання екструзійного процесу чи зниження навантаження на електродвигун.

Також оператор може проаналізувати чи задовольняє поточна потужність електродвигуна екструдера при переході на нові технологічні параметри, режими переробки чи новий тип полімеру.

Блок “Потужність, що витрачається на обертання шнека, Вт” (рис. 5.33).

Потужність, що витрачається на обертання шнека, Вт	
Діаметр шнека, см	2
Довжина шнека, см	48
Крок нарізки шнека, см	1.5
Величина зазору між гребнем і шнеком, см	0.2
Тиск в екструдері, Па	8000000
Частота обертання шнека, об/с	2.5
Діаметр серцевини шнека в зоні плавлення, см	1
Діаметр серцевини шнека в зоні пластифікації, см	1
Діаметр серцевини шнека в зоні дозування, см	1.2
Глибина нарізки шнека в зоні плавлення, см	0.5
Глибина нарізки шнека в зоні пластифікації, см	0.5
Глибина нарізки шнека в зоні дозування, см	0.4

Результати розрахунків: 633 Вт

Рисунок 5.33 – Блок “Потужність, що витрачається на обертання шнека, Вт”

В розділі розглянуто візуальну складову розробленого Web-додатку та інтегровану систему підтримки прийняття рішення по управлінню екструзією полімерних матеріалів.

Web-додаток складається з двох сторінок – сторінка авторизації оператора-технолога, котра призначена для його ідентифікації та визначення прав доступу, та головної сторінки, з якою, безпосередньо, взаємодіє авторизований оператор-технолог.

На головній сторінці оператору-технологу відображається мнемосхема екструзійної лінії та поточний стан її агрегатів (екструдер, протягувач та намотувач). Такий підхід дає змогу оперативно оцінити стан екструзійного виробництва та прийняти відповідні рішення по управлінню.

Перейшовши на вкладку “Монітор реального часу”, оператор-технолог має змогу задати нові технологічні параметри екструзійної лінії (температуру нагрівачів, швидкість обертання шнека, швидкість протягувача та намотувача і т.д.) та проаналізувати динаміку зміни температури нагрівання екструдера.

Вкладка “Оптимізація” відповідає за взаємодію з системою підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів, на якій відображаються всі вхідні та вихідні змінні математичної моделі. На цій вкладці оператор може оперативно отримати результат оптимальної розрахункової кількості теплоти, необхідної для прогріву екструдера та потужності електродвигуна екструдера в залежності від поточних характеристик обладнання, режиму роботи лінії та фізико-хімічних властивостей полімеру.

Вихідні розрахункові значення СППР відображаються як на вкладці “Оптимізація”, так і на головній сторінці з мнемосхемою екструзійної лінії у вигляді компактних графічних блоків – це дає змогу операторові-технологу оперативно звіряти поточні значення технологічних параметрів (кількість теплоти та потужність електродвигуна) з розрахунковими параметрами СППР та корегувати їх в залежності від величини відхилення.

Змінюючи значення вхідних параметрів СППР на вкладці “Оптимізація”, оператор-технолог може прогнозувати в який бік зміниться розрахункова кількість теплоти та потужність електродвигуна – це дає змогу прийняти відповідні рішення щодо зміни технологічних параметрів екструдера, щоб отримати максимальну ефективність роботи виробничої лінії.

ВИСНОВКИ

1. В кваліфікаційній роботі магістра було розглянуто технологічний процес екструзії полімерних матеріалів та складові агрегати екструзійної лінії – екструдер, протягувач та намотувач. Спираючись на літературні джерела, було проаналізовано особливості фізико-хімічних процесів, що протікають в одношнековому екструдері в момент екструзії. Перелічено ключові полімерні матеріали та їх фізико-хімічні властивості, котрі використовуються для виготовлення полімерного філаменту та впливають на режим роботи екструзійної лінії.

2. Досліджено енергетичний баланс та потужність екструдера. Виконано математичний розрахунок теплового балансу екструдера, звідки виведено залежність необхідної кількості теплоти для прогріву полімерної пробки в міжшнековому просторі при поточних технологічних параметрах екструдера та фізичних властивостей полімерного матеріалу, що переробляється. Виконано математичний розрахунок необхідної потужності електродвигуна, щоб забезпечити максимальну ефективність протікання екструзійного процесу та збільшити його довготривалий строк служби, шляхом попередження надлишкових навантажень.

