

**Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»**

Інститут електроенергетики
(інститут)

Факультет інформаційних технологій
(факультет)

Кафедра інформаційних технологій та комп'ютерної інженерії
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи ступеня магістра
(бакалавра, спеціаліста, магістра)

студента Матлашевський Микола Миколайович
(ПІБ)

академічної групи 123М-19-1
(шифр)

спеціальності 123 «Комп'ютерна інженерія»
(код і назва спеціальності)

за освітньо-професійною програмою «Комп'ютерна інженерія»
(офіційна назва)

на тему «Комп'ютерна система моніторингу енергетичних показників
видобувного комбайну вугільної шахти»
(назва за наказом ректора)

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
Кваліфікаційної роботи	проф. Цвіркун Л.І.			
розділів:				
Теоретичний розділ	проф. Цвіркун Л.І.			
Синтез системи	доц. Ткаченко С.М.			
Розроблення програмного забезпечення	ас. Бешта Л.В.			
Експериментальний розділ	доц. Ткаченко С.М.			
Рецензент				
Нормоконтролер	проф. Цвіркун Л.І.			

**Дніпро
2020**

ЗАТВЕРДЖЕНО:
завідувач кафедри
Інформаційних технологій
та комп'ютерної інженерії
(повна назва)

« _____ » грудня 2020 року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня магістр
(бакалавра, спеціаліста, магістра)

студенту Матлашевський М.М. академічної групи 123М-19-1
(прізвище та ініціали) (шифр)

спеціальності 123 «Комп'ютерна інженерія»

за освітньою-професійною програмою 123 «Комп'ютерна інженерія»
(офіційна назва)

на тему «Комп'ютерна система моніторингу енергетичних показників
видобувного комбайну вугільної шахти»,

затверджену наказом ректора НТУ «Дніпровська політехніка» від _____ № _____

Розділ	Зміст	Термін виконання
<i>Стан питання і завдання дослідження</i>	<i>На основі матеріалів виробничих практик, інших науково-технічних джерел обґрунтувати необхідність досліджень роботи комп'ютерної системи управління і мережі рівня підприємства.</i>	<i>01.10.2020 р.</i>
<i>Теоретична частина</i>	<i>На основі науково-технічних джерел проаналізувати теоретичні методи дослідження комп'ютерних мереж</i>	<i>15.10.2020 р.</i>
<i>Синтез комп'ютерної системи підприємства</i>	<i>На основі характеристик комп'ютерної системи розробити програму людино-машинного інтерфейсу системи управління та налаштування комп'ютерної мережі.</i>	<i>01.11.2020 р.</i>
<i>Експериментальний розділ</i>	<i>Розробити математичну модель комп'ютерної мережі, провести обчислювальний експеримент і розробити рекомендації що до покращення параметрів.</i>	<i>15.11.2020 р.</i>
<i>Графічна частина</i>	<i>Графічні результати роботи подати у вигляді рисунків схем таблиць на 10 арк. формату А4.</i>	<i>05.12.2020 р.</i>

Завдання видано _____

(підпис керівника)

проф. Цвіркун Л.І.

(прізвище, ініціали)

Дата видачі 01 жовтня 2020 р.

Дата подання до екзаменаційної комісії

10.12.2020 р.

Прийнято до виконання _____

(підпис студента)

Матлашевський М.М.

(прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: ____ с., ____ рис., ____ табл., ____ дод., ____ джерел.

Об'єкт дослідження система управління швидкістю очисного комбайна.

Наукове обґрунтування методів аналізу й синтезу комп'ютерної системи моніторингу енергетичних показників видобувного комбайну вугільної шахти.

Отримана модель об'єкта управління на підставі аналітичного опису, виходячи з параметрів електродвигунів подачі очисного комбайна. В результаті досліджень встановлено, що на етапі зміни уставки швидкості подачі комбайна найкращі результати по управлінню швидкістю дає ПД-регулятор, а на етапі стабілізації швидкості найкращі результати дає ПДД-регулятор.

Вперше для управління швидкістю подачі комбайна було використано комбінований регулятор на підставі ПД-регулятора та ПДД-регулятора. Регулятори були розроблені таким чином, щоб їх перемикання виконувалося безударно. Отримана система управління на базі такого регулятора значно перевищує по якості інші системи управління.

На базі безперервного регулятора було розроблено його цифрову модель на підставі якої розроблено програмне забезпечення для програмного програмованого логічного контролера zenon Logic.

Також було розроблено програмне забезпечення людино-машинного інтерфейсу яке відображає переміщення очисного комбайна, його енергетичні показники та реєструє аварійні ситуації.

Розроблене програмне забезпечення може бути використано у складі системи управління очисним комбайном та системи моніторингу роботи очисних комбайнів в вугільній промисловості.

Використання розробленого програмного забезпечення може дозволити підвищити видобуток вугілля за рахунок зменшення простою очисних комбайнів.

**ОЧИСНИЙ КОМБАЙН, СИСТЕМА УПРАВЛІННЯ, ПІД-РЕГУЛЯТОР,
SCADA СИСТЕМА**

ABSTRACT

Explanatory note: _____ pp., _____ figures, _____ tables, _____ annexes, _____ sources.

The object of the study is the speed control system of a cleaning combine. The purpose of the work is the development of a speed control system for a cleaning combine.

Scientific substantiation of methods of analysis and synthesis of a computer system for monitoring the energy performance of a coal mine mining combine..

As a result of the study it was found that at the stage of changing the set point of the feed rate of the combine, the best results on speed control are given by the PD controller, and at the stage of speed stabilization, the best results are obtained by the PID controller.

For the first time to control the feeding speed of the combine, a combined regulator based on the PD controller and the PID controller was used. The regulators were designed in such a way that their switching would be performed unstressed. The received control system on the basis of such a regulator is considerably ahead of the known systems in terms of quality.

On the basis of a continuous controller, its digital model was developed, on the basis of which software was developed for the programmed programmable logic controller zenon Logic.

Also, software was developed for the human-machine interface that displays the movement of the harvesting machine, its energy indicators and registers emergencies.

The developed software can be used as part of a control system for a cleaning combine and a system for monitoring the work of harvesting combines in the coal industry.

CLEANING COMBINE, CONTROL SYSTEM, PID-REGULATOR, SCADA SYSTEM

ЗМІСТ

Вступ.....	8
1 Стан питання і постановка задачі.....	10
1.1 Загальна характеристика об'єкта управління.....	10
1.2 Характеристика і організаційна структура виробництва.....	11
1.3 Огляд технології процесу видобутку вугілля.....	13
1.4 Огляд і характеристика об'єкта впровадження	18
1.5 Аналіз режимів роботи обладнанням очисного комбайна	22
1.5.1 Огляд принципів управління швидкістю очисного комбайну	23
1.5.2 Аналіз способів управління швидкістю подачі комбайна	24
1.6 Постановка завдання.....	26
1.7 Висновки по розділу	27
2 Теоретична частина.....	28
2.1 Обґрунтування схеми даних та процесів.....	28
2.2 Розрахунок основних характеристик моделі об'єкта управління.....	28
2.2.1 Критерії якості функціонування системи управління.....	33
2.3 Висновки по розділу	33
3 Синтез комп'ютерної системи	35
3.1 Постановка завдання дослідження.....	35
3.2 Синтез системи управління.....	35
3.3 Розробка принципової схеми системи керування.....	42
3.3.1 Розробка вимог до принципової схеми системи керування	42
3.3.2 Аналіз входів автоматизованої системи контролю	43
3.3.3 Вибір елементної бази системи управління	45
3.3.4 Реалізація принципової схеми системи контролю	48

	6
3.4 Робастність системи управління.....	49
3.5 Розробка програмного забезпечення.....	56
3.5.1 Призначення і область застосування програми	56
3.5.2 Обґрунтування технічних характеристик програми	57
3.5.2.1 Постановка завдання на розробку програми.....	57
3.5.2.2 Опис алгоритму функціонування програми.....	58
3.5.3 Опис і обґрунтування вибору методу організації вхідних та вихідних даних	59
3.5.3.1 Опис і обґрунтування вибору та складу технічних та програмних засобів	59
3.5.4 Використані технічні засоби	61
3.5.4.1 Виклик і завантаження програми	61
3.5.4.2 Вхідні і вихідні дані	61
3.5.4.3 Використані програмні засоби	61
3.5.4.4 Використані технічні засоби	61
3.5.4.5 Виклик і завантаження програми	61
3.5.4.6 Вхідні і вихідні дані	61
3.5.5 Очікувані техніко-економічні показники	62
3.6 Висновки по розділу	62
4 Розробка програмного забезпечення.....	63
4.1 Цифрова модель системи управління	63
4.2 Висновки по розділу	66
5 Експериментальний розділ.....	67
5.1 Розробка програмного забезпечення системи управління.....	67
5.2 Розробка програмного забезпечення людино-машинного інтерфейсу	69
5.3 Перевірка функціонування програмного забезпечення	74
5.4 Висновки по розділу	77
Висновки	78
Перелік посилань.....	80

	7
Додаток А Програмні модулі системи управління.....	82
Додаток Б Програмне забезпечення системи управління.....	90
Додаток В Програмне забезпечення людино-машинного інтерфейсу.....	96
Додаток Г Програмне забезпечення тестування.....	100

ВСТУП

На сьогоднішній день 50 % електроенергії в Україні виробляється на теплоелектростанціях з кам'яного вугілля. З урахуванням падіння цін на світових ресурсних ринках і зростання конкуренції між виробниками питання зменшення енерговитрат вийшло на перший план.

Основними вимогами до сучасної системи управління швидкістю подачі комбайна є відсутність автоколивань, швидке відпрацювання збурень які зв'язані зі зміною навантаження на робочий орган, якісна підтримка швидкості переміщення комбайну, контроль за струмом, напругою та температурою двигунів як подачі комбайну, так і різання.

Крім того останні дослідження показують, що наявність якісної системи управління швидкістю подачі комбайна дозволить більш ефективно використовувати робочий час за рахунок управління кількістю метану в повітрі. Управління кількістю метану в повітрі може бути забезпечено за рахунок зміни швидкості подачі комбайну. Як показують дослідження якісне управління швидкістю може забезпечити оптимальну кількість видобутого вугілля без простоїв через підвищену кількість метану в повітрі.

Мета роботи і завдання дослідження.

Мета даної кваліфікаційної роботи – розробка автоматизованої системи управління швидкістю очисного комбайну вугільної шахти.

Завданням даної роботи є проектування та розробка системи управління, яка дозволить управляти швидкістю комбайну без автоколивань та статичної похибки.

Об'єкт дослідження

Об'єктом дослідження є автоматизована системи управління швидкістю подачі комбайна.

Предмет і методи дослідження

Предмет дослідження - автоматизована системи управління швидкістю подачі комбайна, методи дослідження - наукове обґрунтування вибору методів

аналізу й синтезу комп'ютерної системи моніторингу енергетичних показників видобувного комбайну вугільної шахти.

Отримання моделі об'єкта управління на підставі аналітичного опису, виходячи з параметрів електродвигунів подачі очисного комбайна.

Ідея роботи

Розробку автоматизованої системи управління швидкістю очисного комбайну, треба проводити на базі вже існуючого обладнання та пристроїв, шляхом доопрацювання та удосконалення ланок технологічного процесу.

Розробити програмне забезпечення людино-машинного інтерфейсу, яке відобразатиме переміщення очисного комбайна, його енергетичні показники та реєстрація аварійних ситуацій.

Використання розробленого програмного забезпечення дозволить підвищити видобуток вугілля за рахунок зменшення простою очисних комбайнів.

Обрана тема актуальна для вугільних шахт та спрямована на зниження собівартості камінного вугілля.

1 СТАН ПИТАННЯ І ПОСТАНОВКА ЗАДАЧІ

1.1 Загальна характеристика об'єкта управління

В Україні зосереджено близько 4 % світових запасів вугілля. Основний видобуток вугілля ведеться в Луганській, Донецькій та Дніпропетровській областях, причому на Луганщину припадає 43 % видобутку. Донецька область займає друге місце з 27 %. На Дніпропетровщину припадає 23 %, а на Львівську та Волинську всього 7 % загального обсягу видобутку.

Україна займає 12-е місце в світі щорічно видобуваючи понад 70 мільйонів тон. Основними споживачами вугілля в Україні є енергетична, хімічна та металургійна промисловості. За останні роки найбільше зростання попиту споживання вугілля продемонстрував енергетичний сектор промисловості. Сумарно на виробництво коксу та електричної енергії витрачається більше 69 % від усього обсягу видобутого вугілля.

На сьогоднішній день для забезпечення істотного підвищення вуглевидобутку можуть бути задіяні 67 резервних ділянки із запасами 13,1 мільярд тон і розрахунковими обсягами видобутку 124,9 мільйона тон на рік.

Видобуток вугілля може здійснюватися двома способами: відкритим та закритим. Відкритий спосіб використовується при глибині залягання пластів до 100 метрів [1]. При такому способі видобутку знімається верхній шар ґрунту, який називається розкриттям. Відкриття в залежності від рельєфу та ґрунту здійснюється скреперами, бульдозерами, роторними екскаваторами, драглайнами. Далі проводять дроблення породи за допомогою екскаваторів, дробарок, водяних гармат, бульдозерів, драглайнів, іноді доводиться застосовувати буропідривну відбійку вугілля.

Переваги видобутку вугілля відкритим способом:

- швидкість видобутку;
- економічність;
- відносна безпека.

Недоліки:

- відчутна шкода навколишньому середовищу;
- вугілля, видобуте у такий спосіб, містить багато домішок.

Закритий спосіб видобутку застосовується при великій глибині залягання пластів вугілля. На рівнинній місцевості до місця залягання вугілля прокладають вертикальні або горизонтальні стовбури, з яких формується шахта.

Видобуток вугілля закритим способом викується в лавах, в одній шахті може бути від однієї до кількох лав [1]. В лаві вугілля добувається за допомогою очисного комбайна, після чого транспортується до скіпового підйому, який підіймає його на поверхню.

Переваги видобутку вугілля закритим способом:

- видобувається вугілля високої якості;
- приносить менше шкоди навколишньому середовищу.

Недоліки:

- найбільш небезпечний спосіб видобутку;
- вимагає істотних фінансових вкладень.

Найбільш поширеним способом видобутку вугілля на території України є закритий спосіб, що пов'язано з тим, що основні запаси вугілля на Україні залягають на глибині від 250 до 1300 метрів, а потужність пласта складає від 0,5 до 1,3 метрів.

1.2 Характеристика і організаційна структура виробництва

Розглянемо організаційну структуру відділу головного енергетика вугільної шахти (рис. 1.1).

Організаційна структура шахти є функціонально-лінійною структурою (штабне управління) та являє собою комбінацію лінійної структури з системою виділення певних функцій.

Лінійно-штабна структура є гарним проміжним ступенем при переході від лінійної структури до більш ефективних структур.

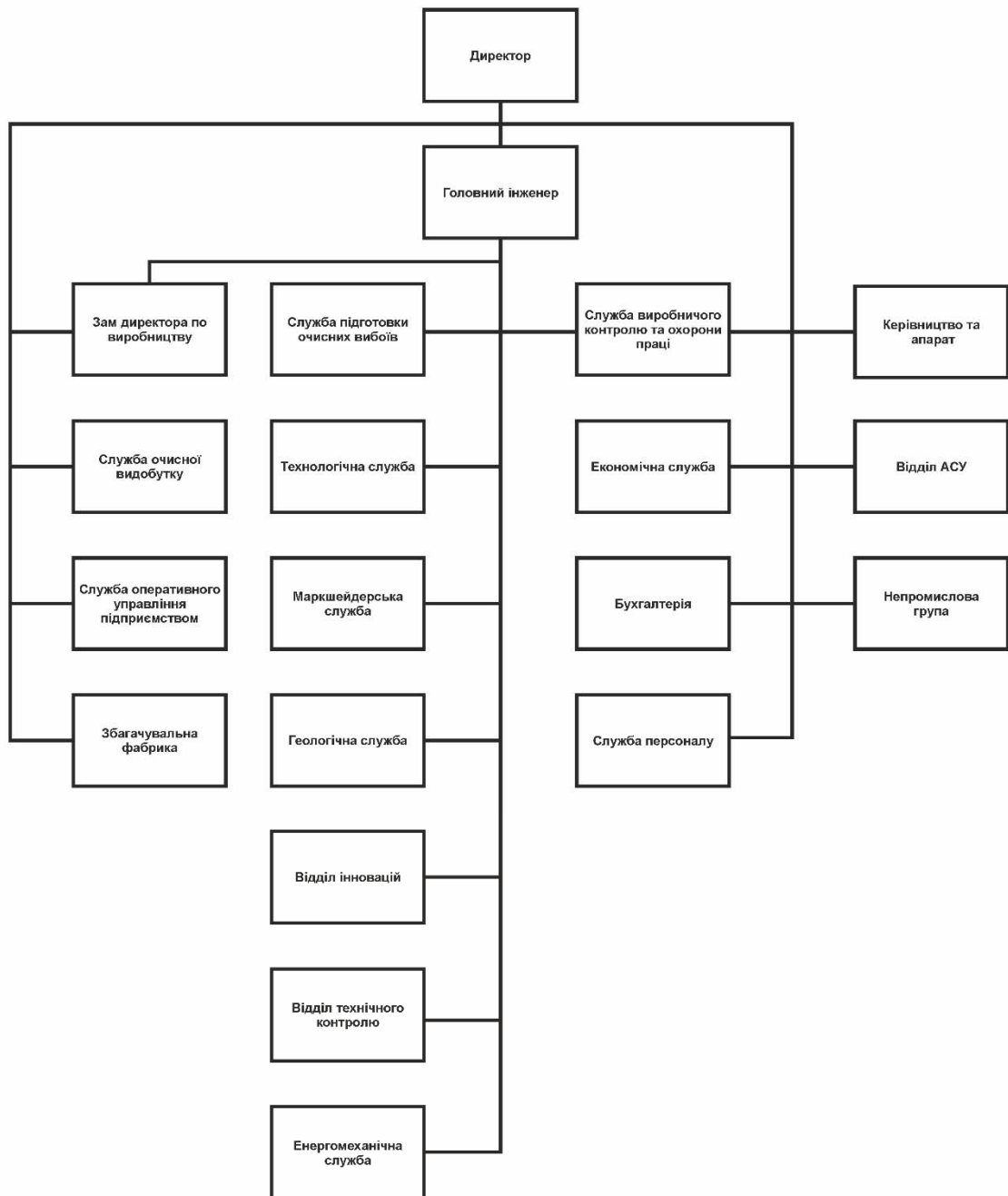


Рисунок 1.1 – Організаційна структура

На чолі організаційної структури знаходиться директор. Головний інженер підприємства є першим заступником директора та знаходиться в його безпосередньому підпорядкуванні.

Також директору підпорядковуються: заступник директора з виробництва, який керує службами: очисної видобутку, оперативного управління виробництвом та збагачувальної фабрикою.

У безпосередньому підпорядкуванні головного інженера знаходяться:- служба підготовки очисних вибоїв, технологічна служба, маркшейдерська служба; геологічна служба, відділ інновацій, відділ технічного контролю, енергомеханічна служба, служба виробничого контролю та охорони праці, економічна служба, бухгалтерія, служба персоналу та служба з господарських питань.

Безпосередньо директору підпорядковується керівництво та апарат при керівництві: провідний спеціаліст по роботі з цінними паперами, начальник відділу ГО та НС, старший інспектор з діловодства, архіваріус, відділ АСУ, непромислова група.

1.3 Огляд технології процесу видобутку вугілля

Незважаючи на стрімкий розвиток технологічного прогресу, підземний метод видобутку вугілля залишається надзвичайно складним і трудомістким сьогодні. Перший успішний досвід промислового застосування механізованих комплексів датується початку сіддесятих ХХ століття, і до цих пір цей вид гірничого обладнання продовжує розвиватися.

Створення та впровадження механізованих комплексів в практику підземного видобутку вугілля відіграло виняткову роль в технічному переозброєння вугільної промисловості, послужила потужним поштовхом для розвитку шахт, призвело до збільшення технічного рівня всіх ланок підземних технологій виробництва. Механізовані комплекси використовується в калійних шахтах і в екстракції ряду інших мінералів в основному з пластичним будовою.

При переході до комплексної механізації очисних робіт, в принципі вирішувалися проблеми створення гідро-інфікованих мобільних секцій кріпів, очищення пересувних скреперів, вузькозахватних комбайнів, струменів. Кінематичне з'єднання цих механізмів та обладнання, що забезпечують

послідовне пересування в циклічних режимах цілого комплексу машин та обладнання.

Очищення механізованого комплексу складається з базової (машини, набору секцій з гідро-інфікованої рухливої рухомої тяги, скребкового конвеєра) та допоміжного обладнання.

Скоординована робота всіх механізмів та обладнання в базових і підтримуючих режимах забезпечується інтегрованою електрикою, освітленням, сигналізатори і системами управління, гідросистемою, що забезпечує потужність переміщення секцій кріпів і конвеєра з переїздом машини, системи повітряного управління у виробництві, зокрема кількісний вміст метану, телефону і гучномовця. Основною вимогою для цього обладнання та систем є забезпечення довгострокової високопродуктивної і безпечної роботи в зміні.

Очисні механізовані комплекси класифікуються за такими основними ознаками:

- для похилих і похилих шарів з кутом нахилу до 35 градусів, а для крутих і крутих шарів з кутом нахилу більше 35 градусів;
- ємністю пласта - для тонких, середніх і потужних шарів;
- у вигляді видобутих мінералів - для рудників (вугледобувних), для шахт (наприклад, видобування калійних);
- за призначенням - загальна мета, для використання в шахтах і рудниках в комбайні, спеціального призначення, при використанні нестандартних методів очищення роботи: з пневматичним або гідро-прокладкою, з виходом вугілля при роботі з потужними шарами у забою на конвеєрній стрічці через люки в паркані щита секцій кріпів (рис. 1.2);
- за типом кінематичних з'єднань між машинами і обладнанням комплексного-агрегатного, повного і комбінованого (агрегатного-повного).

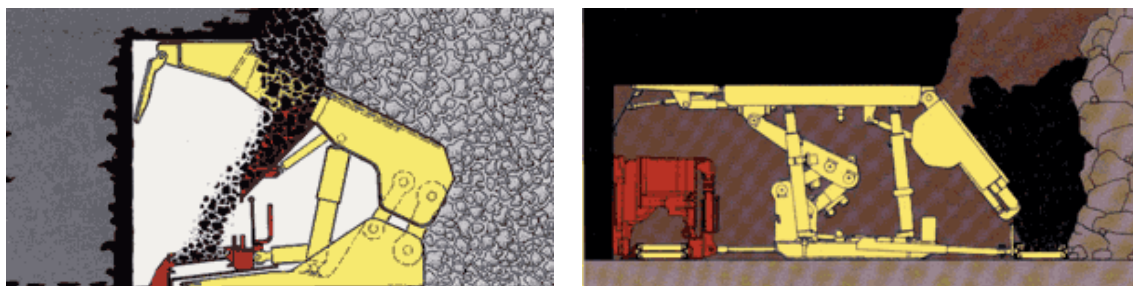


Рисунок 1.1 - Механізована схема з виходом вугілля з даху в забій приміщення

В даний час використовують потужні регульовані електроприводи подачі і різання, що забезпечують високу інтенсивність робіт при швидкості до 10...45 м/хв. Потужність двигуна різання варіюється від 75 кВт до 600 кВт в залежності від розміру і розташування комбайна, а подача приводу варіюється від 10 кВт до 75... 125 кВт з встановленою потужністю на комбайні від 85 кВт до 1 940 кВт або більше.



Рисунок 1.2 - Сучасний комплекс

Комбайни оснащені ручним системами прямого контролю (використовуються при налаштуванні комбайнів і тестуванні їх режимів роботи), дистанційних систем управління радіоуправлінням (від переносних дистанційних), з програмним автоматизованим контролем при високих швидкостях і потужності комбайна.

Датчики системи і мікропроцесорне обладнання контролюють стан комбайнів вузлів і систем, контролюють і оцінюють режим роботи, і надають необхідну інформацію до оператора комбайна, лавового оператора, диспетчеру.

Регулювання руху вздовж границі вугільного шва і ґрунту найбільш ефективно виконується спеціальними гідравлічними циліндрами, встановленими з боку конвеєрного ремня і зміною нахилу конвеєра для очищення в залежності від відхилення опори конвеєрна площину з ґрунту.



Рисунок 1.3 – Ланцюжний пристрій

Ланцюжний пристрій забезпечує всі кінематичні з'єднання, забезпечуючи напрямок циклічних рухів машини, секції лавового кріплення і кріплення пар, шляхової апаратури.

На задній частині конвеєра встановлюються рейки для подачі механізму (рис. 1.3) або канали, передбачені для тягових і непрацюючих гілок ланцюга заgonу різання.

Механізоване кріплення в процесі взаємодії з вуглецевими масивами зазвичай виконують три основні функції: управління покрівлею, активне обвал над забійним простором і огороження забою від руйнування порід покрівлі. Тому шлях і характер взаємодії з боковими скелями (тобто ступенем виконання основних функцій) виділяють види механізованої кріпі: підтримуюча, підтримуюча-захисна, захисна-опора.

За конструкційними схемою, механізовані кріпі розрізняються: агрегатна, повна і повна-агрегатна.

У агрегатних кріпленнях структурні підрозділи (секції) мають загальну групу кінематичного з'єднання по довжині лави, фіксуючи інтервали розташування секцій, забезпечуючи напрямок циклічного пересування і кінематичне взаємодія секцій кріплення через стрічковий конвеєр, або базовий брус. У той же час рух секцій здійснюється за рахунок їх регулярного витягування до базового конвеєра.

У повній механізованій кріпці, секції неоднорідні інтегровані в комплекти і не мають загальної групи зв'язку по довжині лави. Рух таких розділів відбувається шляхом рухомої секції.



Рисунок 1.4 – Дворядні одиничні секції механізованих комплексів

Ефективність комплексного механізованого комплексу оцінюється системою абсолютних і специфічних техніко-технічних і економічних показників, що забезпечують порівняльну оцінку як самої апаратури, так і ефективності її обладнання. Проте, гірничо-геологічні, та гірничо-технологічні умови функціонування механізованих комплексів суттєво різняться, що до показників ефективності комплексного механізованого забою.

Чим вище інтенсивність очищення, тим більше роль для ефективної і стійкої роботи відіграють системи контролю машин, конвеєра, механізованої кріпці.

Сучасні цифрові програмовані системи автоматизації забезпечують контроль технічного стану, та режимів роботи всіх механізмів, обладнання та

систем з виходом необхідної інформації, включаючи візуальний дисплей, операторам машини та диспетчеру, а також вибір режимів роботи від пульта дистанційного керування і введення обмежень, перехід при необхідності від автоматичного до дистанційного або ручного управління.

1.4 Огляд і характеристика об'єкта впровадження

При закритому способі видобутку вугілля використовуються очисні комбайни спільно з механізованими кріпленнями і забійними конвеєрами які разом утворюють механізований очисний комплекс [1-5].

Очисні комбайни призначені для відділення від масиву вугілля і навантаження відокремленої гірської маси на забійний конвеєр [1-3]. Очисні комбайни для крутих і круто похилих пластів забезпечують виконання тільки відділення вугілля від масиву, оскільки транспортування гірської маси для цих умов здійснюється самопливом за рахунок гравітаційних сил.

Основним зовнішнім середовищем для очисних комбайнів є:

- гірський масив як система природного характеру, що включає підсистеми руйнуйомого вугільного пласту, його ґрунту і покрівлі;
- при виїмці пологопохилих пластів – забійний конвеєр як технічна система, що включає підсистему граткового ставу (забезпечує для очисних комбайнів функції опорного і направляючого механізмів базування, функції сприйняття і передачі тягового зусилля) і підсистему транспортування відокремленої гірської маси.

До сучасних очисних комбайнів висуваються наступні вимоги:

- високий рівень автоматизації комбайнів як мехатронізованих або мехатронних систем при управлінні ними, забезпеченні необхідних захистів і блокувань, моніторинг технологічної та діагностичної інформації та виконання ряду інших функцій сервісного характеру;
- досить повне охоплення ймовірних областей використання за характеристиками руйнації вугільних пластів і механічними характеристиками породних прошарків і включень;

- використання раціональних технологічних схем роботи з максимальною механізацією і мінімальним обсяг кінцевих і допоміжних операцій;
- хороша входження в гіпсометрію пластів.

На Україні широкого поширення набув сучасний вузькозахватний очисний комбайн УКД 200-250 який відповідає всім перерахованим вимогам (рис. 1.5) [1, 2].

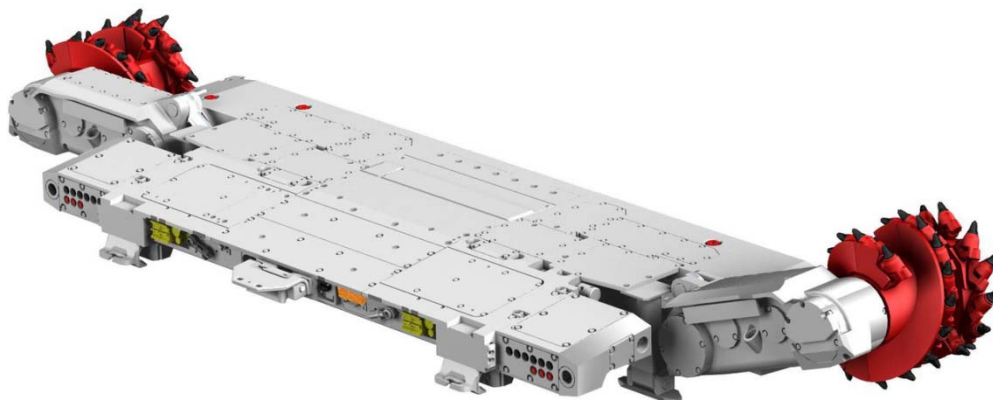


Рисунок 1.5 – Очисний комбайн

Очисний комбайн УКД 200-250 (табл. 1.1) призначений для виїмки вугілля в очисних вибоях, які порушуються з напрямком простягання пластів потужністю 0,85 - 1,3 м з кутом падіння до 35° , а також по падінню і повстання - до 10° , при опірності вугілля різанню до 360 кН/м.

Комбайн УКД 200-250 може застосовуватися в складі механізованих комплексів типу 1МКДД, 1МДТ, МДМ, 1МКД80, 1МКД90, оснащених скребковим конвеєром типу СПЦ26 (СПЦ163), СП26У (СП251) [1-4].

Комбайн оснащений винесеною системою подачі, має ширину захвату 0,63, 0,8 або 0,9 м і може оснащуватися шнеками діаметром 800, 900 мм з тангенціальними різцями типу РГ501.

Таблиця 1.1 – Технічні характеристики очисного комбайну

№	Назва	Опис
1	Застосування по виймаємо потужності пласта	850-1300 мм
2	Продуктивність при опірності вугілля різанню 120 кН/м	5,5 т/мин

3	Продуктивність при опірності вугілля різанню 240 кН/м	4,4 т/мин
4	Продуктивність при опірності вугілля різанню 360 кН/м	3,3 т/мин
5	Сумарна номінальна потужність приводів	330 кВт
6	Потужність приводів виконавчих органів	220 кВт
7	Потужність приводу винесення системи подачі	2x55 кВт
8	Номінальна напруга	1140 В
9	Максимальна робоча швидкість подачі, не менше	5 м/мин
10	Максимальне тягове зусилля системи подачі, не менше	200 кН
11	Маса, не більше	14440 кг

Основними технічними і експлуатаційними особливостями комбайна є:

- висока надійність і ресурс, забезпечуються конструктивними параметрами і використанням високоміцних матеріалів;
- збільшена енергоозброєність приводу різання до 220 кВт;
- забезпечення мінімальної потужності пласта 0,85 м;
- зменшення непродуктивних витрат часу на усунення відмов за рахунок підвищення міцності зубчастих передач і підшипникових вузлів.

Основними конструктивними особливості комбайна є:

- розташування корпусу комбайна в уступі вибою;
- безфланцеве з'єднання основних силових вузлів;
- розрахункова довговічність силових елементів редуктора групи 15000 годин;
- наявність системи охолодження редукторів ріжучої частини;
- адаптована до верхняками кріплення конструкція порталу;
- оснащення шнеків системою пилоподавлення з захистом від фрикційного іскріння.

Спеціально для комбайнів українського виробництва серії УКД компанія ДОНГИПРОУГЛЕМАШ сумісно з компанією BARTEC розробила автоматизовану систему управління (рис. 1.6). Дана система застосовується на комбайнах УКД 200-250 [1]. До складу системи входять:

- перетворювач частоти шахтний DYNAVERT 250 / 1140-60da;
- головний контролер забою ВРІС;
- пульт управління забою SP 1.0.

Головний контролер забою ВІРС побудований на базі промислового контролера ІРС виробництва компанії BARTEC. Контролер має іскро-вибухобезпечне виконання, володіє двома рідкокристалічними та одним багаторядковим дисплеями для візуалізації процесу роботи і налаштування системи контролю та управління очисним комбайном. Налагодження та зміна режимів роботи системи здійснюється за допомогою вбудованої клавіатури. Зв'язок з контролером забезпечується іскробезпечним виконанням польової шини CAN відповідно до протоколу CANopen.

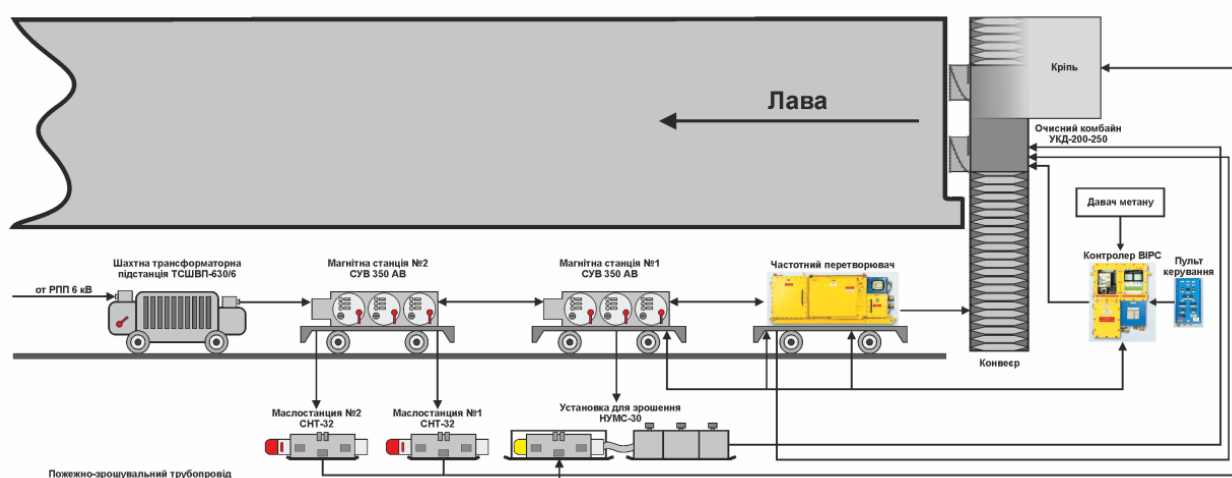


Рисунок 1.6 – Структура системи контролю очисного комбайну

Перетворювач частоти з проміжним контуром струму DYNAVERT 250/1140-60da призначений для управління багатодвигунним приводом з вирівнюванням моменту. Він забезпечує рекуперацію електроенергії в мережі та має цифрове керування. Зв'язок перетворювача частоти з головним контролером забою ВІРС забезпечується іскробезпечним виконанням польової шини CAN відповідно до протоколу CANopen.

Місцевий пульт управління SCP 1.0 призначений для дистанційного керування очисним комбайном. Він являє собою набір світлодіодної і кнопкової арматури які задовольняють вимогам іскро-вибухобезпеки. Зв'язок між пультом управління та головним контролером забою ВІРС забезпечується

іскробезпечним виконанням польової шини CAN відповідно до протоколу CANopen.

Основне призначення системи управління очисним комбайном підвищення продуктивності скорочуючи час простоїв комбайна завдяки застосуванню бортової системи діагностики. Система управління у своєму розпорядженні має потужний набір інструментів, включаючи розширені засоби діагностики та дружній до користувача багатомовний інтерфейс. Система дозволяє операторові програмувати кілька профілів роботи очисного комбайна для більш ефективного відпрацювання пластів зі змінними параметрами твердості вугілля та породи.

1.5 Аналіз режимів роботи обладнанням очисного комбайна

Управління роботою очисного комбайна реалізується іскро-вибухобезпечною системою управління компанії BARTEC. Дана система виконує функції управління комбайном, діагностику стану обладнання, захист від перевантажень електродвигунів виконавчих органів. Інформація про технологічний процес видобутку вугілля та стан вузлів комбайна відображається на моніторі головного контролера забою ВІРС та відображається світлодіодним арматурою пульта управління SP 1.0. Управління комбайном може здійснюватися оператором трьома способами: з пульта розташованого безпосередньо на комбайні, з пульта управління SP 1.0 або з пульта дистанційного керування.

Автоматизовані системи управління більш високого рівня (підприємства та виробництва) можуть отримати доступ до інформації про технологічний процес видобутку вугілля в конкретній виробці з головного контролера забою ВІРС по іскробезпечному виконанню польової шини CAN відповідно до протоколу CANopen.

Система виконує функції управління:

- електродвигунами поворотних редукторів;
- механізмами подачі;

- насосними установками;
- електромагнітними гальмами;
- електрогидророзподільниками.

Система керує роботою насосної станції, яка підтримує тиск в магістралі гідрообладнання комбайна, а також пристроїв охолодження і пилоподавлення. Пристрої охолодження відповідають за підтримання нормальної робочої температури електродвигунів, електромагнітних гальм, корпусів блоків приводів виконавчих органів. Пристрої пилоподавлення відповідають за гасіння вугільного пилу який утворюється при різанні та навантаженні гірської маси.

Однією з найважливіших функцій системи управління BARTEC є управління електродвигунами подачі очисного комбайна. Це пов'язано з тим, що при веденні видобувних робіт від швидкості комбайна безпосередньо залежить продуктивність комбайна та кількість метану що виділяється.

1.5.1 Огляд принципів управління швидкістю очисного комбайну

В системі автоматизованого контролю роботи очисного комбайна основними контрольованими параметрами є швидкість подачі комбайна і споживана потужність його електроприводів.

Для контролю положення очисного комбайна на вал приводу подачі встановлюється імпульсний давач, що дає один імпульс на оборот. На підставі імпульсів розраховується швидкість подачі комбайна:

$$v = 2\pi n j R, \quad (1.1)$$

де v – швидкість подачі комбайна (м/с), n – імпульси (імп/хв), j – коефіцієнт передачі редуктора, R – радіус передачі (м).

На підставі значення швидкості (1.1) розраховується переміщення комбайна:

$$l = \int_0^{\tau} v dt, \quad (1.2)$$

де l – переміщення комбайну (м), τ – поточний час (с), v – швидкість подачі комбайна (м/с), dt – крок квантування за часом.

Контроль споживаної потужності розраховується за значенням струму, вимірюваного на двох фазах і значенням напруги. На підставі значень струму визначається коефіцієнт корисної дії електроприводу, після чого розраховується значення потужності:

$$P = 3UI\cos\varphi, \quad (1.3)$$

де P – потужність (Вт), U – фазна напруга (В), I – фазний струм (А).

Відповідно до технологічного процесу видобутку вугілля передбачається визначення оптимальної та максимально допустимої швидкості подачі очисного комбайна. Швидкість комбайна визначається відповідно до теорії різання по твердості порід, в яких виконується видобуток вугілля. Швидкість подачі обирається таким чином, що б не допускати критичних навантажень на устаткування які призводять до його підвищеного зносу.

1.5.2 Аналіз способів управління швидкістю подачі комбайна

Система управління швидкістю подачі очисного комбайну повинна вимірювати швидкість комбайну та підставі бажаного значення формувати керуючий вплив який задає швидкість обертання двигунів подачі та формує швидкість горизонтального переміщення комбайну.

Відомий спосіб управління комбайном, що забезпечує регулювання навантаження на нього шляхом зміни швидкості подачі, що включає завдання уставок струмів номінального і відсічення та безперервне вимірювання миттєвого струму двигуна приводу різання гірського комбайна. Управління швидкістю подачі здійснюється у функції величини відхилення миттєвого значення споживаного струму від заданого струму. Даний спосіб може викликати автоколивання швидкості подачі, які визначаються низькою швидкодією гідравлічних систем подачі.

Також є спосіб управління, що включає завдання уставок струму двигуна, рівного його номінальному, і струму відсічення, визначення ефективного струму двигуна і порівняння його з уставкою струму і підтримання їх рівності зміною швидкості подачі комбайна. Однак вказаний спосіб має недолік, що полягає в

тому, що в ньому не враховується функціональна залежність між швидкістю подачі і ефективним струмом двигуна приводу різання комбайна, що впливає на надійність і точність системи.

Крім того відомий спосіб управління за допомогою регулятора навантаження типу "Уран", що включає завдання номінального струму і струму відсічення електродвигуна приводу різання та безперервну зміну миттєвого значення споживаного струму. Цей спосіб має суттєві недоліки які виражені у тому, що він не захищає електродвигун від перегріву.

Відомий спосіб управління комбайном, що забезпечує регулювання навантаження приводу різання шляхом зміни швидкості подачі вугільного комбайна, що включає завдання уставок номінального струму і струму відсічення і безперервне вимірювання миттєвого струму електродвигуна приводу різання, використовуюваного для визначення ефективного струму, який порівнюють з струмом уставки, і підтримують їх рівність зміною швидкості подачі комбайна, порівнюють миттєві значення струму електродвигуна приводу різання гірського комбайна з заданим струмом відсічення і при перевищенні миттєвим струмом струму відсічення миттєво знижують швидкість подачі. Недоліком даного способу є низька надійність управління через відсутність прогнозування максимальних навантажень, що зумовило наявність другого каналу регулювання - відсічення.

Таким чином на даний час запропоновано багато способів управління швидкістю комбайну. Основними недоліками даних способів є наявність автоколивань в системі управління, що визвано використанням пропорційного керування. Крім того використання пропорційного керування призводить до того, що величина відхилення швидкості від заданою напряму залежить від навантаження на ріжучий орган комбайну. Також в багатьох системах управління відсутня паралельна система контролю температури двигуна, а виконується контроль тільки току двигуна який відповідає навантаженню. Дані рішення призводять до перегріву двигуна та виходу його зі строю.

Останні дослідження з управління швидкістю комбайну, показали, що при збільшені кількості метану в повітрі, яке викликано процесом різання не обов'язково повинно призводити до зупинки процесу добичі вугілля. Альтернативним рішенням є відповідне зменшення швидкості комбайну, що призводить до поступового зменшення кількості метану в повітрі при цьому не виникає зупинка добичі вугілля. Якщо система управління швидкістю комбайну має достатньо високі показники якості вона дозволяє продовжувати добичу вугілля в оптимальному режимі при даному рівні метану в повітрі. Таким чином розробка системи управління швидкістю комбайну нового покоління дозволить підвищити продуктивність роботи комбайну не вносячи змін до його конструкції.

1.6 Постановка завдання

Виходячи з аналізу існуючих рішень можливо зробити висновок, що на даний час немає системи управління швидкістю подачі комбайну яка би задовольняла усім вимогам.

Основними вимогами до сучасної системи управління швидкістю подачі комбайна є відсутність автоколивань, швидке відпрацювання збурень які зв'язані зі зміною навантаження на робочий орган, якісна підтримка швидкості переміщення комбайну, контроль за струмом, напругою та температурою двигунів як подачі комбайну, так і різання.

Крім того останні дослідження показують, що наявність якісної системи управління швидкістю подачі комбайна дозволить більш ефективно використовувати робочий час за рахунок управління кількістю метану в повітрі. Управління кількістю метану в повітрі може бути забезпечено за рахунок зміни швидкості подачі комбайну. Як показують дослідження якісне управління швидкістю може забезпечити оптимальну кількість видобутого вугілля без простоїв через підвищену кількість метану в повітрі.

Таким чином, мета даної кваліфікаційної роботи – розробка автоматизованої системи управління швидкістю очисного комбайну вугільної шахти.

Завданням даної роботи є проектування та розробка системи управління яка дозволить управляти швидкістю комбайну без автоколивань та статичної похибки.

Рішення даного завдання дозволить підвищити кількість вугілля, що видобувається за рахунок зменшення простою обладнання яке виникає через підвищену кількість метану в повітрі.

1.7 Висновки по розділу

Основними напрямками вирішення поставленого завдання є розробка математичної моделі об'єкта управління. Визначення чітких критеріїв за якими можливо буде коректно оцінити якість функціонування системи управління.

Визначення структури системи управління та типу регулятора для неї. На підставі обраної структури системи управління та типу регулятора виконання синтезу системи управління.

Перевірка якості функціонування синтезованої системи управління без збурень та при їх наявності. Розробка програмного забезпечення системи управління за отриманою моделлю та перевірка якості її функціонування.

2 ТЕОРЕТИЧНА ЧАСТИНА

2.1 Обґрунтування схеми даних та процесів

Згідно з завданням в даному розділі потрібно виконати розробку математичної моделі об'єкта управління в якості якого виступає електропривід подачі комбайна.

Крім того необхідно визначити чіткі критерії якості функціонування системи управління які можуть бути використані у подальшому для контролю якості функціонування синтезованої системи управління.

Таким чином з початку буде отримана математична модель об'єкта управління, після чого визначенні критерії якості функціонування системи управління.

2.2 Розрахунок основних характеристик моделі об'єкта управління

Видобувний комбайн має електрогідравлічну систему переміщення. Від швидкості яку формує система переміщення залежить потужність його видобування. За рух комбайну відповідають два електродвигуни АИР250М6, кожний з яких має потужність 55 кВт. Так як навантаження на двигуни розподіляється рівномірно можливо зробити висновок, що модель об'єкта управління може включати динаміку тільки одного двигуна, що було підтверджено відповідними дослідженнями [6]. Таким чином модель об'єкта управління може бути побудована на підставі добре вивченої моделі системи управління асинхронного електропривода (рис. 2.1).

До складу моделі входить пропорційний регулятор, частотний перетворювач, модель електроприводу виходом якої є момент на його валу та модель електроприводу виходом якої є швидкість обертання вала. На вхід моделі задається уставка швидкості, на виході моделі отримується значення швидкості електроприводу.

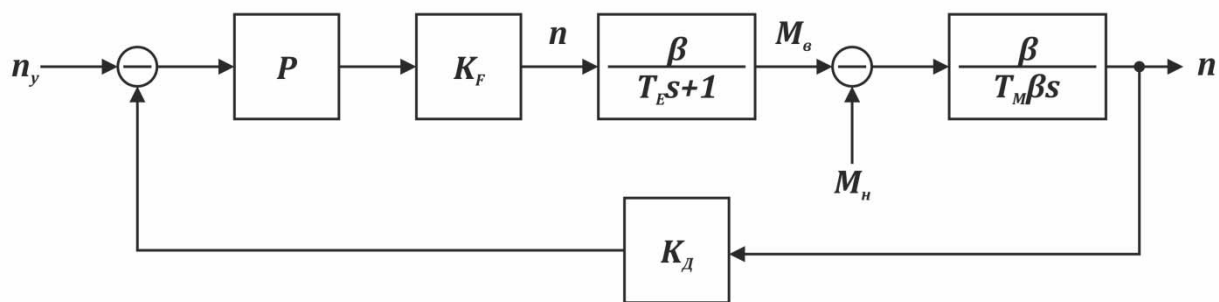


Рисунок 2.1 – Структура системи управління асинхронним електроприводом

Проаналізувавши структуру моделі системи асинхронного електропривода [7] на її підставі була розроблена структура моделі переміщення комбайну, яка наведена на рис. 2.2.

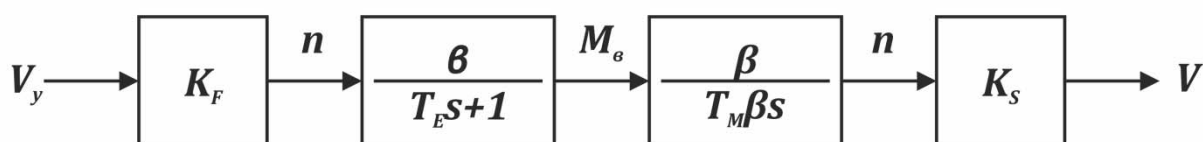


Рисунок 2.2 – Структура моделі переміщення комбайну

До складу моделі входить частотний перетворювач, модель електроприводу виходом якої є момент на його валу, модель електроприводу виходом якої є швидкість обертання валу, та модель перетворення обертання валу в переміщення комбайну. На вхід моделі задається уставка швидкості, на виході моделі отримується швидкість переміщення комбайну.

За для побудови моделі об'єкта управління в математичному пакеті необхідно виконати розрахунок її параметрів. Розрахунок параметрів виконується для моделей електроприводу виконується на підставі параметрів електродвигуна. В якості електродвигуна на комбайні використовується двигун з параметрами відповідними до двигуна АИР250М6, технічні характеристики котрого наведені в табл. 2.1.

Таблиця 2.1 – Технічні характеристики двигуна подачі комбайна

Потужність, кВт	Частота обертання, об/хв	Напруга, В	Сила струму, А	КПД, %	cosφ	Мп/Мн	Іп/Ін	J, кг*м ²
55	1000 (980)	380	104	92.8	0.86	2.1	7.0	1.48

Розглянемо розрахунок параметрів моделі об'єкта управління на підставі параметрів електродвигуна. Модуль жорсткості:

$$\beta = \frac{2 \cdot M_K}{n \cdot s_K}, \quad (2.1)$$

де β – модуль жорсткості, M_K – критичний момент на валу електроприводу (Н · м), n – швидкість обертання ротора двигуна (об/с), s_K – критичне ковзання електродвигуна.

Критичний момент на валу двигуна:

$$M_K = \lambda \cdot M_H, \quad (2.2)$$

де λ – перевантажувальна властивість електродвигуна, M_H – номінальний момент на валу електроприводу (Н · м).

Номінальний момент електродвигуна:

$$M_H = 9.55 \cdot \frac{P_H}{n}, \quad (2.3)$$

де P_H – номінальна потужність електродвигуна.

Швидкість обертання ротора електродвигуна:

$$n = \frac{n_p}{60}, \quad (2.4)$$

де n_p – реальна швидкість обертання електродвигуна (об/хв).