Досліджено вплив попереднього підігріву полімеру, частоти обертання електродвигуна та теплоємності полімеру на кінцеву розрахункову кількість теплоти – отримано лінійну залежність в усіх випадках.

Досліджено вплив зазору між шнеком і циліндром на кінцеву розрахункову потужність електродвигуна, отримано експоненціальну залежність.

3. Розроблено структуру, вхідні та вихідні параметри математичної моделі системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів, котра складається з розглянутих математичних залежностей – теплового балансу та розрахункової потужності екструдера.

4. Розглянуто розроблений малогабаритний екструдер та його технічне забезпечення: описано склад та призначення технічних засобів автоматизації, приведено принципові електричні схеми управління та живлення.

5. В результаті застосування прогресивних Web-технологій, таких як React, Redux, WebSocket, MaterialUI та Recharts для клієнтської частини, Express.js, WebSocket, COM та PostgreSQL для серверної частини, розроблено Web-додаток, який вирішує задачі дистанційного диспетчерського управління екструзійним виробництвом та дає змогу гнучко налаштовувати інтерфейс оператора-технолога та здійснювати моніторинг стану агрегатів екструзійної лінії та перебігу технологічного процесу екструзії.

6. Розроблений мовою JavaScript функціональний модуль системи підтримки прийняття рішення по управлінню процесом екструзії полімерних матеріалів, до складу якого входить математична модель енергетичного балансу та потужності екструдера, інтегровано в розроблений Web-додаток – це дозволяє розраховувати рекомендації, щодо підвищення ефективності роботи екструзійної лінії.

7. Система підтримки прийняття рішення по управлінню екструзією полімерних матеріалів у складі прогресивного Web-додатку дає змогу операторові-технологу оперативно отримувати поточний стан екструзійної лінії у вигляді мнемосхеми в режимі реального часу, змінювати значення технологічних параметрів, виконувати автоматичний розрахунок оптимальної кількості теплоти та потужності електродвигуна у разі перелаштування виробничої лінії, зміни режиму роботи чи типу полімерної сировини.

8. Розроблена система підтримки прийняття рішення попереджує ризик випуску бракованої продукції, знос обладнання, витрати на попереднє моделювання та проведення експериментів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Chris Rauwendaal. Polymer Extrusion 5th edition: book. USA : Hanser Publishers, 2014. 934 p.
2. Ким В. С. Теория и практика экструзии полимеров: учебник для студентов высших учебных заведений. Москва : Химия, 2005. 556 с.
3. Труфанова Н. М, Щербинин А. Г., Янков В. И. Плавление полимеров в экструдерах: книга. Москва : НИЦ Регулярная и хаотическая динамика, 2009. 226 с.
4. Обладнання для екструзії полімерів. Современные технологии производства: веб-сайт. URL: <https://extxe.com/7186/ustrojstva-dlja-jekstrudirovanija-polimerov/> (дата звернення – 01.11.2021).
5. Торнер Р. В., Акутин М. С. Оборудование заводов по переработке пластмасс : книга. Москва : Химия, 1986. 400 с.
6. Экструзия пластмасс: Описание и технология переработки. Полімери: веб-сайт. URL: <https://polymers.com.ua/экструзия-пластмасс-описание-и-техно/> (дата звернення – 01.11.2021).
7. Ревяко М. М, Прокопчук Н. Р. Теоретические основы переработки полимеров : учеб. пособие для студентов по специальностям “Химическая технология органических веществ, материалов и изделий”, “Машины и технологии обработки материалов давлением”. Минск : БГТУ, 2009. 305
8. Сагинов С. Н. Исследование и моделирование процесса движения полимера в одношнековом экструдере. *Фундаментальные исследования*. 2012. №12. С. 179 -183.
9. Субботин Е. В., Щербинин А. Г., Труфанова Н. М. Численный анализ процессов тепломассопереноса полимеров в винтовом канале экструдера с учетом теплопроводности шнека. *Вычислительная механика сплошных сред*. 2015. №3. С. 329-339.

10. Сокольський О. Л. Моделювання обладнання і процесів перероблення полімерних матеріалів методом екструзії : монографія. Київ : КПІ ім. Ігоря Сікорського, 2020. 252 с.

11. Мантя Ла Ф. Вторичная переработка пластмасс : книга. Санкт-Петербург : Профессия, 2006. 400 с.

12. Клинков А. С., Беляев П. С., Соколов М. В. Утилизация и вторичная переработка полимерных материалов: учеб. пособие. Тамбов : Изд-во Тамб. гос. техн. ун-та, 2005. 80 с.