Критичне ковзання електродвигуна:

$$s_K = s_H \cdot (\lambda + \sqrt{\lambda^2 - 1}), \quad (2.5)$$

де s_H – номінальне ковзання електродвигуна.

Номінальне ковзання електродвигуна:

$$s_H = \frac{n_H - n_p}{n_H}, \quad (2.6)$$

де n_H – номінальна швидкість електродвигуна (об/хв).

Електромагнітна постійна часу двигуна:

$$T_E = \frac{1}{\omega \cdot s_K}, \quad (2.7)$$

де T_E – електромагнітна постійна часу двигуна (с), ω – кутова частота напруги живлення (рад/с).

Кутова частота напруги живлення:

$$\omega = 2\pi f, \quad (2.8)$$

де f – частота обертів електромагнітного поля двигуна (Гц).

Електромеханічна постійна часу електродвигуна:

$$T_M = \frac{\omega \cdot s_K \cdot (J + J_K)}{2 \cdot p \cdot M_K}, \quad (2.9)$$

де T_M – електромеханічна постійна часу двигуна (с), J – момент інерції електродвигуна ($\text{кг} \cdot \text{м}^2$), J_K – механічний момент інерції комбайну, приведений до вала двигуна ($\text{кг} \cdot \text{м}^2$), p – кількість пар полюсів електродвигуна.

Коефіцієнт перетворення лінійної швидкості комбайна до обертів електроприводу:

$$K_R = \frac{n_p}{V_M \cdot 60}, \quad (2.10)$$

де K_R – коефіцієнт перетворення лінійної швидкості комбайна до обертів електроприводу, V_M – максимальна швидкість комбайна (м/мин).

Коефіцієнт перетворення керуючого впливу який подається на частотний перетворювач:

$$K_F = \frac{n_p}{100 \cdot 60}, \quad (2.11)$$

де K_F – коефіцієнт перетворення керуючого впливу.

Коефіцієнт перетворення обертів електропривода до лінійної швидкості комбайна:

$$K_S = \frac{V_M}{n_p}. \quad (2.12)$$

На підставі наведених формул розраховані параметри моделі об'єкта управління які наведені в табл. 2.2.

Таблиця 2.2 – Параметри моделі об'єкта управління

β	M_K	M_H	n	s_K	s_H	T_E	ω	T_M	K_R	K_F	K_S
1747.85	536	1126	16.33	0.0789	0.02	0.04	314.159	0.0159	3.27	0.163	0.306

Модель об'єкта управління розроблена в графічній середі імітаційного моделювання Simulink наведена на рис. 2.3.

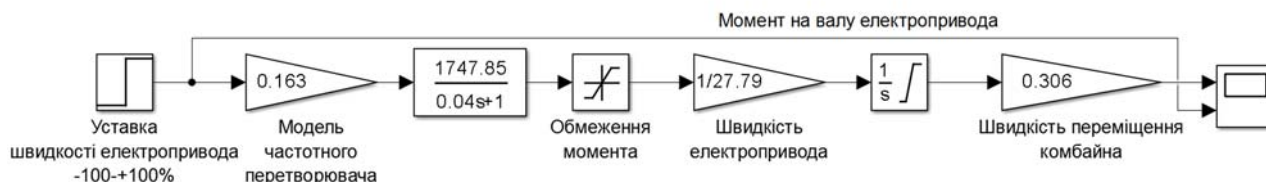


Рисунок 2.3 – Модель об'єкта управління

Результат роботи моделі наведено на рис. 2.4.



Рисунок 2.4 – Результат роботи моделі

Як бачимо по нижньому графіку на вхід моделі об'єкта управління була подана потужність 100 %. На верхньому графіку бачимо, що модель об'єкта почала відразу реагувати на значення уставки потужності, швидкість пішла в гору і досягла 5 м/хв.

Виходячи з результатів моделювання можливо зробити висновок, що аналітично отримана модель відповідає об'єкту управління та може бути використана при виконанні синтезу системи управління.

2.2.1 Критерії якості функціонування системи управління

Згідно з вимогою до технологічного процесу основними критеріями якості які висуваються до системи управління є відсутність автоколивань. Тобто перехідний процес в системі управління повинен бути монотонним.

Перерегулювання швидкості подачі комбайна повинно бути не більше 5%, що дозволить використовувати дану систему в складі системи управління концентрацією метану в повітрі.

Статична похибка системи по збуренню повинна бути рівна 0% за для оптимального відпрацювання діаграми швидкостей в оптимальному режимі при наявності збурень які виникають із-за різної міцності порід, та через зміну уставки швидкості.

2.3 Висновки по розділу

В межах теоретичного розділу було проаналізовано устрій комбайну та виявлено, що за його переміщення відповідають два двигуни. Кожен двигун несе рівномірне навантаження виходячи з цього управління двигунами в процесі переміщення є однаковим.

На підставі моделі системи управління асинхронним електродвигуном була розроблена модель об'єкта управління в якості якого виступає двигуни переміщення комбайну. Модель має вигляд передавальної функції реалізованої в графічному середовищі імітаційного моделювання Simulink. Параметри моделі були отримані внаслідок математичного розрахунку на підставі параметрів електродвигуна подачі комбайна.

Через особливості моделювання асинхронного електроприводу дана модель відображає лише динаміку перехідного процесу, її повноцінне використання можливе лише в замкнутій системі керування. Тому усі подальші дослідження будуть виконані в процесі синтезу системи управління швидкістю комбайна.

Згідно до технологічного процесу були обрані критерій якості роботи системи управління. Перехідний процес повинен бути монотонним, перерегулювання повинно бути менше 5%, статична похибка при наявності збурення повинна бути рівна 0.

3 СИНТЕЗ КОМП'ЮТЕРНОЇ СИСТЕМИ

3.1 Постановка завдання дослідження

Згідно з завданням в даному розділі потрібно виконати синтез системи управління швидкістю подачі комбайна на базі раніш отриманої математичної моделі об'єкта управління в якості якого виступає електропривід подачі комбайна.

Для синтезу системи управління повинно бути використано пропорційний, пропорційно-інтегруючий та пропорційно-інтегровально-диференціальний регулятори.

Оцінка якості функціонування системи управління повинна бути проведена з використанням раніш отриманих критеріїв якості.

Результатом синтезу системи управління повинна бути модель розроблена для графічного середовища імітаційного моделювання Simulink яке входить до складу математичного пакету MATLAB.

3.2 Синтез системи управління

Система управління швидкістю є класичною системою управління [8], яка складається з регулятора, об'єкта управління, негативного зворотного зв'язку за швидкістю та уставки швидкості (рис. 3.2).

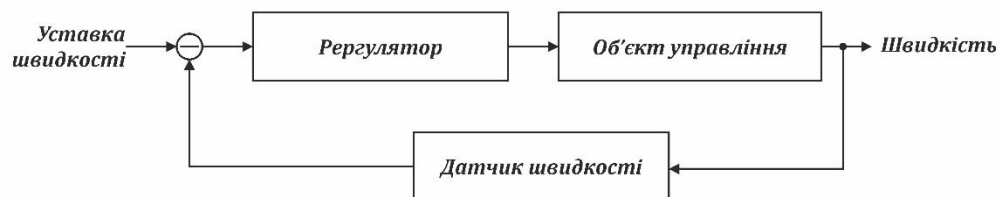


Рисунок 3.1 – Структура системи управління

Як бачимо з структури на вхід системи задається уставка швидкості. Значення помилки системи управління розраховується як різниця значення уставки та швидкості яка вимірюється за допомогою датчика швидкості. На

підставі значення помилки регулятор розраховує керуючий вплив який подається на об'єкт управління.

В якості об'єкта управління виступає модель яка була розроблена в теоретичному розділі. В свою чергу згідно з завданням в якості регулятора виступає структура отримана на базі ПД-регулятор [9, 10]. Таким чином необхідно розглянути функціонування системи управління при П, ПІ, ПІД та ПД-регуляторах [11-13]. Модель системи управління розроблена на базі П-регулятора наведена на рис. 3.2. П-регулятор реалізовано у вигляді передавальної функції знаменник якої дорівнює 1.

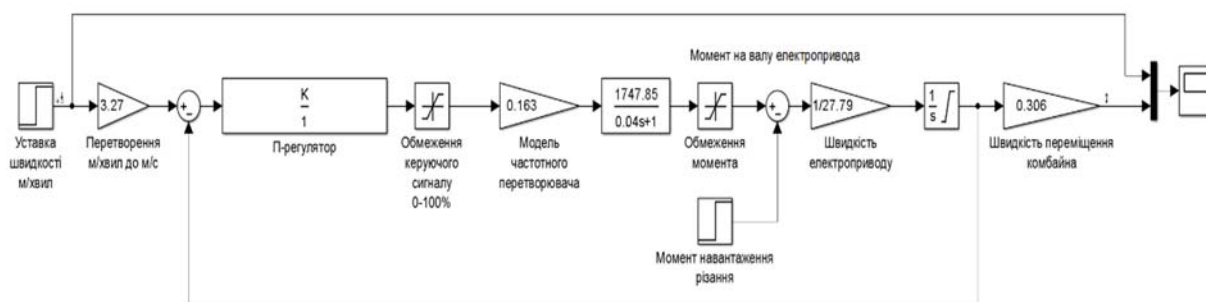


Рисунок 3.2 – Модель системи управління на базі П-регулятора

Налаштування регулятора виконаємо за допомогою методів: Зіглера-Нікольса, AMIGO та робастного методу (табл. 3.1). Результати моделювання наведені на рис. 3.3.

Таблиця 3.1 – Параметри П-регуляторів

№	Назва методу налаштування	К
1	Метод Зіглера-Нікольса	2,439
2	AMIGO	0,732
3	Робастний метод	0,732

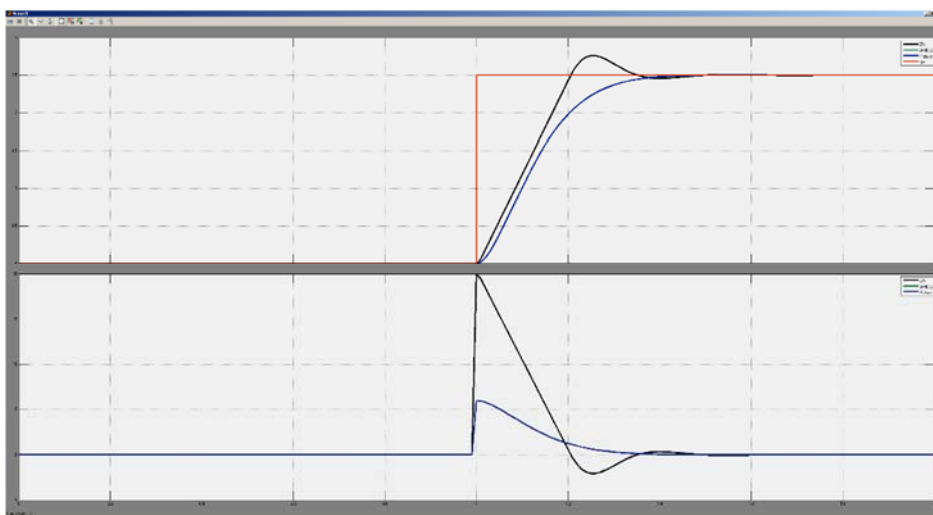


Рисунок 3.3 – Результати моделювання системи управління на базі П-регулятора

Так як параметри регулятора при використанні метода AMIGO та робастного метода збігаються на графіку відображаються дві характеристики. Оцінка якості налаштування системи управління наведена в табл. 3.2.

Таблиця 3.2 – Оцінка якості налаштування системи управління на базі П-регуляторів

№	Назва методу налаштування	Час наростання		Час встановлення		Перерегулювання		Інтегральна оцінка
		%	с	%	с	%		
1	Метод Зіглера-Нікольса	100,00	0,19	36,66	0,26	10,440	0,261	5,489
2	AMIGO	66,67	0,26	70,70	0,13	0,150	0,003	5,114
3	Робастний метод	66,67	0,26	70,70	0,13	0,150	0,003	5,114

Як бачимо метод Зіглера-Нікольса дає більш агресивний результат за рахунок більшого коефіцієнта підсилення. При цьому перерегулювання складає 10,44 %, що не відповідає вимогам. При використанні параметрів отриманих за рахунок робастного методу перерегулювання прагне до нуля, однак при цьому погіршується час наростання на 33 %. Розглянемо далі використання ПІ-

регулятора (табл. **Ошибка! Источник ссылки не найден.**), результати моделювання якого наведені на рис. 3.4.

Таблиця 3.3 – Параметри ПІ-регуляторів

№	Назва методу налаштування	К	T_i , с
1	Метод Зіглера-Нікольса	40,144	2,195
2	AMIGO	1,360	0,854
3	Робастний метод	0,444	1,406

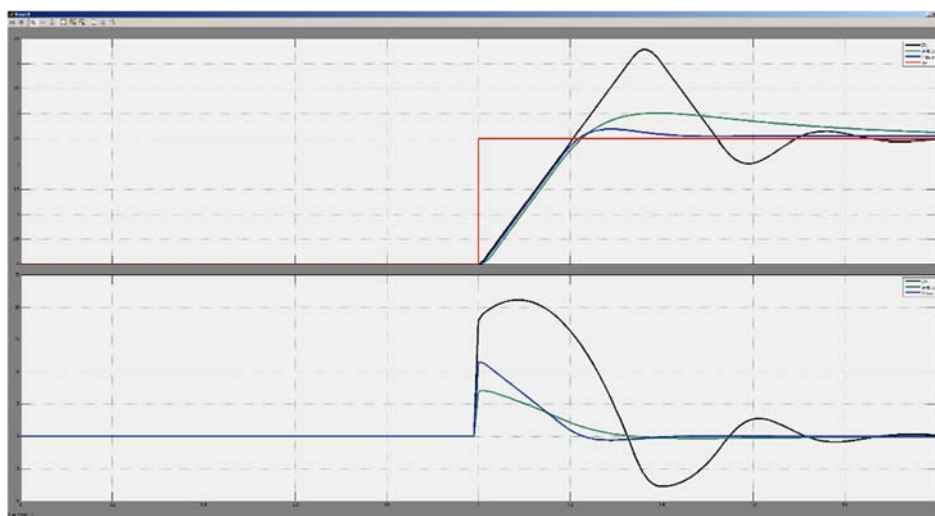


Рисунок 3.4 – Результати моделювання системи управління на базі ПІ-регулятора

Оцінка якості налаштування системи управління наведена в табл. 3.4.

Таблиця 3.4 – Оцінка якості налаштування системи управління на базі ПІ-регуляторів

№	Назва методу налаштування	Час наростання		Час встановлення		Перерегулювання		Інтегральна оцінка
		%	с	%	с	%		
1	Метод Зіглера-Нікольса	100,00	0,19	-	-	71,340	1,784	7,112
2	AMIGO	95,24	0,20	-	-	20,470	0,512	6,607
3	Робастний метод	100,00	0,19	-	-	7,694	0,192	5,652

Як бачимо використання ПІ-регулятора не можливо так як перерегулювання у всіх трьох методах є вищим за вимоги, а перехідні процеси мають коливальний характер або сильно затягнуті. Розглянемо далі використання ПД-регулятора (табл. 3.5), результати моделювання якого наведені на рис. 3.5.

Таблиця 3.5 – Параметри ПД-регуляторів

№	Назва методу налаштування	K	T_i , с	T_d , с	T_N , с
1	Метод Зіглера-Нікольса	2,926	0,080	0,020	0,002
2	AMIGO	0,044	0,320	0,020	0,002
3	Робастний метод	4,972	2,595	0,028	0,013

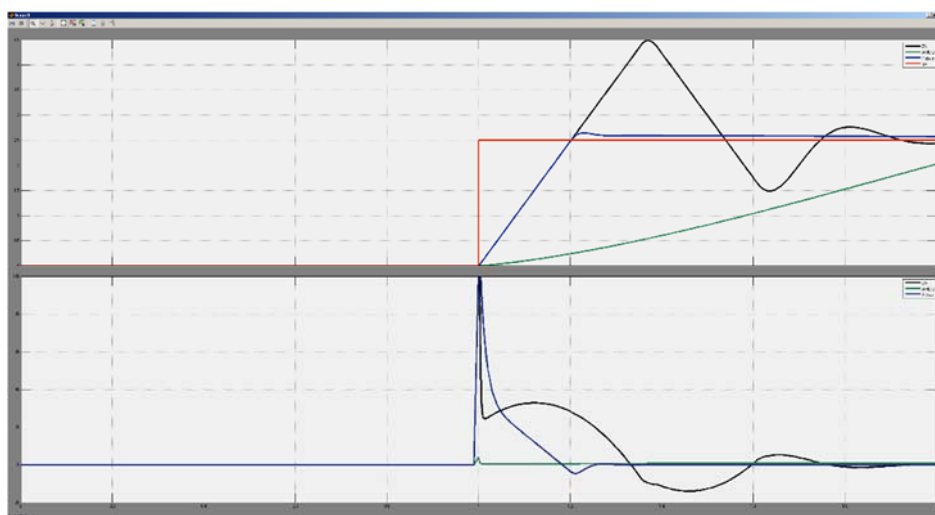


Рисунок 3.3 – Результати моделювання системи управління на базі ПД-регулятора

Оцінка якості налаштування системи управління наведена в таблиці **Ошибка! Источник ссылки не найден..** Як бачимо використання ПД-регулятора також не можливо так як метод AMIGO дав дуже мале значення коефіцієнта підсилення за рахунок чого перехідний процес дуже затягнутло. При методі Зіглера-Нікольса перехідна характеристика має коливальний характер. При робастному методі перерегулювання виходить за обмеження.

Розглянемо далі використання ПД-регулятора (табл. **Ошибка! Источник ссылки не найден.**), результати моделювання якого наведені на рисунку **Ошибка! Источник ссылки не найден.**

Таблиця 3.6 – Оцінка якості налаштування системи управління на базі ПД-регуляторів

№	Назва методу налаштування	Час наростання		Час встановлення		Перерегулювання		Інтегральна оцінка
		%	с	%	с	%		
1	Метод Зіглера-Нікольса	100	0,19	-	-	79,54	1,988	7,276
2	AMIGO	-	-	-	-	-	-	1,141
3	Робастний метод	100	0,19	-	-	5,788	0,014	5,761

Таблиця 3.7 – Параметри ПД-регуляторів

№	Назва методу налаштування	K	T _d , с	T _N , с
1	Робастний метод	5,276	0,024	0,012

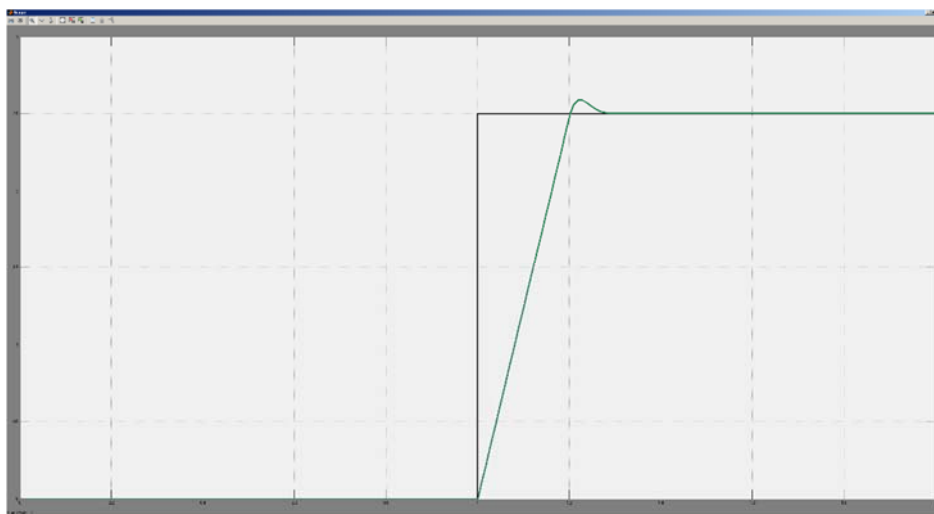


Рисунок 3.6 – Результати моделювання системи управління на базі ПД-регулятора

Оскільки із наведених методів тільки робастний дає розрахунок параметрів ПД регулятора аналіз функціонування системи управління буде виконуватися по одному комплекту параметрів (табл. 3.7). Як бачимо даний регулятор дав результати які за домовляють поставленим вимогам. Час наростання

мінімальний, час встановлення прагне до 100 %, а перерегулювання не перевищує 5 %.

Таким чином система управління відповідає заданим вимогам лише при використанні пропорційного та пропорційно-диференціюючого регуляторів параметри котрих розраховано за робастним методом.

Таблиця 3.7 – Оцінка якості налаштування системи управління на базі ПД-регуляторів

№	Назва методу налаштування	Час наростання		Час встановлення		Перерегулювання		Інтегральна оцінка
		%	с	%	с	%		
1	Робастний метод	100	0,19	83,79	0,08	3,597	0,08991	5,431

Крім виконання переводу швидкості комбайну система управління також повинна компенсувати збурення в усталеному режимі. Для перевірки реакції системи подаємо на неї збурення в час рівний 1 секунда. Результат моделювання наведено на рис. 3.7.

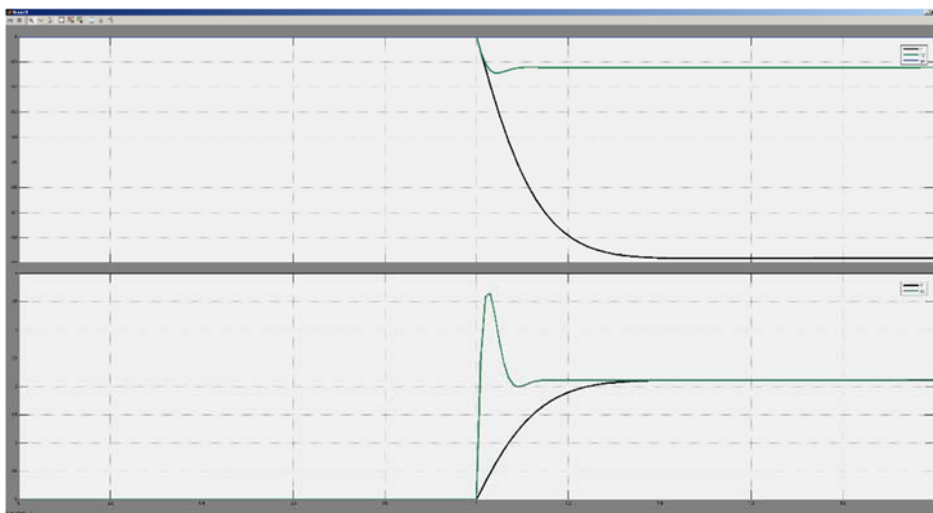


Рисунок 3.7 – Результати моделювання системи управління зі збуренням

Як бачимо з результатів моделювання при наявності збурення обидва регулятора мають усталену похибку. З метою компенсації усталеної похибки

використаємо ПІ-регулятор та ПІД регулятори параметри яких наведені в табл. 3.8. Результати моделювання наведені на рис. 3.8.

Таблиця 3.8 – Параметри регуляторів

№	Назва регулятора	К	T_i , с	T_d , с	T_N , с
1	ПІ-регулятор	1,406	0,200	-	-
2	ПІД-регулятор	4,972	0,100	0,0276	0,0125



Рисунок 3.8 – Результати моделювання системи управління зі збуренням

Як бачимо використання регуляторів з інтегруючою складовою дозволяє компенсувати статичну похибку. При чому використання ПІД-регулятора дозволяє зменшити час реакції системи управління на збурення та відхилення дійсного значення.

Таким чином можливо зробити висновок, що при зміні уставки швидкості комбайну у якості регулятора повинен використовуватися ПІД-регулятор, а при стабілізації швидкості його переміщення ПІД регулятор.

3.3 Розробка принципової схеми системи керування

3.3.1 Розробка вимог до принципової схеми системи керування

Виходячи з опису процесу контролю та технічних вимог, апаратна частина повинна підтримувати такі функції:

- отримання значень з датчиків по шині CAN згідно з протоколом CANOpen від контролера ВРС;
- стеження за еталонами режимів для керуючого технологічного обладнання;
- збір інформації о перебігу технологічного процесу за допомогою серверу;
- відображення зібраної інформації на автоматизованому робочому місці;
- зберігання операторної бази даних протоколу процесу.

Для визначення кількості і розподілу вхідних та вихідних сигналів необхідно провести аналіз та класифікацію входів автоматизованої системи контролю.

3.3.2 Аналіз входів автоматизованої системи контролю

В результаті аналізу отримана класифікація входів, яка наведена у табл. 3.10.

Для вибору елементної бази нижнього рівня потрібно провести угруповання вхідних сигналів з табл. 3.10 за належністю обладнання та електричних параметрів. Результати виконаного угруповання обладнання наведені у табл. 3.11.