13. Порівняння фізико-хімічних властивостей полімерів для 3D-друку: веб-сайт. URL: <https://www.simplify3d.com/support/materials-guide/properties-table/> (дата звернення – 01.11.2021).

14. 25 популярних полімерів для 3D-друку та їх особливості: веб-сайт. URL: <http://3dprintstory.org/25-samih-populyarnih-materialov-dlya-3d-pechati> (дата звернення – 01.11.2021).

15. Офіційний сайт компанії Plexiwire: веб-сайт. URL: <https://shop.plexiwire.com.ua/> (дата звернення – 01.11.2021).

16. Чистякова Т. Б. Полосин А. Н. Методы и технологии синтеза математических моделей процессов экструзии в гибких производствах полимерных материалов. *Вестник СГТУ*. 2011. №4. С. 170-180.

17. Шембель А. С., Антипина О. М. Сборник задач и проблемных ситуаций по технологии переработки пластмасс : учебное пособие для техникумов. Москва: Химия, 1990. 272 с.

18. Соколов М. В., Клинков А. С., Ефремов О. В., Беляев П. С., Однолько В. Г. Автоматизованное проектирование и расчет шнековых машин: монография : Москва : “Издательство Машиностроение-1”, 2004. 248 с.

19. Мікроконтролер Arduino MEGA та його Data Sheet. Arduino+: веб-сайт. URL: <https://arduinoplus.ru/arduino-mega2560/> (дата звернення – 01.11.2021).

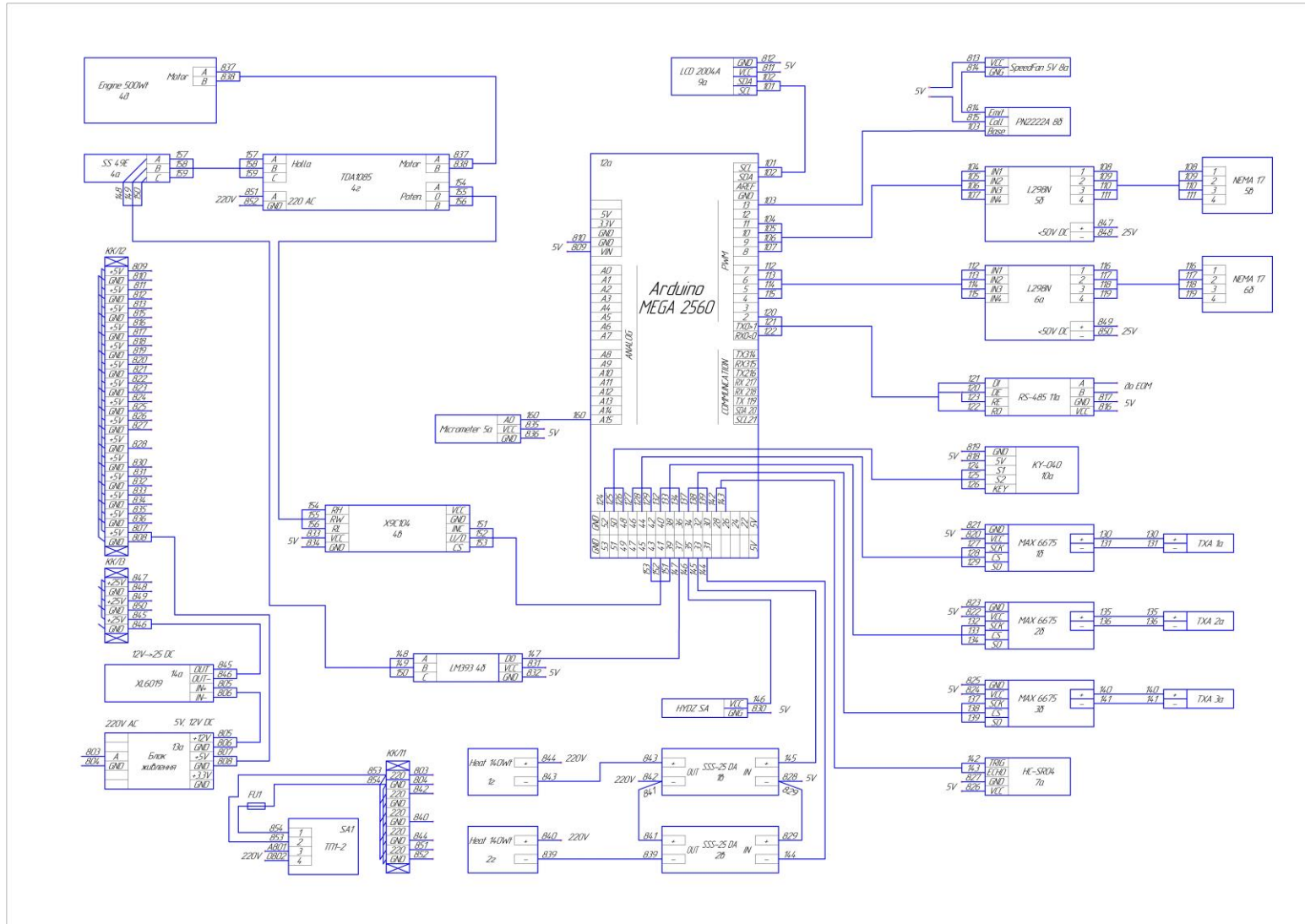
20. Обзоры модулей Arduino. 3DiY: веб-сайт. URL: <https://3d-diy.ru/wiki/arduino-moduli/> (дата звернення – 01.11.2021).