Табл. 3.10 – Перелік вхідних сигналів автоматизованої системи контролю роботи очисного комбайну

№ п/п.	Найменування інформації	Ідентифікатор	Напр. вх./вих.	Функція	Вид	Джерело/Отримувач	Форма подання		Період вв./вив. сек.
							Зовнішня	Внутрішня	
1.	Споживана потужність приводом переміщення 1	1-1	Вхід	Контр.	CAN	Давач MBP	24 В	Згідно протоколу	0,1
2.	Споживана потужність органом 1	2-1	Вхід	Контр.	CAN	Давач MBP	24 В	Згідно протоколу	0,1
3.	Положення органа 1	3-1	Вхід	Контр.	Імпульсний	Давач EEх OG 9	24 В	32 біт	0,1
4.	Положення комбайну	4-1	Вхід	Контр.	CAN	Давач LIMES LI20	24 В	Згідно протоколу	0,1
5.	Тиск в гідравлічній системі комбайну	5-1	Вхід	Контр.	Струм	Давач ДМ5007АЕх	24 В	Згідно протоколу	0,1
6.	Положення органа 2	6-1	Вхід	Контр.	Імпульсний	Давач EEх OG 9	24 В	32 біт	0,1
7.	Споживана потужність органом 2	7-1	Вхід	Контр.	CAN	Давач MBP	24 В	Згідно протоколу	0,1
8.	Тиск в гідравлічній системі кріпії	8-1	Вхід	Контр.	Струмовий інтерфейс	Давач ДМ5007АЕх	4-20 мА	16 біт	0,1
9.	Концентрація метану	9-1	Вхід	Контр.	Інтерфейс напруги	Давач ТХМ-2,8-2	0-0,4 мВ	16 біт	0,1
10.	Швидкість повітря	10-1	Вхід	Контр.	Струмовий інтерфейс	Давач ТХ1322	4-20 мА	16 біт	0,1
11.	Споживана потужність приводом переміщення 2	11-1	Вхід	Контр.	CAN	Давач MBP	24 В	Згідно протоколу	0,1

Таблиця 3.11 – Зведена таблиця входів системи контролю роботи очисного комбайну

№ п/п.	Пристрій	Опис входу або виходу	Кількість входів або виходів
		Аналогові входи	
1.	Контроль тиску (2 шт.)	ДМ5007АЕх, сигнал 4-20 мА	2
2.	Контроль швидкості повітря	ТХ1322, сигнал 4-20 мА	1
3.	Контроль метану	ТХМ-2,8-2, 0-0,4 мВ	1
РАЗОМ			4
		Імпульсні входи	
4.	Контроль положення органу (2 шт.)	Давач ЕЕх ОG 9, 24 В	2
РАЗОМ			2
		Інтелектуальні пристрої	
5.	Контроль споживаної потужності (4 шт.)	МВР, шина CAN	4
6.	Контроль положення комбайну (1 шт.)	LIMES LI20, шина CAN	1
РАЗОМ			5

Результати аналізу входів використанні для вибору елементної бази систему контролю. Перелік вхідних сигналів буде використаний при розробці алгоритму роботи системи.

3.3.3 Вибір елементної бази системи управління

З початку оборемо апаратне забезпечення контролю для системи керування очисним комбайном яке реалізується контролером BARTER VIPC.

В якості давача вимірювання тиску доцільного обрати давач з токовим виходом 4-20 мА з діапазоном вимірювання від 0 до 16 Атм. У якості давача обрано п'єзорезистивний компактний перетворювач тиску підвищеної міцності ДМ5007АЕх з наступними характеристиками:

- діапазон вимірювання від 0 до 16 Атм;
- токовий вихід 4-20 мА;
- ступінь захисту IP65;
- вибухозахист 0ЕхіаІІСТ5Х.

В якості вимірювача швидкості повітря доцільно обрати давач з токовим виходом 4-20 мА з діапазоном вимірювання від 0,5 до 5 м/сек. Обрано давач швидкості потоку повітря TX1322 з наступними характеристиками:

- діапазон вимірювання від 0,5 до 20 м/сек;
- токовий вихід 4-20 мА;
- ступінь захисту IP65
- вибухозахист PO Ia.

В якості вимірювача рівня метану обрано термокatalітичний давач шахтного метану ТХМ-2,8-2 з наступними характеристиками:

- діапазон вимірювання від 0 до 2,5 %;
- діапазон вихідної напруги від 0 до 0,4 мВ;
- межі основної абсолютної похибки $\pm 0,2$ %.

В якості вимірювача положення органу комбайну доцільно обрати давач ударостійкий з діапазоном 600 імпульсів на оборот. У якості давача обрано вибухозахищений енкодер EEx OG 9 з наступними характеристиками:

- число імпульсів на оборот від 1 до 5000;
- ступінь захисту IP56;
- вибухозахист II 2G Ex de IIC T6.

В якості давача визначення положення комбайну доцільно обрати давач LIMES LI20 який використовує магнітну систему для визначення положення та має CAN інтерфейс.

В якості давача вимірювання споживання потужності доцільно обрати шахтну систему вимірювання потужності яка підключається до польової шини CAN з використанням іскробезпечного рішення. Основні вимоги до вибору давача:

- номінальна напруга ~ 1140 В;
- номінальний струм 250 А;
- максимальний струм на протязі 1 секунди 3000 А;
- можливість підключення до польової шини CAN;
- вибухозахист.

У даний час на Україні Торезький електротехнічний завод виробляє давачі у вибухозахищеному виконанні. Для даної роботи обрані 4 давачі МВР з наступними характеристиками:

- номінальна напруга ~ 1140 В;
- номінальний струм 320 А;
- максимальний струм на протязі 1 секунди 8000 А;
- інтерфейс зв'язку CAN;
- вид та рівень захисту РВ ЗВ Іа (Exdial).

Усі давачі підключаються до польової шини CAN, що вимагає від контролеру окремого порту CAN.

У якості контролеру обрано контролер VIPA 314-6CF02 [9-11]:

- напруга живлення 24 В;
- споживана потужність 6 Вт;
- 8 дискретних входів;
- 8 входів/виходів;
- 4 аналогові входи;
- 2 аналогові виходи;
- вбудований Ethernet інтерфейс.

Підключення давачів положення виконується через дискретні рахункові входи.

Для підключення до контролеру давачів з інтерфейсом CAN з протоколом CANOpen обрано модуль комунікаційного процесора VIPA 342-1CA70 з споживаною потужністю 2,75 Вт.

Для забезпечення живлення програмованого логічного контролеру та давачів обрано два блока живлення VIPA 307-1BA00 з максимальним током навантаження 2,5 А.

Оборемо апаратне забезпечення системи контролю роботи очисного комбайну [11]. Збір даних з контролера ВІРС буде виконуватися програмованим логічним контролером VIPA 314-2AG12 з блоком живлення VIPA 307-1BA00 за допомогою модулю комунікаційного процесора VIPA 342-1CA70. Передача

даних від контролера на поверхню буде реалізована на базі RS-485 трансформованого до оптичного сигналу. Трансформація сигналу буде виконуватися завдяки перетворювачам TCF-142-S-ST на стороні ПЛК та ICF-1150-M-ST на стороні серверу, який підключений до блока живлення Carlo Gavazzi SPD 2405 . На сервері буде встановлена картка RS-485 інтерфейсу CP-132.

Таким чином усі потреби системи контролю роботи очисного комбайну задоволені, а на підставі обраної елементної бази може бути побудована принципова схема системи контролю.

3.3.4 Реалізація принципової схеми системи контролю

На підставі обраної елементної бази реалізована система контролю роботи очисного комбайну в частині передачі даних від штрекового контролера ВІРС до серверу (рис. 3.9).

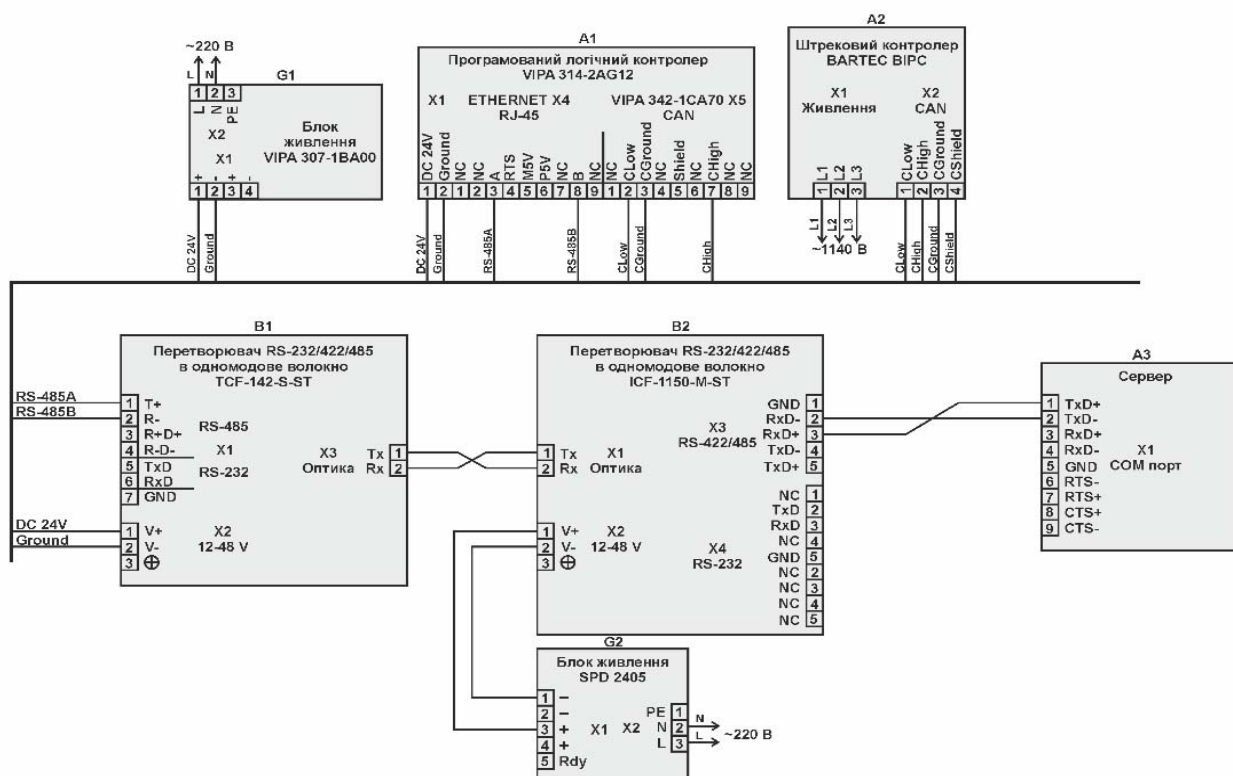


Рисунок 3.9 – Принципова схема системи контролю

Згідно з схемою штрековий контролер (A2) живиться з трифазної мережі. Програмований логічний контролер VIPA 314-2AH12 (A1) живиться від блока живлення VIPA 307-1BA00 (G1), також від цього блока живлення живиться перетворювач до одномодового оптичного волокна TCF-142-S-ST (B1) на стороні ПЛК. Перетворювач ICF-1150-M-ST (B2) на стороні персонального комп'ютера (A3) живиться від блока живлення SPD 2405 (G2).

Програмований логічний контролер (A1) отримує дані від штрекового контролеру (A2) по CAN шині згідно з протоколом CANOpen. Далі він очікує на запит від персонального комп'ютера (A3) на відправку даних. По приходу запита програмований логічний контролер формує Modbus посилку та відправляє її по RS-485 інтерфейсі. Данні з контролеру надходять до перетворювача TCF-142-S-ST (B1) який перетворює їх з електричних сигналів до оптичних. Оптичний сигнал надходить до перетворювача ICF-1150-M-ST (B2) який перетворює його до електричного сигналу RS-485. Після чого пакет Modbus отримується персональним комп'ютером (A3). Посилання запиту даних від персонального комп'ютера (A3) до програмованого логічного контролера (A1) виконується зворотнім чином.

Отримана принципова схема може бути використана для розробки алгоритму роботи системи контролю.

3.4 Робастність системи управління

З часом використання об'єкту управління його параметри змінюються. Тому потрібно виконати перевірку як система управління буде реагувати на зміну параметрів об'єкта управління.

Виконаємо перевірку отриманих систем управління на робастність. Для цього задаємо зміну коефіцієнта підсилення об'єкта управління на -10 %, далі задаємо зміну електромагнітної постійної часу на +10 %, після чого електромеханічної постійної часу на +10 %. Результати моделювання для ПД-регулятора наведені на рис. 3.10.

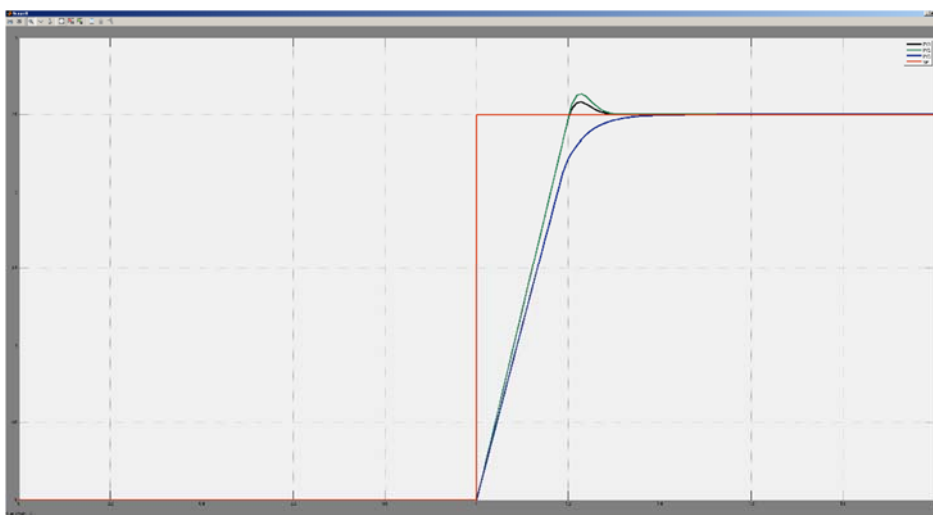


Рисунок 3.10 – Результати моделювання системи управління на базі ПД-регулятора

Оцінка якості налаштування системи управління наведена в табл. 3.13. Як бачимо при збільшенні електромагнітної постійної часу перерегулювання виходить за межі допусків. Для рішення цієї проблеми виконаємо корекцію часу диференціювання до 0,026.

Таблиця 3.13 – Оцінка якості налаштування системи управління на базі ПД-регуляторів

№	Зміна параметра	Час наростання		Час встановлення		Перерегулювання %	Перерегулювання	Інтегральна оцінка
		%	с	%	с			
1	Немає	100,00	0,19	83,79	0,08	3,597	0,090	5,431
2	$K - 10\%$	100,00	0,19	83,79	0,08	3,265	0,082	5,430
3	$T_E + 10\%$	100,00	0,19	81,17	0,09	5,394	0,135	5,445
4	$T_M + 10\%$	90,48	0,21	75,94	0,11	0,110	0,002	5,274

Оцінка якості налаштування системи управління після корекції постійної часу диференціювання наведена в табл. 3.14.

Таблиця 3.14 – Оцінка якості налаштування системи управління на базі ПД-регуляторів

№	Зміна параметра	Час наростання		Час встановлення		Перерегулювання %	Перерегулювання	Інтегральна оцінка
		%	с	%	с			
1	Немає	100,00	0,19	86,41	0,07	2,741	0,069	5,424
2	$K - 10\%$	100,00	0,19	86,41	0,07	2,346	0,059	5,423
3	$T_E + 10\%$	100,00	0,19	81,17	0,09	4,428	0,111	5,438
4	$T_M + 10\%$	90,48	0,21	73,32	0,12	0,062	0,002	5,268

Перевіримо на робастність систему управління на базі ПД-регулятора. Результати моделювання наведені на рис. 3.11.

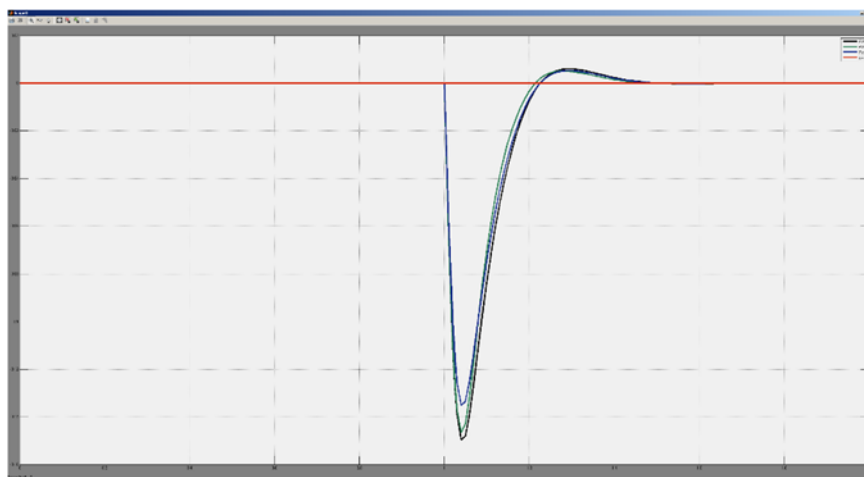


Рисунок 3.11 – Результати моделювання системи управління на базі ПД-регулятора

Як бачимо при кожному регуляторі система управління виконує поставленні перед нею функції, при чому параметри дійсного значення зостаються у межах допустимих значень.

На підставі обраних регуляторів розроблена система управління моделі якої наведена на рис. 3.12.

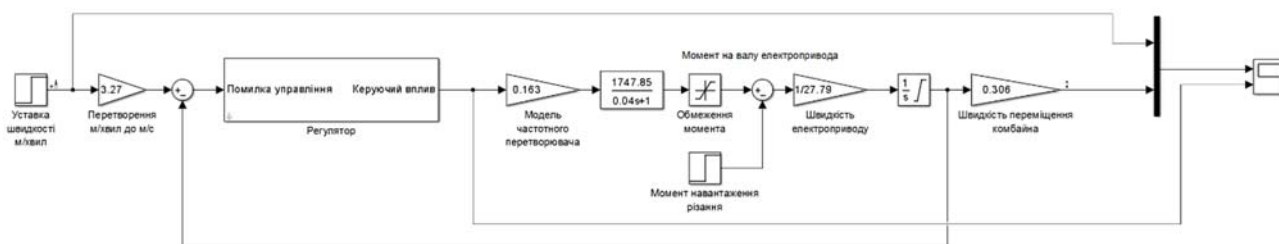


Рисунок 3.12 – Модель системи управління

У моделі системи управління використано комбінований регулятор який розроблений на базі ПД та ПІД регулятора (рис. 3.13).

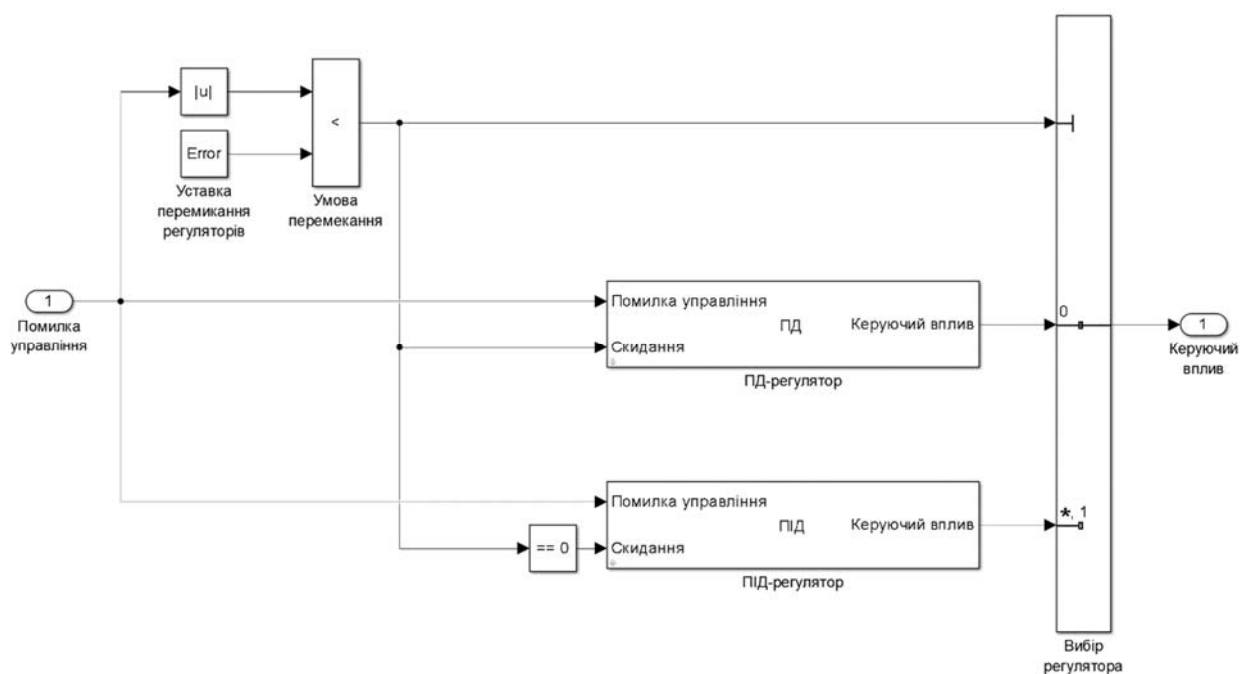


Рисунок 3.13 – Модель регулятора

На вхід кожного з регуляторів заходить значення похибки. На підставі похибки системи управління за допомогою аналогового мультиплексора та блока умови перемикання виконується обирання який з регуляторів у даний час формує керуючий вплив. Якщо похибка більша за задане значення використовується ПД-регулятор, якщо похибка менша за задане значення використовується ПІД регулятор. Збільшення значення перемикання призводить до зменшення агресивності системи управління. За рахунок входу скидання ПІД та ПД регуляторів, доки в якості регулятора використовується інший регулятор

інтегруюча та диференційна складові другого регулятора призначаються таким чином щоб на виході відповідних ланок було значення яке дорівнює нулю.

В системі управління в якості ПД-регулятора використана модель наведена на рис. 3.14. У моделі в якості диференційної складової використано реальну диференційну ланку (рис. 3.15), яка представлена у вигляді простора станів, для надання можливості її скидання.

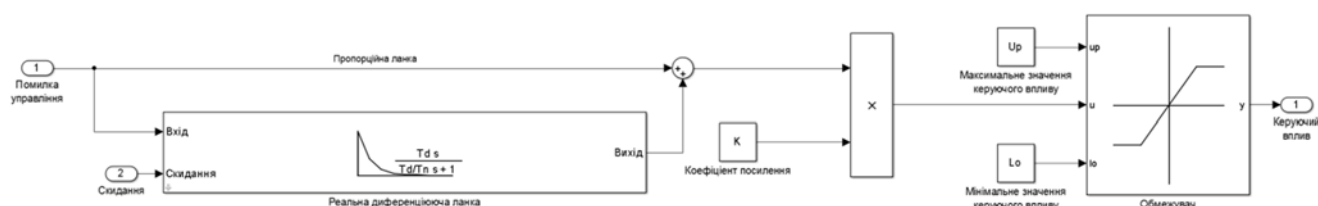


Рисунок 3.14 – Модель ПД-регулятора

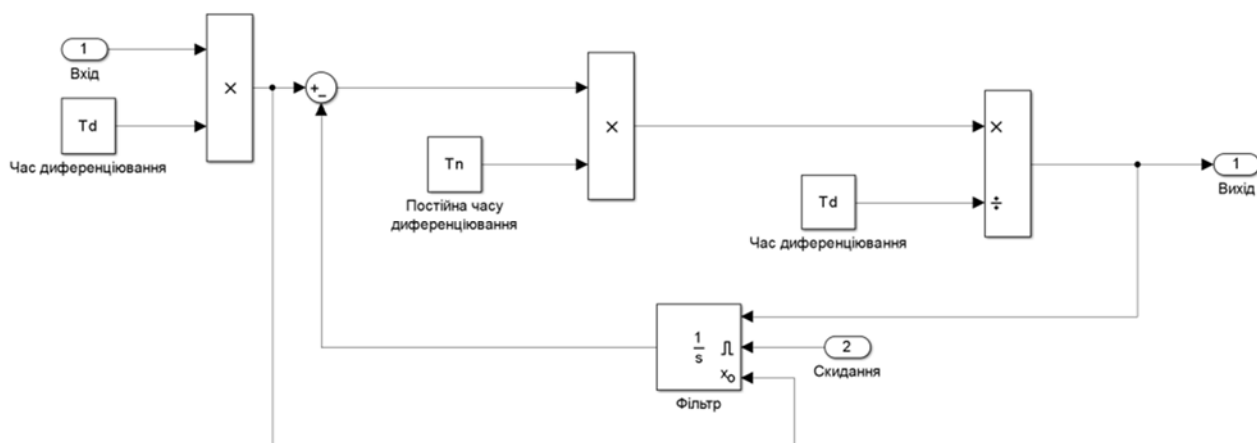


Рисунок 3.15 – Модель ПД-регулятора

В якості ПД-регулятора використана його стандартна форма (рис. 3.16) з реалізацією захисту від переповнення та реальною диференційною ланкою. Модель диференційної ланки відповідає моделі використаній в ПД-регуляторі, модель інтегруючої ланки наведена на рис. 3.17.

Модель засувки наведена на рис. 3.18.

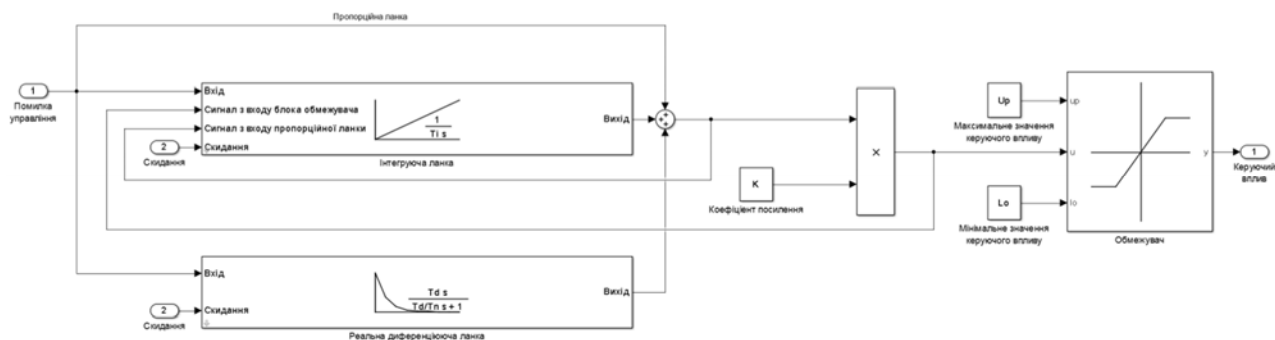


Рисунок 3.16 – Модель ПІД-регулятора

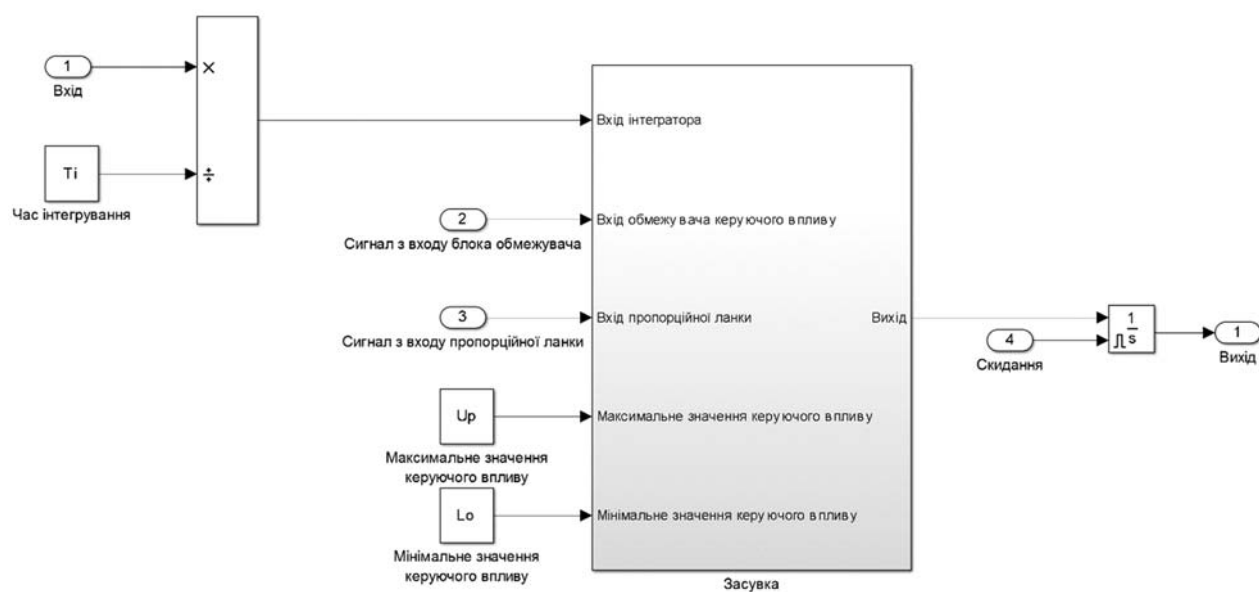


Рисунок 3.17 – Модель інтегруючої ланки

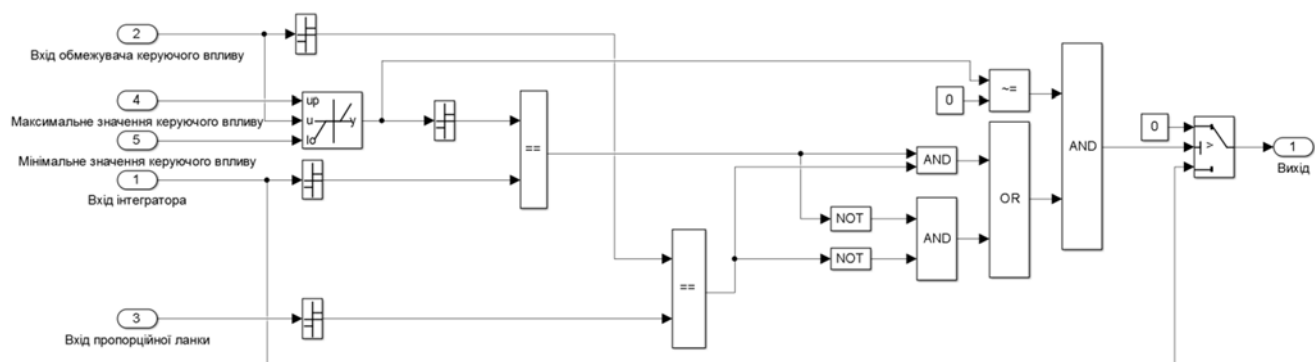


Рисунок 3.18 – Модель засувки

Коефіцієнт підсилення в моделях регуляторів встановлено на їх виходах. Таке рішення дозволяє легко корегувати усі параметри регуляторів. За

коефіцієнтом підсилення розміщено блок обмеження керуючого впливу. Параметри інтегруючої складової задаються як час інтегрування. Другий вхід кожного регулятора використано для реалізації скидання інтегруючої складової до нуля, та диференційної складової таким чином, щоб на її виході було значення порівняне нулю.

Остаточні параметри налаштування регуляторів наведені в табл. 3.14.

Таблиця 3.14 – Параметри регуляторів

№	Назва регулятора	K	T _i , с	T _d , с	T _N , с
1	ПД-регулятор	5,276	-	0,026	0,012
2	ПД-регулятор	30,000	0,100	0,0276	0,0125

Результат роботи моделі системи управління наведено на рис. 3.19, а результати оцінки якості функціонування системи управління в табл. 3.15.

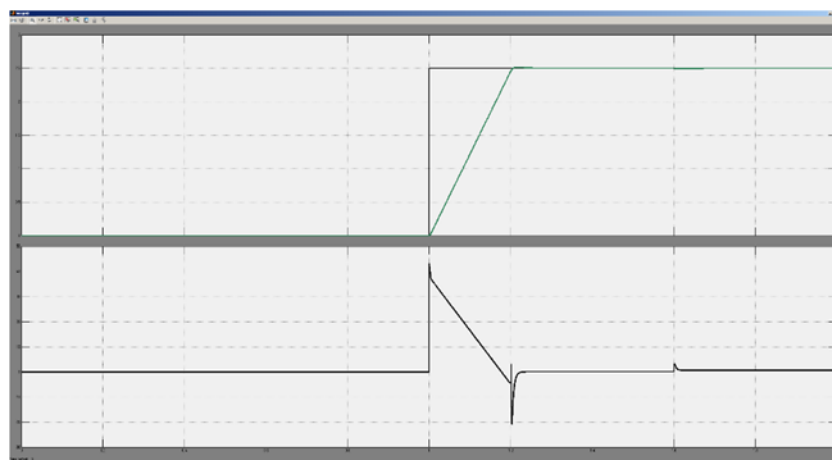


Рисунок 3.19 – Результат моделювання

Таблиця 3.15 – Оцінка якості налаштування системи управління

Час наростання		Час встановлення		Перерегулювання	Перерегулювання	Інтегральна оцінка
%	с	%	с	%		
100	0,19	99,50	0,02	0,4827	0,01207	5,401

Приклад функціонування системи управління при різноманітних значеннях уставок наведено на рис. 3.20.

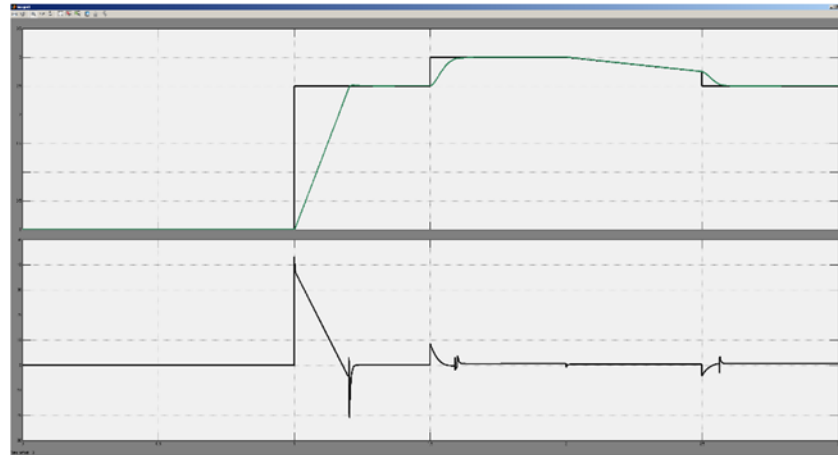


Рисунок 3.20 – Результат моделювання

Приклад функціонування системи управління при різноманітних значеннях збурення наведено на рис. 3.21.

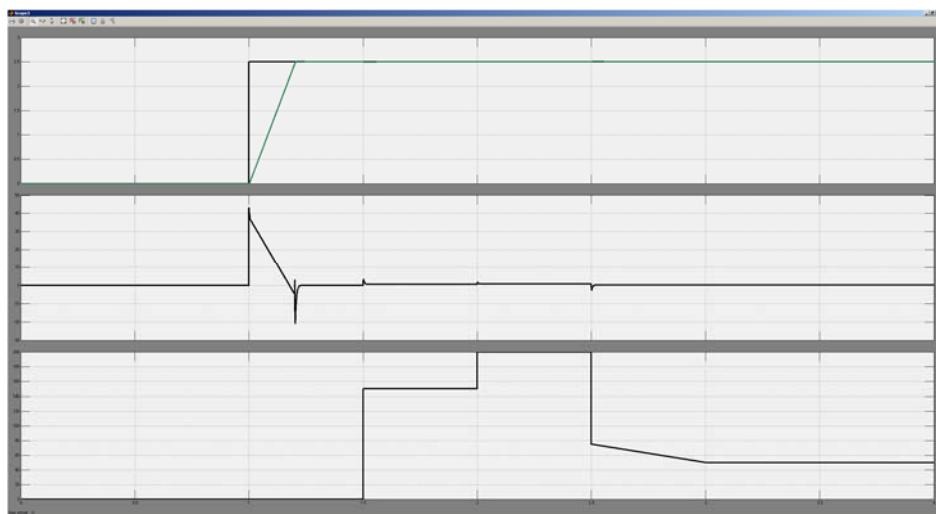


Рисунок 3.21 – Результат моделювання

3.5 Розробка програмного забезпечення

3.5.1 Призначення і область застосування програми

Програмне забезпечення призначене для автоматизованого управління видобувного комбайну вугільної шахти.

Розроблюване програмне забезпечення є нижнім рівнем програмного забезпечення АСУ. Виходячи з вимог програма повинна забезпечувати

управління об'єктом, отже до її складу повинна входити підсистема управління технологічним обладнанням. Дана підсистема складається з пристроїв збору інформації (датчики виміру струму і напруги, переміщення, положення виконавчих пристроїв), еталонів стану обладнання (завданній режим роботи комбайну), системи автоматичного контролю стану обладнання, програми управління, яка повинна реалізувати управління виконавчих пристроїв (приводи різання та подачі).

Крім того програма повинна забезпечувати візуалізацію та контроль, за технологічним процесом, тому до неї повинна входити підсистема інформаційного забезпечення роботи оператора. Дана підсистема складається з реєстрації параметрів процесу, людино-машинного інтерфейсу та сигналізації досягнення параметрами заданих та аварійних значень.

Також системою повинно забезпечуватися архівування технологічних параметрів, режимів роботи, процесів, які відбуваються, при цьому в неї повинна бути підсистема ведення архіву параметрів та подій, включно з базою даних та резервним сховищем.

Таким чином згідно з вимогами підсистема автоматизованого управління технологічним обладнанням представляє собою апаратно програмний комплекс, до якого входять кнопки, датчики, прилади узгодження, індикацій, виконавські пристрої та інше.

Підсистема інформаційного забезпечення роботи оператора представляє собою апаратно програмний комплекс на базі персонального комп'ютера з SCADA системою zenon Supervisor.

3.5.2 Обґрунтування технічних характеристик програми

3.5.2.1 Постановка завдання на розробку програми

Програмне забезпечення системи автоматизованого управління реалізує функції управління турбоповітродувкою.

Програмне забезпечення людино-машинного інтерфейсу має вікна: інформації про об'єкт, технологічний процес, хронологічний список подій та

інформаційний список тривоги. Виходячи з того, що людино-машинний інтерфейс повинен відображати декілька зображень, він повинен складатися з двох шаблонів на одному з котрих повинні відображатися кнопки переключення зображення, а на іншому самі зображення.

На основному зображенні будуть відображатися інформації про об'єкт управління, технологічний процес, зміну тиску в колошнику, хронологічний список подій та інформаційний список тривоги.

На наступному етапі опрацювання, на основі наявних даних, проводиться розбиття сформульованої завдання розробки програмного забезпечення на окремі підзадачі, блоки, процедури і функції. Після цього складається алгоритм роботи програми і відбувається побудова детальної схеми алгоритму програми, за якою згодом буде написана керуюча програма.

3.5.2.2 Опис алгоритму функціонування програми

Виходячи з технічних вимог і прийнятого протоколу обміну, програма нижнього рівня АСУ повинна виконувати наступні функції:

- взаємодія через гальванічно розв'язаної лінії зв'язку за узгодженим протоколом між контролером і пультом оператора;
- отримання сигналів від датчиків технологічного обладнання;
- видачу керуючих сигналів на виконавчі механізми технологічного обладнання;
- стеження за аварійними ситуаціями в роботі технологічних машин, виконання аварійної зупинки приводів;
- зберігання на верхньому рівні і видачу на нижній еталонів спрацьовування обладнання для регулювання режиму роботи обладнання.

Реалізація програми можлива тільки з урахуванням таких особливостей застосовуваного промислового контролера. ПЛК VIPA System 200V підтримує мову програмування FBD (функціональні блоки), таким чином, програма управління АСУ розділена на функціональні складові. Головним блоком

програми є основною алгоритм опитування, який циклічно виконується контролером. Його головним завданням є виклик необхідних в даний момент функціональних блоків програми. Так в функціональні блоки виділені алгоритми:

- алгоритм управління обладнанням;
- лгоритм відстеження та відпрацювання аварійних зупинок;
- алгоритм регулювання подачі різання;
- алгоритм регулювання подачі переміщення.

Поряд з основними алгоритмами необхідно провести початкову установку значень змінних при включенні контролера. Це можна виконати в блоці, за замовчуванням активному при запуску VIPA 200V - у другому, після чого активізувати інший блок виконання.

3.5.3 Опис і обґрунтування вибору методу організації вхідних та вихідних даних

Програма повинна використовувати такі вхідні дані:

- сигнали спрацьовування датчиків;
- в ручному режимі запуск процесу кнопкою пуск з панелі оператора.

Програма повинна видати наступні вихідні дані:

- інформацію про стан технологічного обладнання у вигляді одинично-позиційного коду станів датчиків, графічного відображення.

Програма зв'язку з верхнім рівнем повинна бути реалізована незалежно від основного програмного забезпечення і поставлятися в комплекті контролера.

3.5.3.1 Опис і обґрунтування вибору та складу технічних та програмних засобів

Технічні і програмні засоби для розробки програми обумовлені довідковим керівництвом по застосуванню контролерів VIPA200. Мова програмування - FBD (функціональні блоки), середовище розробки - Zenon (інструментальне середовище - безкоштовна, підтримує всі мови стандарту MEK 61131-3).

Це програмне забезпечення передбачається використовувати в складі ПО АСУТП. Технічні засоби - комп'ютер, сумісний з IBM PC (XT, AT, PC2), підключення здійснюється по порту RS-232.

3.5.4 Опис розробленої програми

3.5.4.1 Загальні відомості

Програмне забезпечення розробляється для прошивки в контролер нижнього рівня. Програмне забезпечення для АСУ розроблено на Techno FBD і являє собою схему, що складається з набору функціональних блоків, пов'язаних між собою через входи і виходи.

Програмне забезпечення розміщується у внутрішньому ПЗУ контролера.

3.5.4.2 Функціональне призначення

Виходячи з технічних вимог і прийнятого протоколу обміну, програма нижнього рівня АСУ ПУШ повинна виконувати наступні функції:

- запуск обладнання згідно технологічного процесу;
- реалізацію опитування датчиків, контролюючих технологічні характеристики об'єкта автоматизації;
- пуск і останов двигунів;
- регулювання продуктивністю двигунів;
- реалізацію відпрацювання процедури аварійної зупинки;
- контроль пересування;

3.5.4.3 Опис логічної структури програми

Програма користувача розміщена в сегменті оперативної пам'яті програм користувача контролера Vira200S і складається з головного організаційного блоку Main, функції DoState, функції Object і функції Controller.

Блок Main викликає Controller і відстежує ознака настання чергового події в системі (сигнали стоп / пуск, необхідність регулювання, зміна значень технологічних величин).

Блок Controller відпрацьовує технологічне карту процесу видобутку.

Функція `bo_ManipulatedValue` викликається Controller для перекладу даних вимірювання навантаження в дискретні ознаки відповідності технологічним еталонним значень параметрів. Функція `_PrevResetState_m` викликається Controller для перекладу даних вимірювання руху в дискретні ознаки відповідності технологічним еталонним значень параметрів.

Алгоритм роботи програмного забезпечення системи автоматизованого управління виглядає наступним чином.

При натисканні на пульті оператора кнопки «Пуск», або під час вступу команди запуску відбувається запуск роботи устаткування.

Заданий рівень тиску забезпечується регулюванням продуктивності комбайну, в залежності від завданого технологічного режиму.

3.5.4.4 Використані технічні засоби

Для виконання програми необхідний модуль контролера VIPA200V - 1 шт., Кабель програмування - 1 шт., Модуль перетворювача інтерфейсу SM702 - 1 шт.; модуль дискретного вводу 221-1BF00 - 2 шт. ; модуль дискретного виводу 222-1NF00 - 1 шт. ; модуль аналогового введення 231-1BD60 - 2 шт. ; система програмування STEP7 v.3.0.6. і старше.

3.3.4.5 Виклик і завантаження програми

Програма спочатку завантажується в енергонезалежну пам'ять VIPA200V з персонального комп'ютера за допомогою системи STEP7 за допомогою кабелю програмування, де і знаходиться весь термін експлуатації системи. Виклик програми на виконання відбувається по включенню живлення контроллера.

Виклик підпрограми відбувається при включенні живлення контроллера після проходження підпрограмою початкової ініціалізації.

3.5.4.6 Вхідні і вихідні дані

Вхідні змінні.

Disturbance	- збурення;
bi_Input	- керуючий вплив;
bi_Disturbance	- збуренняж
bo_Output	- дійсне значення.

Вихідні змінні:

l_State_0	- стан апериодичної ланки;
l_State_1	- стан інтегруючої ланки;
l_Temp	- допоміжна змінна;

3.5.5 Очікувані техніко-економічні показники

Розроблений варіант технічного рішення дозволить скоротити кількість обслуговуючого персоналу за рахунок автоматизації та дистанційного контролю за роботою технологічного обладнання.

3.6 Висновки по розділу

У результаті дослідження встановлено, що система управління на базі ПД-регулятора дає найкращі результати в режимах зміни швидкості комбайну, а система управління на базі ПДД-регулятора дає найкращі результати в режимах стабілізації швидкості.

На підставі цього розроблена система управління в якій основну керуючу роль виконує ПД-регулятор, ПДД-регулятор функціонує при стабілізації швидкості. Завдяки чому виконується компенсація статичної помилки, та стабілізація при виникненні збурень.

Таким чином розроблено систему управління, яка відповідає усім поставленим вимогам, та може бути використана на етапі розробки програмного забезпечення системи управління.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Цифрова модель системи управління

Згідно з завданням необхідно розробити програмне забезпечення системи управління та людино-машинний інтерфейс для SCADA системи zenon Supervisor 7.10. Програмне забезпечення повинно бути розроблено таким чином, щоб у якості об'єкта управління можливо було використовувати реальний об'єкт або його програмну модель. Виходячи з цього, модель системи, повинна бути представлена у вигляді двох програмних модулів: регулятора та об'єкта управління [12].

За для перетворення моделей математичного пакету MATLAB до тексту програми необхідно, щоб усі елементи даних моделей були цифрові. Таким чином, спочатку необхідно виконати заміну усіх неперервних елементів моделі системи управління на цифрові. Далі виконати поділ моделі на модель регулятора та модель об'єкта управління. Після чого перетворення цифрових моделей до програмних модулів, котрі будуть використані при розробці програмного забезпечення системи управління.

Модель регулятора виконана у вигляді окремого блоку тому перетворення буде почато з неї. В моделі регулятора усі неперервні елементи представлені у вигляді інтегруючих ланок, блоки яких необхідно замінити дискретними інтегруючими ланками. Отримана модель цифрової інтегруючої ланки представлена на рис. 4.1.

Модель засувки перероблена з Simulink моделі на код функції на мові MATLAB:

```
function output = Clamping(inpInt, inpSat, inpP, satUp, satLo)

% Зона нечутливості
if inpSat > satUp
    deadZone = inpSat - satUp;
elseif inpSat >= satLo
    deadZone = 0;
else
    deadZone = inpSat - satLo;
end

% Керуючий вплив збігається за знаком з сигналом управління на вході
% інтегратора, 1 означає, що керуючий вплив вийшло за обмеження,
```

```

% інтегратор насичений
equal1 = (sign(deadZone) == sign(inpInt));

% Керуючий вплив збігається за знаком з значенням регулятора перед посиленням
equal2 = (sign(inpSat) == sign(inpP));

% (deadZone ~= 0) - керуючий вплив вийшло за обмеження
if (deadZone ~= 0) && ((equal1 && equal2) || (~equal1 && ~equal2))
    output = 0;
else
    output = inpInt;
end

```



Рисунок 4.1 – Модель цифрової інтегруючої ланки

Отримана модель цифрової диференційної ланки представлена на рис. 4.2.

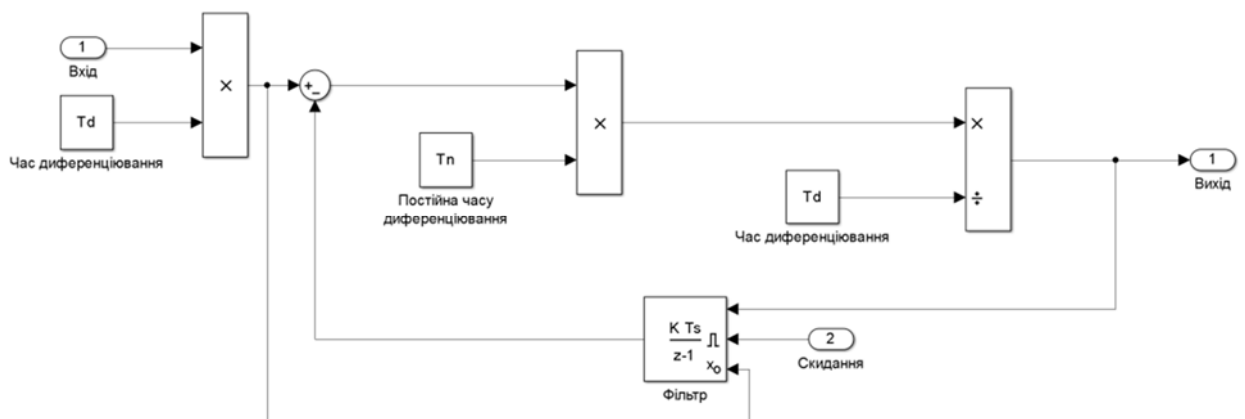


Рисунок 4.2 – Модель цифрової диференційної ланки

Модель об'єкта управління була винесена до окремого блока підсистеми. Передавальна функція моделі була перетворена за допомогою стандартної функції MATLAB з неперервної форми:

$$W(s) = \frac{1747,85}{0,04s+1}, \quad (4.1)$$

до цифрової:

$$W(z) = \frac{4,3642}{z-0,9975}. \quad (4.2)$$

Отримана цифрова модель об'єкта управління наведена на рис. 4.3.

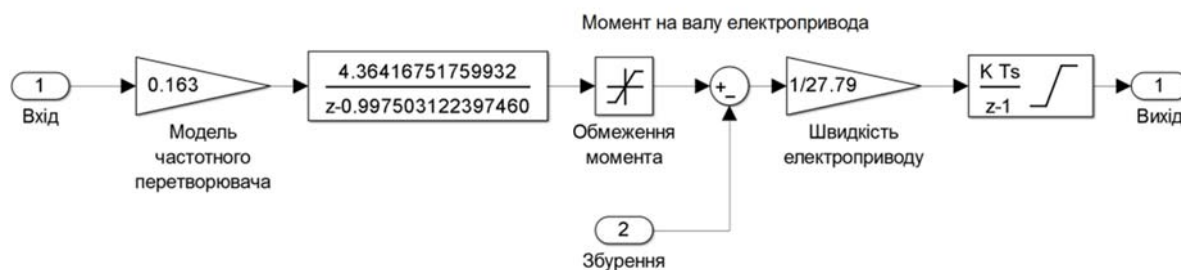


Рисунок 4.3 – Цифрова модель об'єкта управління

Порівняння результатів моделювання безперервної та цифрової моделі виконано за допомогою метода нормованого середньо квадратичного відхилення. Моделі збігаються на 99,98 %. Таким чином отримана цифрова модель може бути використана за для створення програмного забезпечення системи управління та перевірки його функціонування.