21. Основы програмування Arduino. Arduino Studio. Arduino.ru: веб-сайт. URL: <http://arduino.ru/Reference> (дата звернення – 01.11.2021).
22. React: веб-сайт. URL: <https://ru.reactjs.org/> (дата звернення – 01.11.2021).
23. Redux: веб-сайт. URL: <https://redux.js.org/> URL: (дата звернення – 01.11.2021).
24. React-Redux: веб-сайт. URL: <https://react-redux.js.org/> (дата звернення – 01.11.2021).
25. WebSocket: веб-сайт. URL: <https://developer.mozilla.org/Web/API/WebSocket> (дата звернення – 01.11.2021).
26. Using WebSockets in React: веб-сайт. URL: <https://dev.to/finallynero/using-websockets-in-react-4fkr> (дата звернення – 01.11.2021).
27. WebSocket: особенности протокола и пример использования на React: веб-сайт. URL: <https://tproger.ru/articles/websocket-osobennosti-protokola-i-primer-ispolzovaniya-na-react/> (дата звернення – 01.11.2021).
28. Recharts: веб-сайт. URL: <https://recharts.org/en-US/>(дата звернення – 01.11.2021).
29. Material UI: веб-сайт. URL: <https://mui.com/>(дата звернення – 01.11.2021).
30. Express.js: веб-сайт. URL: <https://expressjs.com/> (дата звернення – 01.11.2021).
31. Особенности работы и внутреннего устройства Express.js: веб-сайт. URL: <https://habr.com/ru/company/ruvds/blog/414079/> (дата звернення – 25.11.2021)
32. PostgreSQL: веб-сайт. URL: <https://www.postgresql.org/> (дата звернення – 01.11.2021).