За допомогою стандартного засобу математичного пакету MATLAB блок підсистеми якій відповідає регулятору та блок підсистеми який відповідає об'єкту управління були перетворені до програмних модулів на мові стандарту MEK 61131-3 “Structured text” (див. дод. А) [13-15]. На базі отриманих модулів подалі буде розроблено програмне забезпечення для SCADA системи zenon.

4.2 Висновки по розділу

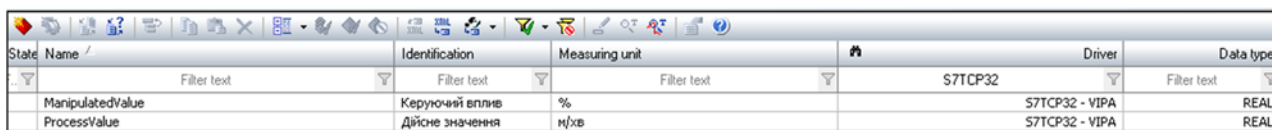
На базі безперервної моделі системи управління розроблена цифрова модель системи управління. З моделі відокремлені модель об'єкта управління та модель регулятора. За допомогою стандартних засобів математичного пакета MATLAB отримані програмні модулі, які реалізують об'єкт управління та регулятор на мові програмування Structured Text.

На базі програмних модулів розроблено програмне забезпечення системи управління для програмного програмованого логічного контролера zenon Logic який входить до складу SCADA системи zenon.

5 ЕКСПЕРИМЕНТАЛЬНИЙ РОЗДІЛ

5.1 Розробка програмного забезпечення системи управління

Відповідно до завдання необхідно розробити програмне забезпечення яке реалізує об'єкт управління та програмне забезпечення яке реалізує регулятор. Спочатку буде розроблено програмне забезпечення яке реалізує об'єкт управління. Для цього в SCADA системі zenon додамо драйвер “S7 TCP-IP driver” для підключення до програмованого логічного контролеру “VIPA 314-2AG12” за допомогою інтерфейсу Ethernet, а також відповідні змінні які наведені на рис. 5.1.



State	Name	Identification	Measuring unit	Driver	Data type
	ManipulatedValue	Керуючий вплив	%	S7TCP32 - VIPA	REAL
	ProcessValue	Дійсне значення	м/хв	S7TCP32 - VIPA	REAL

Рисунок 5.1 – Змінні програмованого логічного контролеру

Програмне забезпечення об'єкта управління буде функціонувати на програмному програмованому логічному контролері zenon Logic який в режимі симуляції буде реалізувати програмований логічний контролер VIPA та об'єкт управління.

На базі отриманого тексту програми з математичного пакету MATLAB розроблено користувацький функціональний блок “Object”. Текст програми якого розроблено на мові програмування Structured Text:

```

1_Temp := LREAL#4.36416751759932 * 1_State_0;

IF 1_State_1 >= LREAL#16.3 THEN
    1_State_1 := LREAL#16.3;
ELSIF 1_State_1 <= LREAL#-16.3 THEN
    1_State_1 := LREAL#-16.3;
END_IF;

IF 1_Temp >= LREAL#1126.0 THEN
    1_Temp := LREAL#1126.0;
ELSIF NOT (1_Temp > LREAL#-1126.0) THEN
    1_Temp := LREAL#-1126.0;
END_IF;

bo_Output := 1_State_1;

1_State_0 := (LREAL#0.163 * bi_Input) - (LREAL#-0.99750312239746 * 1_State_0);

```

```

l_State_1 := (((l_Temp - bi_Disturbance) * LREAL#0.035984166966534725) *
LREAL#0.0001) + l_State_1;
IF l_State_1 >= LREAL#16.3 THEN
    l_State_1 := LREAL#16.3;
ELSIF l_State_1 <= LREAL#-16.3 THEN
    l_State_1 := LREAL#-16.3;
END_IF;

```

В функціональному блоці виконується обмеження керуючого впливу та значення моменту. Після чого на підставі керуючого впливу розраховується дійсне значення. Змінні “l_State_0” та “l_State_1” відповідають стану аперіодичної та інтегруючої ланки відповідно.

Виклик функціонального блока “Object” виконується в тілі основної програми, яка наведена на рис. 5.2.



Рисунок 5.2 – Основна програма об’єкта управління

На входи блока об’єкта управління задається дійсне значення “ManipulatedValue” та значення збурення “l_Disturbance”. З виходу блока дійсне значення записується до змінної “ProcessValue”. Змінні “ManipulatedValue” та “ProcessValue” належать SCADA системі zenon. Змінна збурення “l_Disturbance” є локальною змінною яке належить до основної програми програмного програмованого логічного контролеру “Main”.

Відповідно до завдання програмне забезпечення контролера повинно функціонувати в SCADA системі zenon на програмному програмованому логічному контролері zenon Logic. Для цього в SCADA системі системі zenon додано відповідний проект zenon Logic “Controller”. Для зв’язку з SCADA системою додані змінні які наведені на рис. 5.3.

На базі отриманого тексту програми з математичного пакету MATLAB розроблено користувацький функціональний блок “Controller”. Текст програми якого написано на мові програмування Structured text.

Виклик функціонального блоку виконується в основному тілі програми, яке наведено на рис. 5.4.

State	Name	Identification	Measuring unit	Driver	Data type
	Filter text	Filter text	Filter text	Stratoning	Filter text
	Controller/Global/ManipulatedValue	Керуючий вплив	%	STRATONNG - zenon Logic: Controller	REAL
	Controller/Global/ProcessValue	Дійсне значення	м/хв	STRATONNG - zenon Logic: Controller	REAL

Рисунок 5.3 – Змінні програмованого логічного контролера

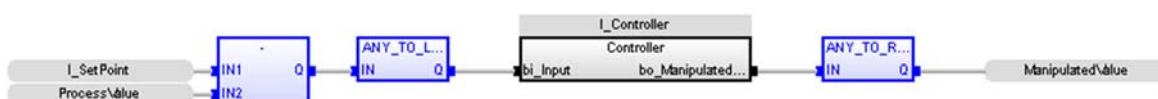


Рисунок 5.4 – Основна програма контролера

На вхід блока задається значення похибки системи управління яке обчислюється як різниця між уставкою “I_SetPoint” та дійсним значенням “ProcessValue”. На виході блока формується керуючий вплив який копіюється до змінної “ManipulatedValue”.

5.2 Розробка програмного забезпечення людино-машинного інтерфейсу

Програмне забезпечення людино-машинного інтерфейсу повинно мати п'ять вікон: інформації про об'єкт управління, технологічного процесу, споживаної потужності приводів різання хронологічного списку подій та інформаційного списку тривог [16-18]. Виходячи з того, що людино-машинний інтерфейс повинен відображати декілька зображень, він повинен складатися з двох шаблонів на одному з котрих повинні відображатися кнопки переключення зображення, а на іншому самі зображення.

Так як людино-машинний інтерфейс повинен відображатися на моніторі з роздільністю екрана 1 920x1 080 пкс, розмір шаблону основного зображення “MainScreenFrame” обрано рівним 1720x1080 пкс (рис. 5.5). На основному зображенні будуть відображатися технологічний процес, споживана потужність, хронологічний список подій та інформаційний список тривог. Розмір

допоміжного шаблону “ButtonScreenFrame” обрано рівним 200x1080 пкс, на ньому будуть відображатися кнопки зміни основного зображення.



Рисунок 5.5 – Шаблон зображення

Далі було розроблено зображення кнопок (рис. 5.6). На ньому розміщені часи, дата, кнопки зображені “ІНФОРМАЦІЯ”, “ТЕХНОЛОГІЯ”, “МОНІТОРИНГ”, “ХСП”, “ІСТ”. Крім того на ньому знаходяться кнопка “ОНОВИТИ” яка відповідає за оновлення людино-машинного інтерфейсу у випадку його перезавантаження та кнопка “ВИХІД” яка закриває людино-машинний інтерфейс.



Рисунок 5.6 – Зображення кнопок

На зображенні інформації про об’єкт управління (рис. 4.10) наведена технічні характеристики очисного комбайна УКД-200-250 та його зовнішній вигляд.

Відповідно до завдання було розроблено технологічне зображення яке наведено на рис. 5.7.



Рисунок 5.7 – Зображення інформації про об'єкт управління

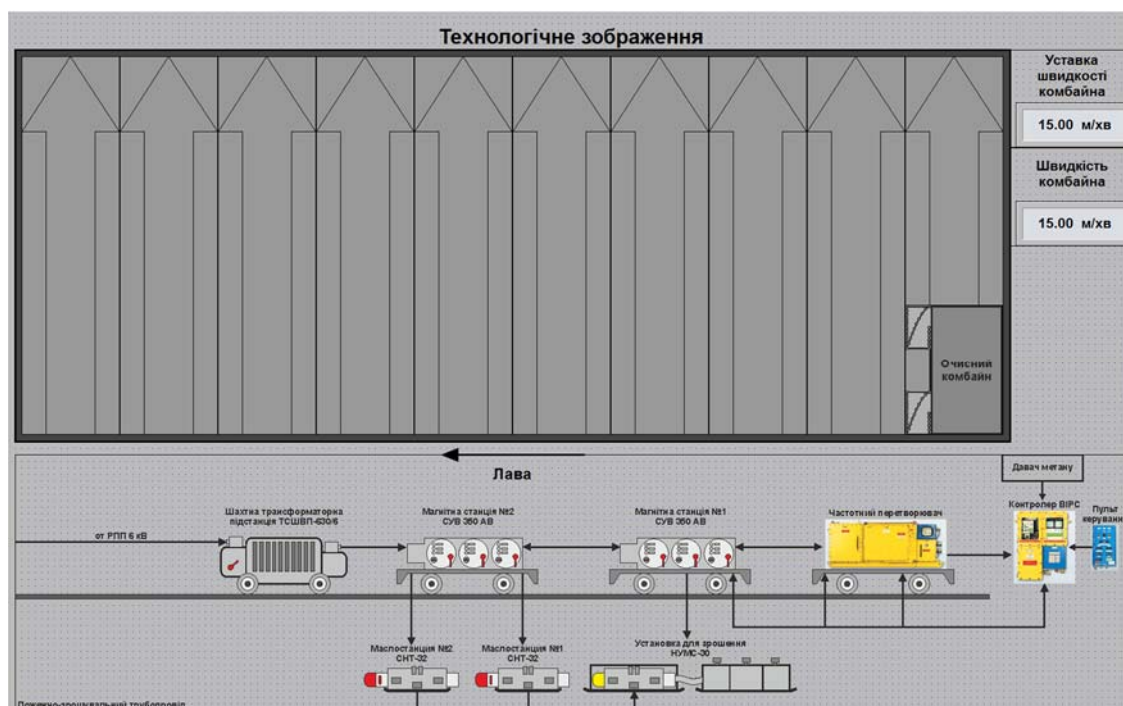


Рисунок 5.8 – Технологічне зображення

На технологічному зображенні наведена структура лави, уставка швидкості подачі комбайна та реальна швидкість його подачі. Крім того на зображенні за допомогою елементів “Bar” реалізовано відображення положення очисного комбайна.

За для моніторингу споживаної потужності розроблено відповідне зображення яке наведено на рис. 5.9. На зображенні наведено умовне відображення комбайна з двома приводами різання. Стан функціонування приводів відображається за допомогою відповідних індикаторів: привід різання №1, привід різання №2, коротке замикання, перевантаження силових ланцюгів. За допомогою тренду відображається зміна споживання потужності у часі.

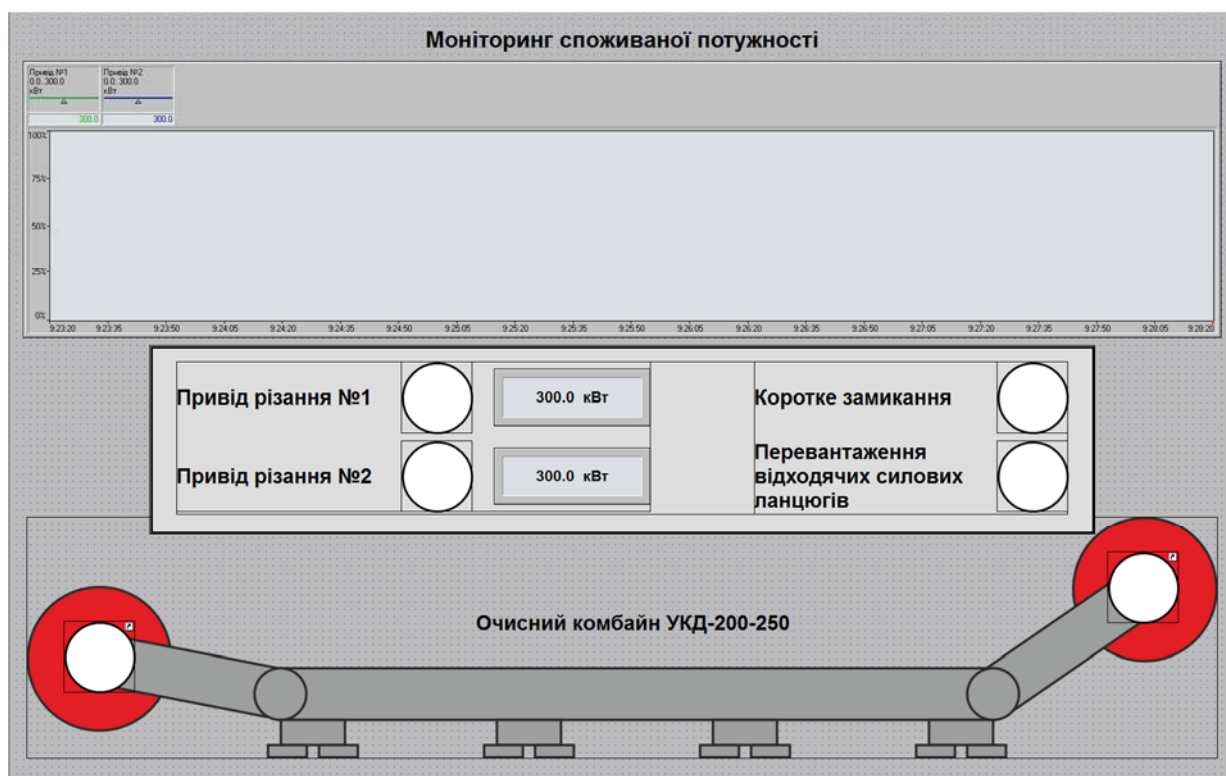


Рисунок 5.9 – Зображення моніторингу споживаної потужності

За для відображення списку подій використано зображення типу “Chronologic event list”. Відповідно до вимог в хронологічному списку подій записується інформація о вмиканні та вимиканні приводів різання, наявності короткого замикання або перевантаження. Розроблене зображення наведено на рис. 5.10.

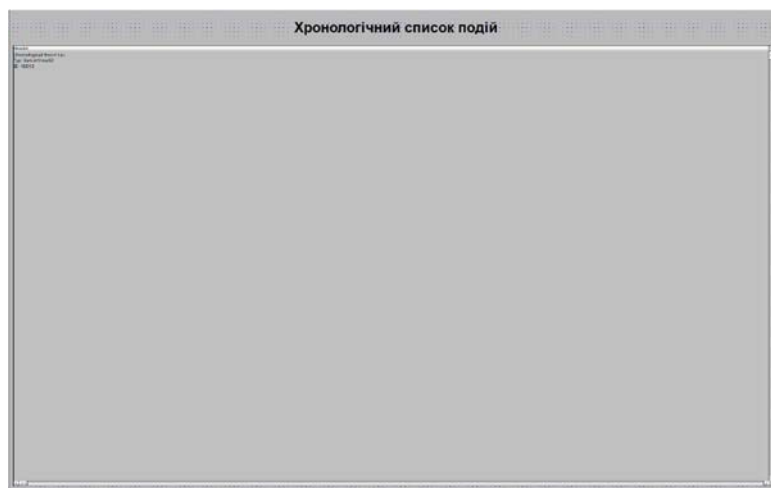


Рисунок 5.10 – Зображення хронологічного списку подій

За для відображення списку тривоги використано зображення типу “Alarm message list”. Відповідно до вимог в інформаційному списку тривоги записується інформація про короткі замикання та перевантаження. Розроблено зображення інформаційного списку тривоги наведено на рис. 5.11.

Для зв'язку зображень з даними програмованого логічного контролера який реалізує об'єкт управління та програмного програмованого логічного контролера який реалізує регулятор використані змінні, список котрих наведено на рис. 5.12.

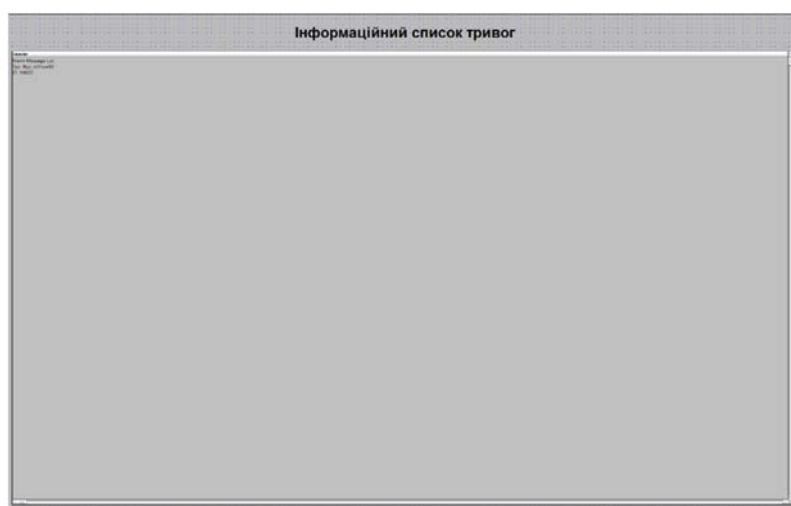


Рисунок 5.12 – Зображення інформаційного списку тривог

State	Name	Identification	Measuring unit	Data type
	Controller/Global/ManipulatedValue	Керуючий вплив	%	REAL
	Controller/Global/ProcessValue	Дійсне значення	м/хв	REAL
	CuttingDrive_1	Привід різання №1		BOOL
	CuttingDrive_2	Привід різання №2		BOOL
	CuttingDrivePhaseCurrentU_1	Привід різання №1 струм фази U	A	REAL
	CuttingDrivePhaseCurrentU_2	Привід різання №2 струм фази U	A	REAL
	CuttingDrivePhaseCurrentV_1	Привід різання №1 струм фази V	A	REAL
	CuttingDrivePhaseCurrentV_2	Привід різання №2 струм фази V	A	REAL
	CuttingDrivePhaseCurrentW_1	Привід різання №1 струм фази W	A	REAL
	CuttingDrivePhaseCurrentW_2	Привід різання №2 струм фази W	A	REAL
	CuttingDrivePhaseVoltageU_1	Привід різання №1 напруга фази U	V	REAL
	CuttingDrivePhaseVoltageU_2	Привід різання №2 напруга фази U	V	REAL
	CuttingDrivePhaseVoltageV_1	Привід різання №1 напруга фази V	V	REAL
	CuttingDrivePhaseVoltageV_2	Привід різання №2 напруга фази V	V	REAL
	CuttingDrivePhaseVoltageW_1	Привід різання №1 напруга фази W	V	REAL
	CuttingDrivePhaseVoltageW_2	Привід різання №2 напруга фази W	V	REAL
	CuttingDriverPower_1	Потужність споживана приводом різання №1	кВт	LREAL
	CuttingDriverPower_2	Потужність споживана приводом різання №2	кВт	LREAL
	Harvester/Global/CombineUp	Напрямок руху комбайна		BOOL
	Harvester/Global/CombineX	Позиція комбайна		REAL
	Harvester/Global/CombineY	Позиція комбайна		REAL
	Harvester/Global/Lava	Значення виборки лави		REAL
	Harvester/Global/LavaDown	Флаг відображення лави		BOOL
	Harvester/Global/LavaUp	Флаг відображення лави		BOOL
	Harvester/Global/State	Стан комбайна		BOOL
	Harvester/Global/Velocit	Швидкість комбайна	м/хв	REAL
	ManipulatedValue	Керуючий вплив	%	REAL
	OvervoltagePowerCircuit	Перевантаження відходящих силових мереж		BOOL
	ProcessValue	Дійсне значення	м/хв	REAL
	ShortCircuit	Коротке замикання		BOOL

Рисунок 5.13 – Список змінних

5.3 Перевірка функціонування програмного забезпечення

Перевірка функціонування програмного забезпечення виконувалася в режимі симуляції. Для того щоби провести повну перевірку було розроблено два допоміжних програмних забезпечення. Замість програмованого логічного контролера використовувався програмний програмований логічний контролер з моделлю об'єкта управління.

В процесі перевірки було встановлено, що усі індикатори технологічного процесу відображають правильну інформацію про швидкість та положення очисного комбайна (рис. 5.14).

На тренді відображається стан споживання потужності, крім того індикатори відображають стан електроприводів різання, датчиків короткого замикання та перенавантаження (рис. 5.15).

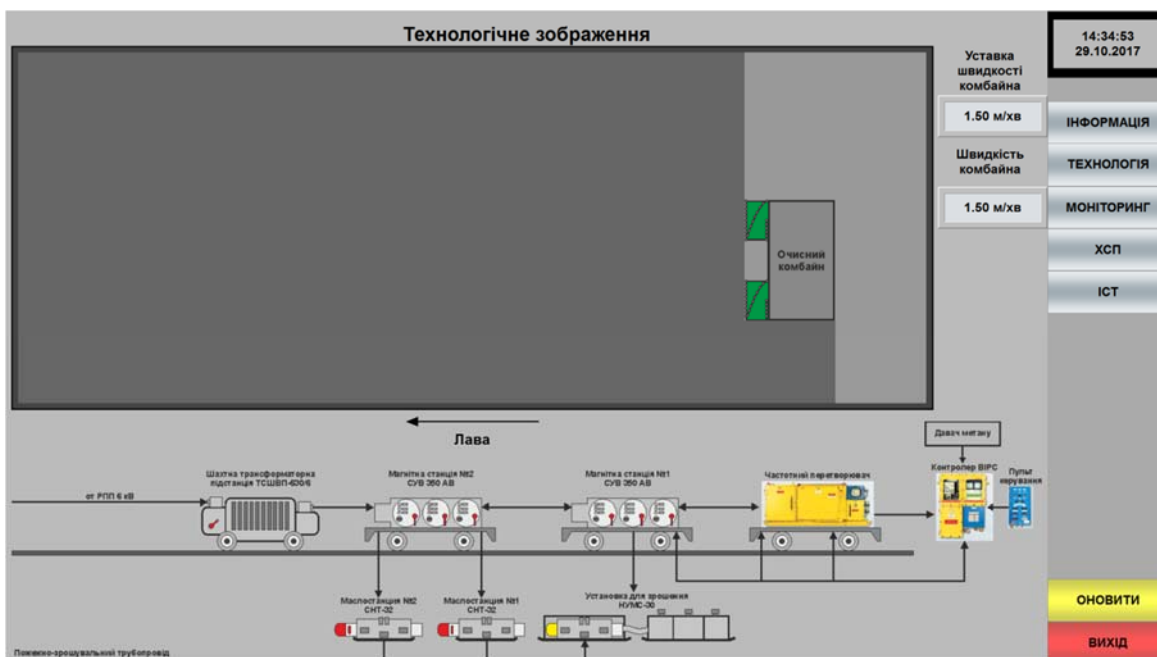


Рисунок 5.14 – Технологічне зображення

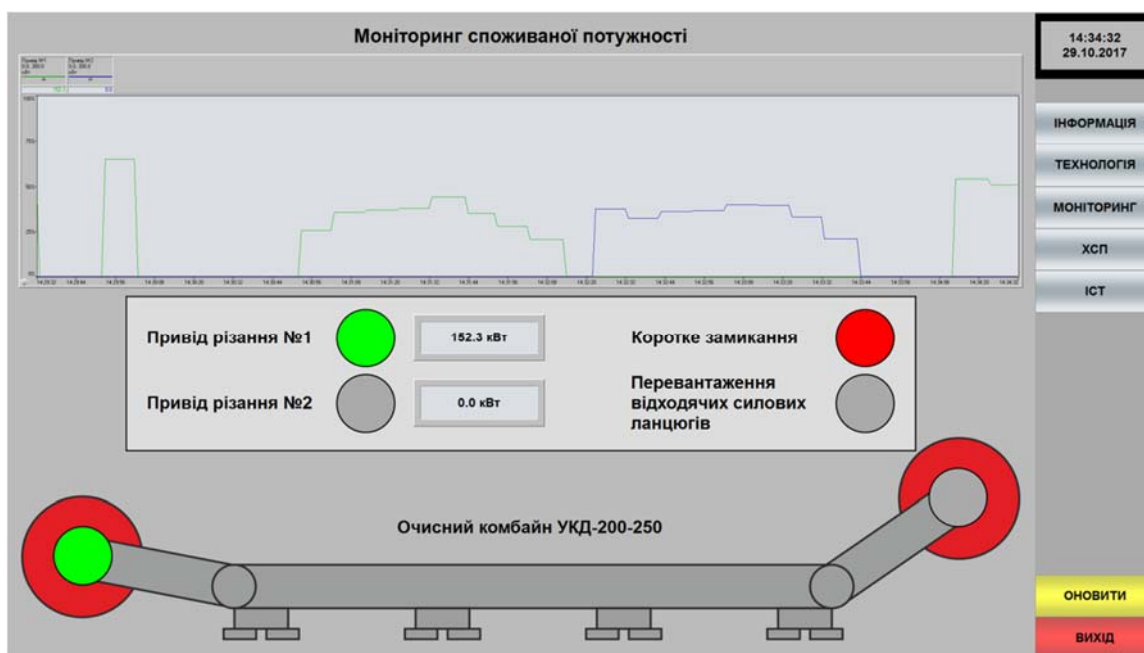


Рисунок 5.15 – Зображення моніторингу споживаної потужності

На зображенні хронологічного списку подій відображаються події які виникають під час роботи системи управління швидкістю очисного комбайна. Серед подій є включення та виключення приводів різання, виникнення короткого замикання та перевантаження (рис. 5.16).