33. Getting Started with Sequelize and Postgres: веб-сайт. URL: <https://dev.to/nedsoft/getting-started-with-sequelize-and-postgres-emp> (дата звернення – 01.11.2021).

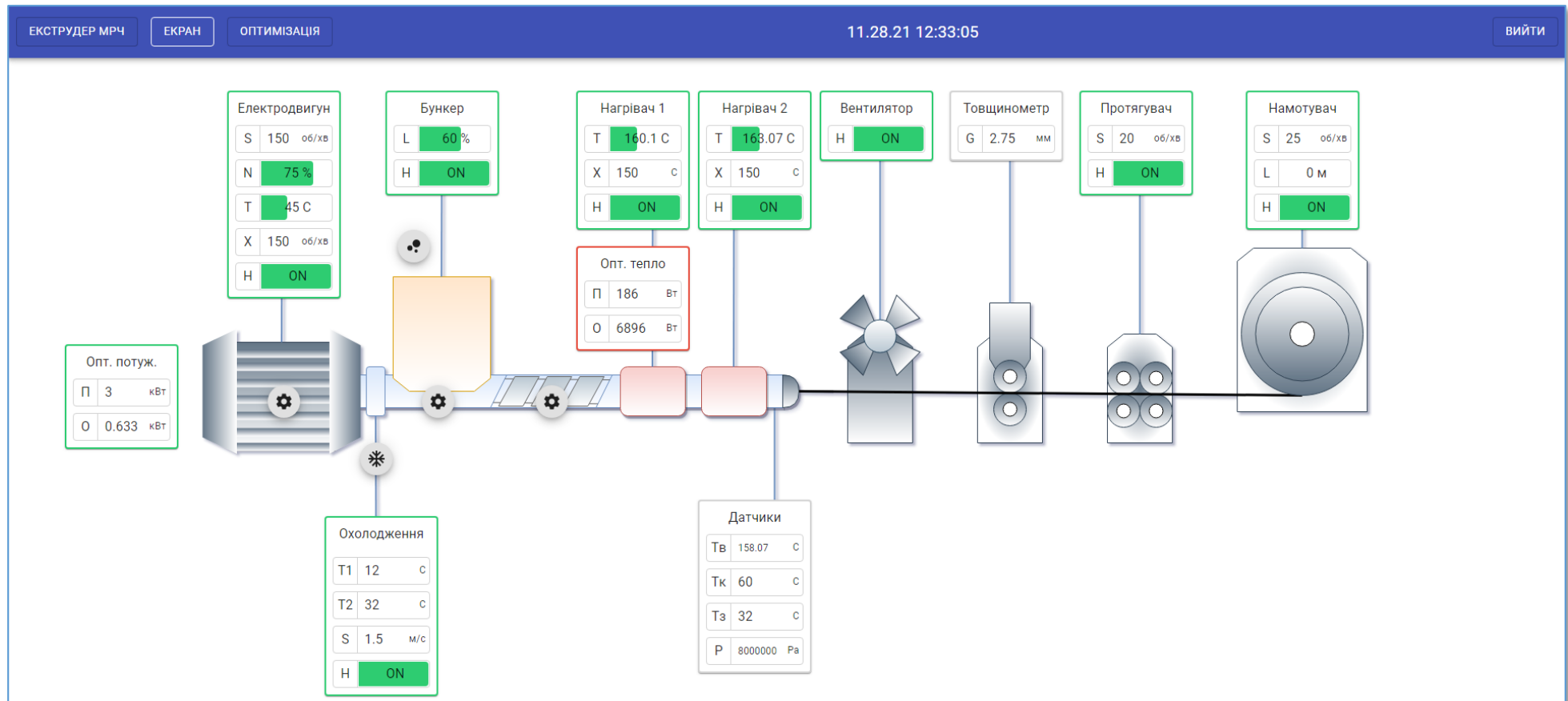
34. SerialPort: веб-сайт. URL: <https://serialport.io/> (дата звернення – 01.11.2021).

35. Баріщенко О.М., Овчинникова І.А. Курсова науково-дослідна робота : методичні вказівки для студентів напряму підготовки 0925 "Автоматизація і комп'ютерно-інтегровані технології" кваліфікації 8.092501 "Автоматизоване управління технологічними процесами" магістр. Запоріжжя: ЗДІА, 2010. 28 с.

ДОДАТОК А – Принципова електрична схема управління



ДОДАТОК Б – Вікно мнемосхеми екструзійної лінії Web-додатку СППР



ДОДАТОК В – Вікно монітору реального часу Web-додатку СППР

ЕКСТРУДЕР МРЧ ЕКРАН ОПТИМІЗАЦІЯ 11.28.21 5:57:00 ВИЙТИ

Електродвигун (шнек)

Швидкість (об/хв)

Потужність (%)

Температура (C)

Контролер

Стан (I/O)

Завдання (об/хв)

SAVE

Нагрівач 1

Температура (C)

PID регулятор

Стан (I/O)

Завдання (об/хв)

Кр

Ki

Kd

SAVE

Нагрівач 2

Температура (C)

PID регулятор

Стан (I/O)

Завдання (об/хв)

Кр

Ki

Kd

SAVE

Датчики

Темп. розплаву (C)

Тиск розплаву (Pa)

Товщина (мм)

Бункер

Рівень (%)

Автозаповнення

Стан (I/O)

SAVE

Вентилятор

Реле

Стан (I/O)

SAVE

Протягувач

Швидкість (об/хв)

Контролер

Стан (I/O)

Завдання (об/хв)

SAVE

Охолодження

Т води на вході (C)

Т води на виході (C)

Контролер

Швид. цирк. води (м/с)

Стан (I/O)

SAVE

Намотувач

Швидкість (об/хв)

Довжина (м)

Контролер

Стан (I/O)

Завдання (об/хв)

SAVE

ДОДАТОК Г – Графічний інтерфейс СППР

Вкладка “Кількість теплоти”