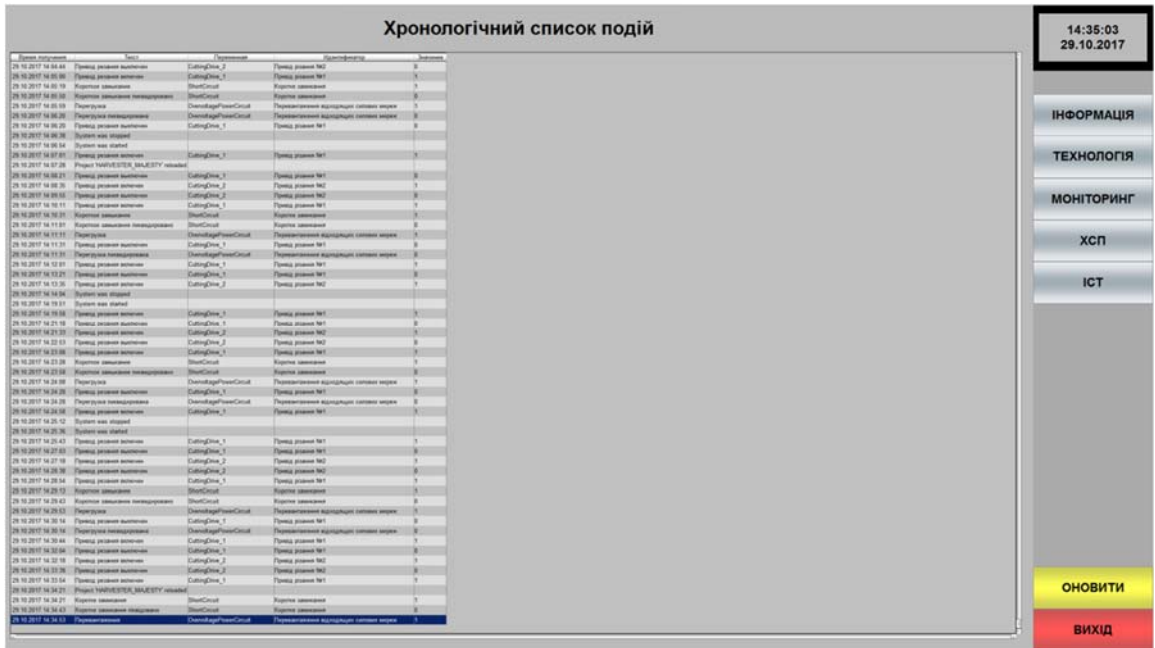


Рисунок 5.16 – Зображення хронологічного списку подій

На зображенні інформаційного списку тривог відображаються аварійні ситуації які потребують реакції оператора, та підтвердження їх отримання (рис. 5.17).

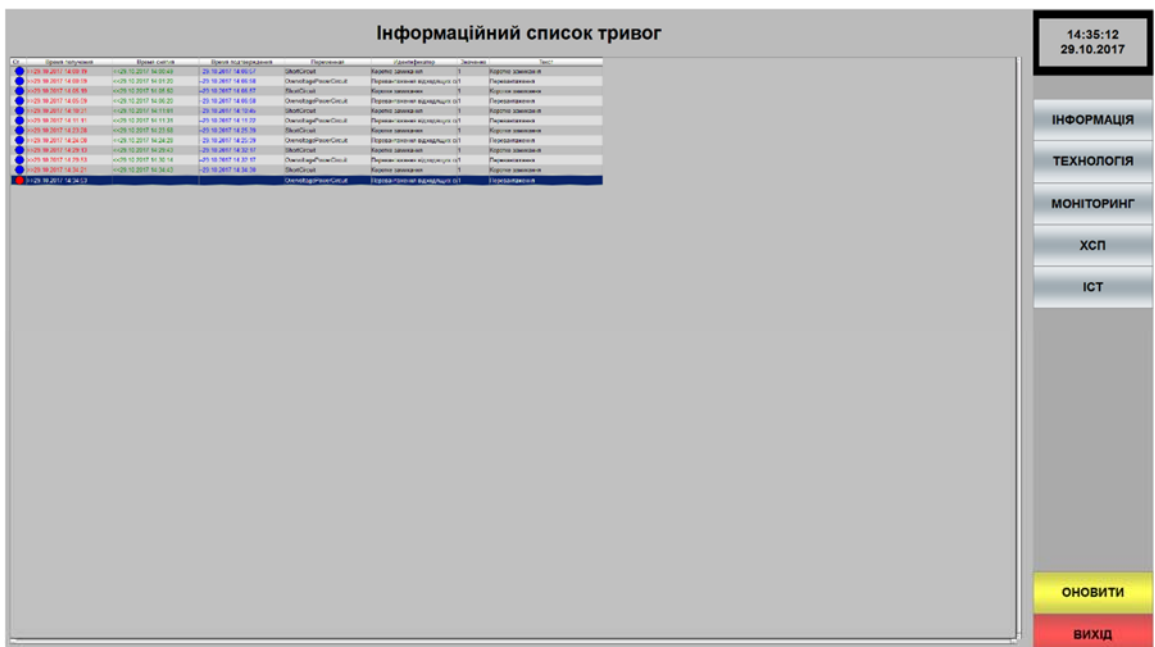


Рисунок 5.17 – Зображення інформаційного списку тривог

5.4 Висновки по розділу

В SCADA системі zenon розроблено людино машинний інтерфейс, який включає відображення інформації про об'єкт управління, технологічний процес, споживану потужність, хронологічний список подій та інформаційний список тривоги.

Виконано перевірку функціонування програмного забезпечення. Програмне забезпечення функціонує відповідно до заданих алгоритмів та забезпечує виконання поставлених завдань.

ВИСНОВКИ

На основі дипломної роботи магістра можна зробити наступні висновки:

1. Автоматизована система управління швидкістю очисного комбайна може бути використана на вугільних шахтах небезпечних по пилу та газу.

2. Особлива увага у роботі привернута до розробки моделей об'єкта управління, регулятора та системи управління в загалом та розробці програмного забезпечення.

3. Для виконання усіх вимог до роботи системи управління необхідно використовувати два регулятора.

4. Пропорційно-диференціюючий регулятор забезпечує найкращі показники якості при зміні уставки швидкості комбайна.

5. Пропорційно-інтегрально-диференціюючий регулятор забезпечує найкращі якості при стабілізації швидкості та для реагування на динамічні збурення.

6. За для одночасного використання обох регуляторів необхідно використовувати такі структури при яких виконується безударне перемикання.

7. На базі стандартних структур розроблено безударний ПД та ПД-регулятори. На базі яких в свою чергу розроблено регулятор швидкості очисного комбайна.

8. На базі безперервної системи управління розроблена цифрова модель, з використанням якої отримані програмні модулі регулятора та об'єкта управління на мові Structured text.

9. На підставі програмних модулів розроблено програмне забезпечення системи управління для програмного програмованого логічного контролера zenon Logic.

10. Розроблено апаратне забезпечення з використанням ПЛК, та програмне забезпечення людино-машинного інтерфейсу, з використанням Zenon, яке складається з п'яти зображень. На зображеннях відображено технологічний процес, змінення споживання потужності та інформаційні списки.

Таким чином у результаті виконання дипломної роботи розроблена модель об'єкта управління. На підставі якої виконані дослідження завдяки яким вдалося встановити структуру регулятора. На базі отриманої структури розроблено регулятор швидкості очисного комбайна. Отриманий регулятор було перетворено з безперервної форми до цифрової. На базі цифрової форми розроблено програмне забезпечення системи управління, яке використано при розробці людино-машинного інтерфейсу системи управління.

ПЕРЕЛІК ПОСИЛАНЬ

1. Горные машины для подземной добычи угля / П.А. Горбатов, Г.В. Петрушкин, Н.М. Лысенко и др. – Донецк: Донецький національний технічний університет, 2006. – 669 с.
2. Морозов В.И. Очистные комбайны. Справочник / В.И. Морозов, В.И. Чуденков, Н.В. Сурина. – М.: Высшее горное образование, 2006. – 650 с
3. Зайков В. И. Эксплуатация горных машин и оборудования / В.И. Зайков, Г.П. Берлявский. – Москва: Высшее горное образование, 2006. – 257 с.
4. Машины и оборудование для шахт и рудников. Справочник / С.Х. Клорикьян, В.В. Старичнев, М.А. Сребный и др. – М.: Московский государственный горный университет, 2002. – 471 с.
5. Балабышко А.М. Гидропривод механизированных крепей / А.М. Балабышко В.П. Ружицкий, К.М. Первов. – М.: Высшее горное образование, 2002. – 138 с.
6. Бубликов А.В. Автоматическое управление добычным комбайном в режиме стабилизации скорости подачи / А.В. Бубликов. – Д.: Збірник наукових праць Національного гірничого університету. – 2013. - №40. С. 91-97
7. Ключев В.И. Теория электропривода: Учеб. для вузов. – 2-е изд. перераб. и доп. / В.И. Ключев. – М.: Энергоатомиздат, 2001. – 704 с.
8. Иванов А.А. Теория автоматического управления и регулирования / А.А. Иванов. – М.: изд-во «Недра», 1970. – с. 352.
9. Шаруда В.Г. Практикум з теорії автоматичного управління: Навчальний посібник. – Дніпропетровськ: Національна гірнична академія України, 2002. – 414 с.
10. Шаруда В.Г. Методи аналізу і синтезу систем автоматичного керування: Навч. посіб / В.Г. Шаруда, В.В. Ткачов, М.П. Фількін. – Д.: Нац. Гірнич. ун-т., 2008. – 543 с.

11. Никулин Е.А. Основы теории автоматического управления. Частотные методы анализа и синтеза / Учеб. пособие для вузов. – СПб.: БХВ-Петербург, 2004. – 640 с.
12. Дорф Р. Современные системы управления / Р. Дорф, Р. Бишоп; пер. С англ. Б.И. Копылова. – М.: Лаборатория Базовых Знаний, 2012. – 832 с.
13. Харазов В.Г. Интегрирование системы управления технологическим процессом / В.Г. Харазов. – СПб: Профессия, 2009. – 592 с.
14. Парк Дж. Сбор данных в системах контроля и управления. Практическое руководство / Дж. Парк, С. Маккей. – М.: ООО «Группа ИДТ», 2006. – 504 с.
15. Динесенко В.В. Компьютерное управление технологическим процессом, экспериментом, оборудованием / В.В. Динесенко. – М.: Горячая линия-Телеком, 2009. – 608 с.
16. Петров И.В. Программируемые контроллеры. Стандартные языки и инструменты / И.В. Петров. – М.: СОЛОН-Пресс, 2003. – 256 с.
17. Парр Э. Програмируемые контролеры: руководство для инженера / Э. Парр. – М.: БИНОМ. Лаборатория знаний, 2007. – 516 с.
18. Минаев И.Г. Программируемые логические контроллеры. Практическое руководство для начинающего инженера / И.Г. Минаев, В.В. Самойленко. – Ставрополь: АРГУС, 2009. – 100 с.

ДОДАТОК А ПРОГРАМНІ МОДУЛІ СИСТЕМИ УПРАВЛІННЯ

A.1 Модуль об'єкта управління

```

(*)
*
* File: ControlSystem_z_Final_ScreenShot.exp
*
* IEC 61131-3 Structured Text (ST) code generated for subsystem
"ControlSystem_z_Final_ScreenShot/Об'єкт управління"
*
* Model name                : ControlSystem_z_Final_ScreenShot
* Model version             : 1.369
* Model creator             : bukap
* Model last modified by    : bukap
* Model last modified on    : Sat Oct 28 21:31:57 2017
* Model sample time        : 0.0001s
* Subsystem name           : ControlSystem_z_Final_ScreenShot/Об'єкт
управління
* Subsystem sample time    : 0.0001s
* Simulink PLC Coder version : 1.6 (R2013b) 08-Aug-2013
* ST code generated on     : Sat Oct 28 21:34:36 2017
*
* Target IDE selection     : 3S CoDeSys 2.3
* Test Bench included      : No
*
*)
FUNCTION_BLOCK sfun_target
VAR_INPUT
    ssMethodType: SINT;
    u: LREAL;
    u_n: LREAL;
END_VAR
VAR_OUTPUT
    b_u: LREAL;
END_VAR
VAR
    Integrator_DSTATE: LREAL;
    DiscreteTransferFcn_states: LREAL;
END_VAR
VAR_TEMP
    rtb_DiscreteTransferFcn: LREAL;
END_VAR
CASE ssMethodType OF
    SS_INITIALIZE:

        (* InitializeConditions for DiscreteTransferFcn: '<S1>/Discrete Transfer
Fcn' *)
        DiscreteTransferFcn_states := 0.0;

        (* InitializeConditions for DiscreteIntegrator: '<S1>/Integrator' *)
        Integrator_DSTATE := 0.0;
    SS_STEP:

        (* DiscreteTransferFcn: '<S1>/Discrete Transfer Fcn' *)
        rtb_DiscreteTransferFcn := 4.36416751759932 *
DiscreteTransferFcn_states;

        (* DiscreteIntegrator: '<S1>/Integrator' *)
        IF Integrator_DSTATE >= 16.3 THEN
            Integrator_DSTATE := 16.3;
        ELSIF Integrator_DSTATE <= -16.3 THEN
            Integrator_DSTATE := -16.3;

```

```

END_IF;

(* Saturate: '<S1>/Обмеження моменту' *)
IF rtb_DiscreteTransferFcn >= 1126.0 THEN
    rtb_DiscreteTransferFcn := 1126.0;
ELSIF NOT (rtb_DiscreteTransferFcn > -1126.0) THEN
    rtb_DiscreteTransferFcn := -1126.0;
END_IF;
(* End of Saturate: '<S1>/Обмеження моменту' *)

(* Output: '<Root>/Вихід' incorporates:
 * DiscreteIntegrator: '<S1>/Integrator' *)
b_u := Integrator_DSTATE;

(* Update for DiscreteTransferFcn: '<S1>/Discrete Transfer Fcn'
incorporates:
 * Gain: '<S1>/Модель частотного перетворювача'
 * Update for Inport: '<Root>/Вхід' *)
DiscreteTransferFcn_states := (0.163 * u) - (-0.99750312239746 *
DiscreteTransferFcn_states);

(* Update for DiscreteIntegrator: '<S1>/Integrator' incorporates:
 * Gain: '<S1>/Швидкість електроприводу'
 * Update for Inport: '<Root>/Збурення'
 * Sum: '<S1>/Sum3' *)
Integrator_DSTATE := (((rtb_DiscreteTransferFcn - u_n) *
0.035984166966534725) * 0.0001) + Integrator_DSTATE;
IF Integrator_DSTATE >= 16.3 THEN
    Integrator_DSTATE := 16.3;
ELSIF Integrator_DSTATE <= -16.3 THEN
    Integrator_DSTATE := -16.3;
END_IF;
(* End of Update for DiscreteIntegrator: '<S1>/Integrator' *)
END_CASE;
END_FUNCTION_BLOCK
VAR_GLOBAL CONSTANT
    SS_INITIALIZE: SINT := 0;
    SS_STEP: SINT := 1;
END_VAR
VAR_GLOBAL
END_VAR

```

A.2 Модуль регулятора

```

(*
 *
 * File: ControlSystem_z_Final_ScreenShot.exp
 *
 * IEC 61131-3 Structured Text (ST) code generated for subsystem
"ControlSystem_z_Final_ScreenShot/Регулятор"
 *
 * Model name                : ControlSystem_z_Final_ScreenShot
 * Model version              : 1.369
 * Model creator               : bukap
 * Model last modified by     : bukap
 * Model last modified on     : Sat Oct 28 21:31:57 2017
 * Model sample time          : 0.0001s
 * Subsystem name              : ControlSystem_z_Final_ScreenShot/Регулятор
 * Subsystem sample time      : 0.0001s
 * Simulink PLC Coder version : 1.6 (R2013b) 08-Aug-2013
 * ST code generated on       : Sat Oct 28 21:32:16 2017
 *
 * Target IDE selection        : 3S CoDeSys 2.3

```

```

* Test Bench included           : No
*
*)
FUNCTION_BLOCK sfun_target
VAR_INPUT
  _ssMethodType: SINT;
  u: LREAL;
END_VAR
VAR_OUTPUT
  u_n: LREAL;
END_VAR
VAR
  _PrevResetState_m: SINT;
  _DSTATE_l: LREAL;
  _IC_LOADING_o: USINT;
  Integrator_PrevResetState: SINT;
  Integrator_DSTATE: LREAL;
  _PrevResetState: SINT;
  _DSTATE: LREAL;
  _IC_LOADING: USINT;
END_VAR
VAR_TEMP
  equal: BOOL;
  guard: BOOL;
  guard_0: BOOL;
  guard_1: BOOL;
  guard_2: BOOL;
  rtb_Switch2_m: LREAL;
  rtb_Divide: LREAL;
  rtb_Divide2: LREAL;
  rtb_LowerRelop1: BOOL;
  rtb_u: LREAL;
  rtb_Divide2_l: LREAL;
  rtb_Sum14_p: LREAL;
  rtb_Divide_j: LREAL;
  rtb_output: LREAL;
  y: LREAL;
  rtb_Compare: BOOL;
  rtb_u_d: BOOL;
END_VAR
guard := FALSE;
guard_0 := FALSE;
guard_1 := FALSE;
guard_2 := FALSE;
CASE ssMethodType OF
  SS_INITIALIZE:

    (* InitializeConditions for DiscreteIntegrator: '<S10>/Фільтр' *)
    _IC_LOADING := 1;
    _PrevResetState := 0;

    (* InitializeConditions for DiscreteIntegrator: '<S5>/Integrator' *)
    Integrator_DSTATE := 0.0;
    Integrator_PrevResetState := 0;

    (* InitializeConditions for DiscreteIntegrator: '<S7>/Фільтр' *)
    _IC_LOADING_o := 1;
    _PrevResetState_m := 0;
  SS_STEP:

    (* RelationalOperator: '<S1>/Умова перемикання' incorporates:
    * Abs: '<S1>/Abs'
    * Constant: '<S1>/Уставка перемикання регуляторів'
    * Inport: '<Root>/Помилка управління' *)

```

```

rtb_u_d := ABS(u) < 0.1;

(* Product: '<S10>/Divide' incorporates:
 * Constant: '<S10>/Час диференціювання'
 * Inport: '<Root>/Помилка управління' *)
rtb_Divide := u * 0.026;

(* DiscreteIntegrator: '<S10>/Фільтр' *)
IF USINT_TO_DINT(_IC_LOADING) <> 0 THEN
  _DSTATE := rtb_Divide;
END_IF;
IF rtb_u_d OR (SINT_TO_DINT(_PrevResetState) <> 0) THEN
  _DSTATE := rtb_Divide;
END_IF;

(* Product: '<S10>/Divide2' incorporates:
 * Constant: '<S10>/ Час диференціювання '
 * Constant: '<S10>/Постійна часу диференціювання'
 * DiscreteIntegrator: '<S10>/Фільтр'
 * Product: '<S10>/Divide1'
 * Sum: '<S10>/Sum' *)
rtb_Divide2 := ((rtb_Divide - _DSTATE) * 83.3) / 0.026;

(* Product: '<S4>/Divide' incorporates:
 * Constant: '<S4>/Коефіцієнт посилення'
 * Inport: '<Root>/Помилка управління'
 * Sum: '<S4>/Sum14' *)
rtb_Switch2_m := (u + rtb_Divide2) * 5.276;

(* RelationalOperator: '<S9>/LowerRelop1' incorporates:
 * Constant: '<S4>/Максимальне значення керуючого впливу ' *)
rtb_LowerRelop1 := rtb_Switch2_m > 100.0;

(* Switch: '<S9>/Switch' incorporates:
 * Constant: '<S4>/Мінімальне значення керуючого впливу '
 * RelationalOperator: '<S9>/UpperRelop' *)
IF rtb_Switch2_m < -100.0 THEN
  rtb_Switch2_m := -100.0;
END_IF;
(* End of Switch: '<S9>/Switch' *)

(* Switch: '<S9>/Switch2' incorporates:
 * Constant: '<S4>/Максимальне значення керуючого впливу ' *)
IF rtb_LowerRelop1 THEN
  rtb_Switch2_m := 100.0;
END_IF;
(* End of Switch: '<S9>/Switch2' *)

(* RelationalOperator: '<S2>/Compare' incorporates:
 * Constant: '<S2>/Constant' *)
rtb_Compare := NOT rtb_u_d;

(* DiscreteIntegrator: '<S5>/Integrator' *)
IF rtb_Compare OR (SINT_TO_DINT(Integrator_PrevResetState) <> 0) THEN
  Integrator_DSTATE := 0.0;
END_IF;
rtb_u := Integrator_DSTATE;

(* Product: '<S7>/Divide' incorporates:
 * Constant: '<S7>/Час диференціювання'
 * Inport: '<Root>/Помилка управління' *)
rtb_Divide := u * 0.0276;

(* DiscreteIntegrator: '<S7>/Фільтр' *)

```

```

IF USINT_TO_DINT( _IC_LOADING_o) <> 0 THEN
    _DSTATE_l := rtb_Divide;
END_IF;
IF rtb_Compare OR (SINT_TO_DINT( _PrevResetState_m) <> 0) THEN
    _DSTATE_l := rtb_Divide;
END_IF;

(* Product: '<S7>/Divide2' incorporates:
 * Constant: '<S7>/ Час диференціювання '
 * Constant: '<S7>/Постійна часу диференціювання'
 * DiscreteIntegrator: '<S7>/Фільтр'
 * Product: '<S7>/Divide1'
 * Sum: '<S7>/Sum' *)
rtb_Divide2_l := ((rtb_Divide - _DSTATE_l) * 80.0) / 0.0276;

(* Sum: '<S3>/Sum14' incorporates:
 * DiscreteIntegrator: '<S5>/Integrator'
 * Inport: '<Root>/Помилка управління' *)
rtb_Sum14_p := (u + Integrator_DSTATE) + rtb_Divide2_l;

(* Product: '<S3>/Divide' incorporates:
 * Constant: '<S3>/Коефіцієнт посилення' *)
rtb_Divide_j := rtb_Sum14_p * 30.0;

(* Switch: '<S6>/Switch' incorporates:
 * Constant: '<S3>/Мінімальне значення керуючого впливу '
 * RelationalOperator: '<S6>/UpperRelop' *)
IF rtb_Divide_j < -100.0 THEN
    rtb_Divide := -100.0;
ELSE
    rtb_Divide := rtb_Divide_j;
END_IF;
(* End of Switch: '<S6>/Switch' *)

(* Switch: '<S6>/Switch2' incorporates:
 * Constant: '<S3>/Максимальне значення керуючого впливу '
 * RelationalOperator: '<S6>/LowerRelop1' *)
IF rtb_Divide_j > 100.0 THEN
    rtb_Divide := 100.0;
END_IF;
(* End of Switch: '<S6>/Switch2' *)

(* MultiPortSwitch: '<S1>/Вибір регулятора' *)
rtb_LowerRelop1 := FALSE;
CASE BOOL_TO_DINT(rtb_u_d) OF
    0:
        rtb_u := rtb_Switch2_m;
    1:
        rtb_LowerRelop1 := TRUE;
ELSE
    rtb_LowerRelop1 := TRUE;
END_CASE;
IF rtb_LowerRelop1 THEN
    rtb_u := rtb_Divide;
END_IF;
(* End of MultiPortSwitch: '<S1>/Вибір регулятора' *)

(* Outport: '<Root>/Керуючий вплив' *)
u_n := rtb_u;

(* Product: '<S5>/Divide' incorporates:
 * Constant: '<S5>/Час інтегрування'
 * Inport: '<Root>/Помилка управління' *)
rtb_Divide := u / 0.1;