ЕКСТРУДЕР МРЧ
ЕКРАН
ОПТИМІЗАЦІЯ
12.13.21 9:09:45
ВИЙТИ

КІЛЬКІСТЬ ТЕПЛОТИ
ПОТУЖНОСТЬ

<p style="text-align: center; font-weight: bold;">Тепло, що витрачається на матеріал, Вт</p> <div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;"> Матеріал ABS </div> <p>Теплоємність матеріалу: 1935 Дж/(кг*К) В'язкість розплаву при переробці: 1748 Па*с</p> <p>Витрата матеріалу, кг/год <input style="width: 100%;" type="text" value="2"/></p> <p>Т полімеру на вході, С <input style="width: 100%;" type="text" value="24"/></p> <p>Т полімеру на виході, С <input style="width: 100%;" type="text" value="160"/></p> <div style="background-color: #004a99; color: white; padding: 5px; text-align: center; margin-top: 10px;"> Результати розрахунків: 146 Вт </div>	<p style="text-align: center; font-weight: bold;">Тепло, що поглинається системою охолодження, Вт</p> <p>Площа поперечного перетину, м2 <input style="width: 100%;" type="text" value="0.006"/></p> <p>Т води на вході, С <input style="width: 100%;" type="text" value="12"/></p> <p>Т води на виході, С <input style="width: 100%;" type="text" value="32"/></p> <p>Швидкість циркуляції води, м/с <input style="width: 100%;" type="text" value="1.5"/></p> <div style="background-color: #004a99; color: white; padding: 5px; text-align: center; margin-top: 10px;"> Результати розрахунків: 3554 Вт </div>	<p style="text-align: center; font-weight: bold;">Тепло, що витрачається у навколишнє середовище, Вт</p> <p>Діаметр корпусу з ізоляцією, мм <input style="width: 100%;" type="text" value="0.4"/></p> <p>Довжина корпусу, м <input style="width: 100%;" type="text" value="1.7"/></p> <p>Температура поверхні корпусу, С <input style="width: 100%;" type="text" value="60"/></p> <p>Температура навколишнього середовища, С <input style="width: 100%;" type="text" value="24"/></p> <div style="background-color: #004a99; color: white; padding: 5px; text-align: center; margin-top: 10px;"> Результати розрахунків: 942 Вт </div>	<p style="text-align: center; font-weight: bold;">Тепло, що виділяється при роботі шнека, Вт</p> <p>Крок нарізки шнека, см <input style="width: 100%;" type="text" value="1.5"/></p> <p>Діаметр шнека, см <input style="width: 100%;" type="text" value="2"/></p> <p>Середня глибина нарізки шнека, см <input style="width: 100%;" type="text" value="0.425"/></p> <p>Величина зазору між гребнем і шнеком, см <input style="width: 100%;" type="text" value="0.2"/></p> <p>Довжина напорної частини шнека, см <input style="width: 100%;" type="text" value="60"/></p> <p>Об'ємна витрата екструдера, см3/с <input style="width: 100%;" type="text" value="17.7"/></p> <p>Тиск в екструдері, Па <input style="width: 100%;" type="text" value="8000000"/></p> <p>Частота обертання шнека, об/с <input style="width: 100%;" type="text" value="2.5"/></p> <div style="background-color: #004a99; color: white; padding: 5px; text-align: center; margin-top: 10px;"> Результати розрахунків: 1288 Вт </div>
---	---	---	--

Кількість теплоти: **3354 Вт**

ЗКИНУТИ ЗМІНИ

Вкладка “Потужність”

ЕКСТРУДЕР МРЧ ЕКРАН ОПТИМІЗАЦІЯ 12.13.21 9:09:45 ВИЙТИ

КІЛЬКІСТЬ ТЕПЛОТИ ПОТУЖНОСТЬ

Потужність, що витрачається на обертання шнека, Вт

Діаметр шнека, см — 2

Довжина шнека, см — 48

Крок нарізки шнека, см — 1.5

Величина зазору між гребнем і шнеком, см — 0.2

Тиск в екструдері, Па — 8000000

Частота обертання шнека, об/с — 2.5

Діаметр серцевини шнека в зоні плавлення, см — 1

Діаметр серцевини шнека в зоні пластифікації, см — 1

Діаметр серцевини шнека в зоні дозування, см — 1.2

Глибина нарізки шнека в зоні плавлення, см — 0.5

Глибина нарізки шнека в зоні пластифікації, см — 0.5

Глибина нарізки шнека в зоні дозування, см — 0.4

Результати розрахунків: 633 Вт

ЗКИНУТИ ЗМІНИ