```

```

(* MATLAB Function: '<S5>/Засувка' incorporates:
 * Constant: '<S5>/Мінімальне значення керуючого впливу'
 * Constant: '<S5>/Максимальне значення керуючого впливу' *)
(* MATLAB Function 'Регулятор/ПІД-регулятор/Інтегруюча ланка/Засувка':
'<S8>:1' *)
(* Зона нечутливості *)
IF rtb_Divide_j > 100.0 THEN
  (* '<S8>:1:4' *)
  (* '<S8>:1:5' *)
  rtb_Switch2_m := rtb_Divide_j - 100.0;
ELSIF rtb_Divide_j >= -100.0 THEN
  (* '<S8>:1:6' *)
  (* '<S8>:1:7' *)
  rtb_Switch2_m := 0.0;
ELSE
  (* '<S8>:1:9' *)
  rtb_Switch2_m := rtb_Divide_j - -100.0;
END_IF;
(* Керуючий вплив збігається за знаком з сигналом управління на вході *)
(* інтегратора, 1 означає, що керуючий вплив вийшло за обмеження, *)
(* інтегратор насичений *)
(* '<S8>:1:15' *)
IF rtb_Divide < 0.0 THEN
  rtb_u := -1.0;
ELSIF rtb_Divide > 0.0 THEN
  rtb_u := 1.0;
ELSE
  rtb_u := rtb_Divide;
END_IF;
IF rtb_Switch2_m < 0.0 THEN
  y := -1.0;
ELSIF rtb_Switch2_m > 0.0 THEN
  y := 1.0;
ELSE
  y := rtb_Switch2_m;
END_IF;
rtb_LowerRelop1 := y = rtb_u;
(* Керуючий вплив збігається за знаком з значенням регулятора перед
посиленням *)
(* '<S8>:1:18' *)
IF rtb_Sum14_p < 0.0 THEN
  rtb_Sum14_p := -1.0;
ELSIF rtb_Sum14_p > 0.0 THEN
  rtb_Sum14_p := 1.0;
END_IF;
IF rtb_Divide_j < 0.0 THEN
  rtb_Divide_j := -1.0;
ELSIF rtb_Divide_j > 0.0 THEN
  rtb_Divide_j := 1.0;
END_IF;
equal := rtb_Divide_j = rtb_Sum14_p;
(* (deadZone ~= 0) - керуючий вплив вийшло за обмеження *)
IF rtb_Switch2_m <> 0.0 THEN
  (* '<S8>:1:21' *)
  IF rtb_LowerRelop1 THEN
    (* '<S8>:1:21' *)
    IF equal THEN
      (* '<S8>:1:21' *)
      guard_0 := TRUE;
    ELSE
      guard_2 := TRUE;
    END_IF;
  ELSE
    guard_2 := TRUE;
  END_IF;
ELSE
  guard_2 := TRUE;
END_IF;

```

```

        guard_2 := TRUE;
    END_IF;
ELSE
    guard := TRUE;
END_IF;
IF guard_2 THEN
    IF NOT rtb_LowerRelop1 THEN
        (* '<S8>:1:21' *)
        IF NOT equal THEN
            (* '<S8>:1:21' *)
            guard_0 := TRUE;
        ELSE
            guard_1 := TRUE;
        END_IF;
    ELSE
        guard_1 := TRUE;
    END_IF;
END_IF;
IF guard_1 THEN
    guard := TRUE;
END_IF;
IF guard_0 THEN
    (* '<S8>:1:22' *)
    rtb_output := 0.0;
END_IF;
IF guard THEN
    (* '<S8>:1:24' *)
    rtb_output := rtb_Divide;
END_IF;
(* End of MATLAB Function: '<S5>/Засувка' *)

(* Update for DiscreteIntegrator: '<S10>/Фільтр' *)
_IC_LOADING := 0;
IF NOT rtb_u_d THEN
    _DSTATE := (0.0001 * rtb_Divide2) + _DSTATE;
END_IF;
IF rtb_u_d THEN
    _PrevResetState := 1;
ELSE
    _PrevResetState := 0;
END_IF;
(* End of Update for DiscreteIntegrator: '<S10>/Фільтр' *)

(* Update for DiscreteIntegrator: '<S5>/Integrator' *)
IF NOT rtb_Compare THEN
    Integrator_DSTATE := (0.0001 * rtb_output) + Integrator_DSTATE;
END_IF;
IF rtb_Compare THEN
    Integrator_PrevResetState := 1;
ELSE
    Integrator_PrevResetState := 0;
END_IF;
(* End of Update for DiscreteIntegrator: '<S5>/Integrator' *)

(* Update for DiscreteIntegrator: '<S7>/Фільтр' *)
_IC_LOADING_o := 0;
IF NOT rtb_Compare THEN
    _DSTATE_1 := (0.0001 * rtb_Divide2_1) + _DSTATE_1;
END_IF;
IF rtb_Compare THEN
    _PrevResetState_m := 1;
ELSE
    _PrevResetState_m := 0;
END_IF;

```



```
      (* End of Update for DiscreteIntegrator: '<S7>/Фільтр' *)
END_CASE;
END_FUNCTION_BLOCK
VAR_GLOBAL CONSTANT
  SS_INITIALIZE: SINT := 0;
  SS_STEP: SINT := 1;
END_VAR
VAR_GLOBAL
END_VAR
```

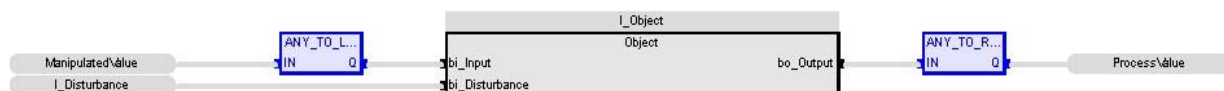
ДОДАТОК Б ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМИ УПРАВЛІННЯ

Б.1 Програмне забезпечення об'єкта управління

Б.1.1 Список програм

[FBD] Main (* Основна програма *)
 [ST] Object (* Об'єкт управління *)

Б.1.2 Основна програма Main



```
l_Object (Object)
ManipulatedValue (* Керуючий вплив *)
l_Disturbance (* Збурення *)
ProcessValue (* Дійсне значення *)
```

Б.1.3 Користувацький функціональний блок Object

```
l_Temp := LREAL#4.36416751759932 * l_State_0;

IF l_State_1 >= LREAL#16.3 THEN
  l_State_1 := LREAL#16.3;
ELSIF l_State_1 <= LREAL#-16.3 THEN
  l_State_1 := LREAL#-16.3;
END_IF;

IF l_Temp >= LREAL#1126.0 THEN
  l_Temp := LREAL#1126.0;
ELSIF NOT (l_Temp > LREAL#-1126.0) THEN
  l_Temp := LREAL#-1126.0;
END_IF;

bo_Output := l_State_1;

l_State_0 := (LREAL#0.163 * bi_Input) - (LREAL#-0.99750312239746 * l_State_0);

l_State_1 := (((l_Temp - bi_Disturbance) * LREAL#0.035984166966534725) * LREAL#0.0001) + l_State_1;
IF l_State_1 >= LREAL#16.3 THEN
  l_State_1 := LREAL#16.3;
ELSIF l_State_1 <= LREAL#-16.3 THEN
  l_State_1 := LREAL#-16.3;
END_IF;
l_Temp (* Допоміжна змінна *)
l_State_0 (* Стан аперіодичної ланки *)
l_State_1 (* Стан інтегруючої ланки *)
bo_Output (* Дійсне значення *)
bi_Input (* Керуючий вплив *)
bi_Disturbance (* Збурення *)
```

Б.1.4 Список змінних

```
Global variables
ManipulatedValue (* Керуючий вплив *)
REAL
Embedded: <Symbol>
```

```

ProcessValue (* Дійсне значення *)
    REAL
    Embedded: <Symbol>
Main
l_Object
    Object
l_Disturbance (* Збурення *)
    LREAL
Object
bi_Input (* Керуючий вплив *)
    LREAL
bi_Disturbance (* Збурення *)
    LREAL
bo_Output (* Дійсне значення *)
    LREAL
l_State_0 (* Стан аперіодичної ланки *)
    LREAL
l_State_1 (* Стан інтегруючої ланки *)
    LREAL
l_Temp (* Допоміжна змінна *)
    LREAL

```

Б.2 Програмне забезпечення регулятора

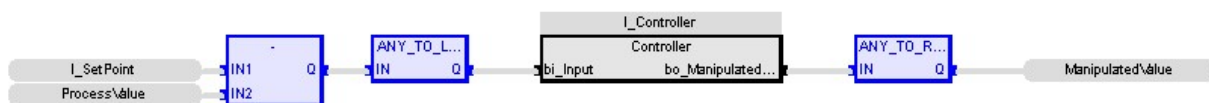
Б.2.1 Список програм

```

[FB] Main (* Основна програма *)
[ST] Controller (* Регулятор *)

```

Б.2.2 Основна програма Main



```

l_Controller (Controller)
ManipulatedValue (* Керуючий вплив *)
l_SetPoint (* Уставка *)
ProcessValue (* Дійсне значення *)

```

Б.2.3 Користувачський функціональний блок Controller

```

rtb_u_d := ABS(bi_Input) < LREAL#0.1;
rtb_Divide := bi_Input * LREAL#;

IF ANY_TO_DINT(_IC_LOADING) <> 0 THEN
    _DSTATE := rtb_Divide;
END_IF;
IF rtb_u_d OR (ANY_TO_DINT(_PrevResetState) <> 0) THEN
    _DSTATE := rtb_Divide;
END_IF;

rtb_Divide2 := ((rtb_Divide - _DSTATE) * LREAL#83.3) / LREAL#0.026;
rtb_Switch2_m := (bi_Input + rtb_Divide2) * 5.276;
rtb_LowerRelop1 := rtb_Switch2_m > LREAL#100.0;

IF rtb_Switch2_m < LREAL#-100.0 THEN
    rtb_Switch2_m := LREAL#-100.0;
END_IF;

IF rtb_LowerRelop1 THEN
    rtb_Switch2_m := LREAL#100.0;

```

```

END_IF;

rtb_Compare := NOT rtb_u_d;
IF rtb_Compare OR (ANY_TO_DINT(Integrator_PrevResetState) <> 0) THEN
    Integrator_DSTATE := LREAL#0.0;
END_IF;
rtb_u := Integrator_DSTATE;

rtb_Divide := bi_Input * LREAL#0.0276;

IF ANY_TO_DINT(_IC_LOADING_o) <> 0 THEN
    _DSTATE_l := rtb_Divide;
END_IF;
IF rtb_Compare OR (ANY_TO_DINT(_PrevResetState_m) <> 0) THEN
    _DSTATE_l := rtb_Divide;
END_IF;

rtb_Divide2_l := ((rtb_Divide - _DSTATE_l) * LREAL#80.0) / LREAL#0.0276;
rtb_Sum14_p := (bi_Input + Integrator_DSTATE) + rtb_Divide2_l;
rtb_Divide_j := rtb_Sum14_p * LREAL#30.0;

IF rtb_Divide_j < LREAL#-100.0 THEN
    rtb_Divide := LREAL#-100.0;
ELSE
    rtb_Divide := rtb_Divide_j;
END_IF;

IF rtb_Divide_j > LREAL#100.0 THEN
    rtb_Divide := LREAL#100.0;
END_IF;

rtb_LowerRelop1 := FALSE;
CASE ANY_TO_DINT(rtb_u_d) OF
    0:
        rtb_u := rtb_Switch2_m;
    1:
        rtb_LowerRelop1 := TRUE;
    ELSE
        rtb_LowerRelop1 := TRUE;
END_CASE;
IF rtb_LowerRelop1 THEN
    rtb_u := rtb_Divide;
END_IF;

bo_ManipulatedValue := rtb_u;
rtb_Divide := bi_Input / LREAL#0.1;
IF rtb_Divide_j > LREAL#100.0 THEN
    rtb_Switch2_m := rtb_Divide_j - LREAL#100.0;
ELSIF rtb_Divide_j >= LREAL#-100.0 THEN
    rtb_Switch2_m := LREAL#0.0;
ELSE
    rtb_Switch2_m := rtb_Divide_j - LREAL#-100.0;
END_IF;

IF rtb_Divide < LREAL#0.0 THEN
    rtb_u := LREAL#-1.0;
ELSIF rtb_Divide > LREAL#0.0 THEN
    rtb_u := LREAL#1.0;
ELSE
    rtb_u := rtb_Divide;
END_IF;
IF rtb_Switch2_m < LREAL#0.0 THEN
    y := LREAL#-1.0;
ELSIF rtb_Switch2_m > LREAL#0.0 THEN

```

```

    y := LREAL#1.0;
ELSE
    y := rtb_Switch2_m;
END_IF;
rtb_LowerRelop1 := y = rtb_u;

IF rtb_Sum14_p < LREAL#0.0 THEN
    rtb_Sum14_p := LREAL#-1.0;
ELSIF rtb_Sum14_p > LREAL#0.0 THEN
    rtb_Sum14_p := LREAL#1.0;
END_IF;
IF rtb_Divide_j < LREAL#0.0 THEN
    rtb_Divide_j := LREAL#-1.0;
ELSIF rtb_Divide_j > LREAL#0.0 THEN
    rtb_Divide_j := LREAL#1.0;
END_IF;
equal := rtb_Divide_j = rtb_Sum14_p;

IF rtb_Switch2_m <> 0.0 THEN
    IF rtb_LowerRelop1 THEN
        IF equal THEN
            guard_0 := TRUE;
        ELSE
            guard_2 := TRUE;
        END_IF;
    ELSE
        guard_2 := TRUE;
    END_IF;
ELSE
    guard := TRUE;
END_IF;
IF guard_2 THEN
    IF NOT rtb_LowerRelop1 THEN
        IF NOT equal THEN
            guard_0 := TRUE;
        ELSE
            guard_1 := TRUE;
        END_IF;
    ELSE
        guard_1 := TRUE;
    END_IF;
END_IF;
IF guard_1 THEN
    guard := TRUE;
END_IF;
IF guard_0 THEN
    rtb_output := LREAL#0.0;
END_IF;
IF guard THEN
    rtb_output := rtb_Divide;
END_IF;

_IC_LOADING := 0;
IF NOT rtb_u_d THEN
    _DSTATE := (LREAL#0.0001 * rtb_Divide2) + _DSTATE;
END_IF;
IF rtb_u_d THEN
    _PrevResetState := 1;
ELSE
    _PrevResetState := 0;
END_IF;

IF NOT rtb_Compare THEN
    Integrator_DSTATE := (LREAL#0.0001 * rtb_output) + Integrator_DSTATE;

```

```

END_IF;
IF rtb_Compare THEN
    Integrator_PrevResetState := 1;
ELSE
    Integrator_PrevResetState := 0;
END_IF;

_IC_LOADING_o := 0;
IF NOT rtb_Compare THEN
    _DSTATE_l := (LREAL#0.0001 * rtb_Divide2_l) + _DSTATE_l;
END_IF;
IF rtb_Compare THEN
    _PrevResetState_m := 1;
ELSE
    _PrevResetState_m := 0;
END_IF;
rtb_u_d
bi_Input (* Дійсне значення *)
rtb_Divide
_IC_LOADING
_DSTATE
_PrevResetState
rtb_Divide2
rtb_Switch2_m
rtb_LowerRelop1
rtb_Compare
Integrator_PrevResetState
Integrator_DSTATE
rtb_u
_IC_LOADING_o
_DSTATE_l
_PrevResetState_m
rtb_Divide2_l
rtb_Sum14_p
rtb_Divide_j
bo_ManipulatedValue (* Керуючий вплив *)
y
equal
guard_0
guard_2
guard
guard_1
rtb_output

```

Б.2.4 Список змінних

```

Global variables
ManipulatedValue (* Керуючий вплив *)
    REAL
    Embedded: <Symbol>
    Profile = STRATON
ProcessValue (* Дійсне значення *)
    REAL
    Embedded: <Symbol>
    Profile = STRATON
Main
l_Controller
    Controller
l_SetPoint (* Уставка *)
    REAL
Controller
bi_Input (* Дійсне значення *)
    LREAL

```

```

bo_ManipulatedValue (* Керуючий вплив *)
  LREAL
_PrevResetState_m
  SINT
_DSTATE_1
  LREAL
_IC_LOADING_o
  USINT
  := USINT#1
Integrator_PrevResetState
  SINT
Integrator_DSTATE
  LREAL
_PrevResetState
  SINT
_DSTATE
  LREAL
_IC_LOADING
  USINT
equal
  BOOL
guard
  BOOL
guard_0
  BOOL
guard_1
  BOOL
guard_2
  BOOL
rtb_Switch2_m
  LREAL
rtb_Divide
  LREAL
rtb_Divide2
  LREAL
rtb_LowerRelop1
  BOOL
rtb_u
  LREAL
rtb_Divide2_1
  LREAL
rtb_Sum14_p
  LREAL
rtb_Divide_j
  LREAL
rtb_output
  LREAL
y
  LREAL
rtb_Compare
  BOOL
rtb_u_d
  BOOL

```

ДОДАТОК В ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЛЮДИНО-МАШИННОГО ІНТЕРФЕЙСУ

В.1 Драйвера

State	Identification	Description	File name
Filter text	Filter text	Filter text	Filter text
	Driver for internal variables		Intern
	Driver for mathematics variables		MATHDR32
	Driver for system variables		SYSDRV
	Modbus RTU		MODRTU32
	VIPA		S7TCP32
	zenon Logic: Controller		STRATONNG
	zenon Logic: Harvester		STRATONNG

В.2 Список змінних

State	Name	Identification	#	Measuring unit	Dr...	Data type
Filter text	Filter text	Filter text	Filter text	Filter text	Filter text	Filter text
	Controller/Global/ManipulatedValue	Керувачий вплив	%	S...		REAL
	Controller/Global/ProcessValue	Дійсне значення	м/хв	S...		REAL
	CuttingDrive_1	Привід різання №1	!	!		BOOL
	CuttingDrive_2	Привід різання №2	!	!		BOOL
	CuttingDrivePhaseCurrentU_1	Привід різання №1 струм фази U	A	!		REAL
	CuttingDrivePhaseCurrentU_2	Привід різання №2 струм фази U	A	!		REAL
	CuttingDrivePhaseCurrentV_1	Привід різання №1 струм фази V	A	!		REAL
	CuttingDrivePhaseCurrentV_2	Привід різання №2 струм фази V	A	!		REAL
	CuttingDrivePhaseCurrentW_1	Привід різання №1 струм фази W	A	!		REAL
	CuttingDrivePhaseCurrentW_2	Привід різання №2 струм фази W	A	!		REAL
	CuttingDrivePhaseVoltageU_1	Привід різання №1 напруга фази U	V	!		REAL
	CuttingDrivePhaseVoltageU_2	Привід різання №2 напруга фази U	V	!		REAL
	CuttingDrivePhaseVoltageV_1	Привід різання №1 напруга фази V	V	!		REAL
	CuttingDrivePhaseVoltageV_2	Привід різання №2 напруга фази V	V	!		REAL
	CuttingDrivePhaseVoltageW_1	Привід різання №1 напруга фази W	V	!		REAL
	CuttingDrivePhaseVoltageW_2	Привід різання №2 напруга фази W	V	!		REAL
	CuttingDriverPower_1	Потужність споживана приводом різання №1	кВт	!		LREAL
	CuttingDriverPower_2	Потужність споживана приводом різання №2	кВт	!		LREAL
	Harvester/Global/CombineUp	Напрямок комбайна		S...		BOOL
	Harvester/Global/CombineX	Позиція комбайна		S...		REAL
	Harvester/Global/CombineY	Позиція комбайна		S...		REAL
	Harvester/Global/Lava	Значення виборки лави		S...		REAL
	Harvester/Global/LavaDown	Флаг відображення лави		S...		BOOL
	Harvester/Global/LavaUp	Флаг відображення лави		S...		BOOL
	Harvester/Global/State	Стан комбайна		S...		BOOL
	Harvester/Global/Velocitv	Швидкість комбайна	м/хв	S...		REAL
	ManipulatedValue	Керувачий вплив	%	S...		REAL
	OvervoltagePowerCircuit	Перевантаження відходящих силових мереж		!		BOOL
	ProcessValue	Дійсне значення	м/хв	S...		REAL
	ShortCircuit	Коротке замикання		!		BOOL

В.3 Шаблони зображень

State	Display...	Name	Background c...	Freel...	Left [pi...	Top [pixels]	Right [...]	Bottom [pi...
Filter text	Filter text	Filter text	Filter text	Filter text	Filter text	Filter text	Filter text	Filter text
	<input checked="" type="checkbox"/>	ALARM STATUS LINE	■ #FF0000	<input checked="" type="checkbox"/>	0	0	2560	18
	<input checked="" type="checkbox"/>	MainScreenFrame	■ #BBBBBB	<input type="checkbox"/>	0	0	1720	1080
	<input checked="" type="checkbox"/>	ButtonScreenFrame	■ #AAAAAA	<input type="checkbox"/>	1720	0	1920	1080

MainScreenFrame
Left: 0; Top: 0; Width: 1720; Height: 1080

ButtonScreenFrame
Left: 1720; Top: 0; Width: 200; Height: 1080

В.4 Функції

State	Name	Type	Parameter
Filter text	Filter text	Filter text	Filter text
	SwitchButtonScreen	Screen switch	ButtonScreen (Standard)
	SwitchMainScreen	Screen switch	MainScreen (Standard)
	SwitchMonitoringScreen	Screen switch	MonitoringScreen (Standard)
	SwitchCELScreen	Screen switch	CELScreen (Chronological Event List) - [*]...
	SwitchAMLScreen	Screen switch	AMLScreen (Alarm Message List) - [*][*]-[...]
	ExitRuntime	Exit Runtime	
	ReloadProject	Reload project online	changed objects
	SwitchTechScreen	Screen switch	TechScreen (Standard)

В.5 Зображення

State	Name	Screen type	Frame	Backgro...	Start function
Filter text	Filter text	Filter text	Filter text	Filter L...	Filter text
	AMLScreen	Alarm Messa...	MainScreenFrame	#BB	< no function linked >
	ButtonScreen	Standard	ButtonScreenFrame	#AF	< no function linked >
	CELScreen	Chronological...	MainScreenFrame	#BB	< no function linked >
	MainScreen	Standard	MainScreenFrame	#BB	SwitchButtonScreen
	MonitoringScreen	Standard	MainScreenFrame	#BB	< no function linked >
	TechScreen	Standard	MainScreenFrame	#BB	< no function linked >

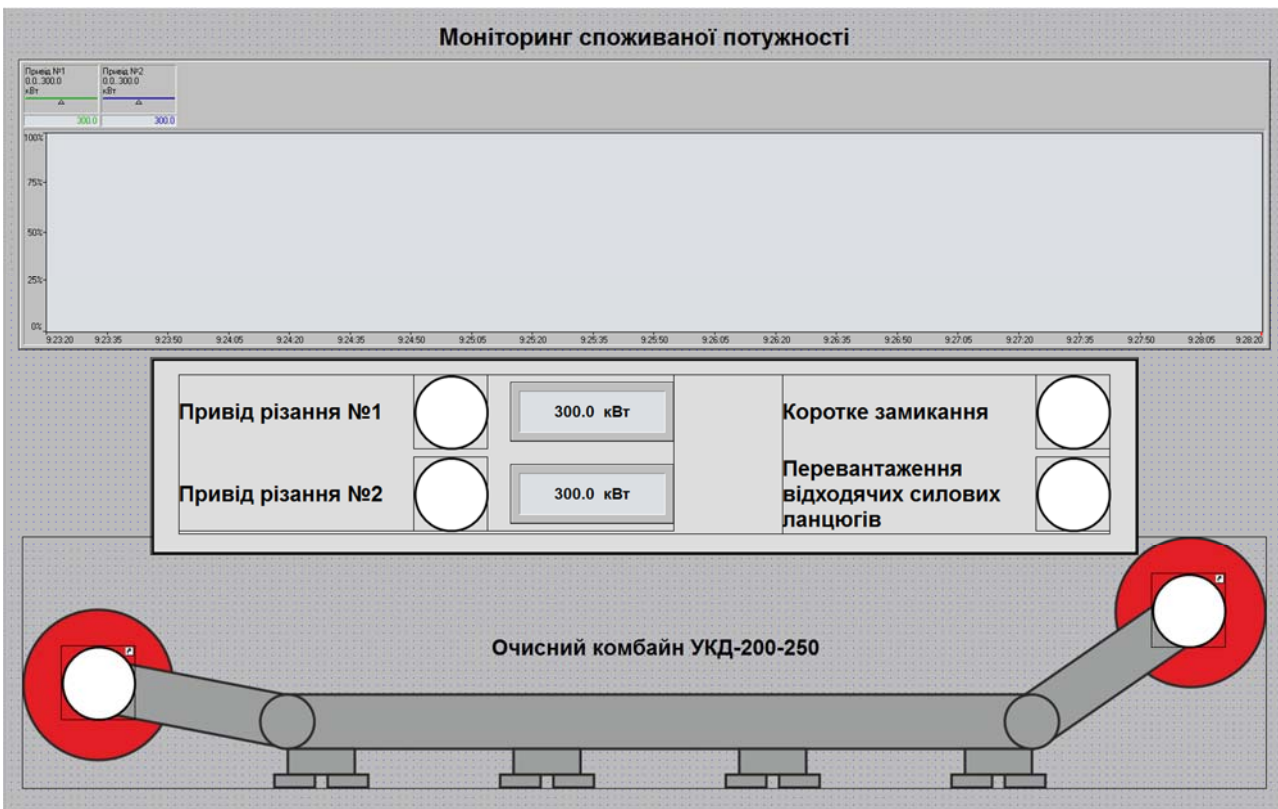
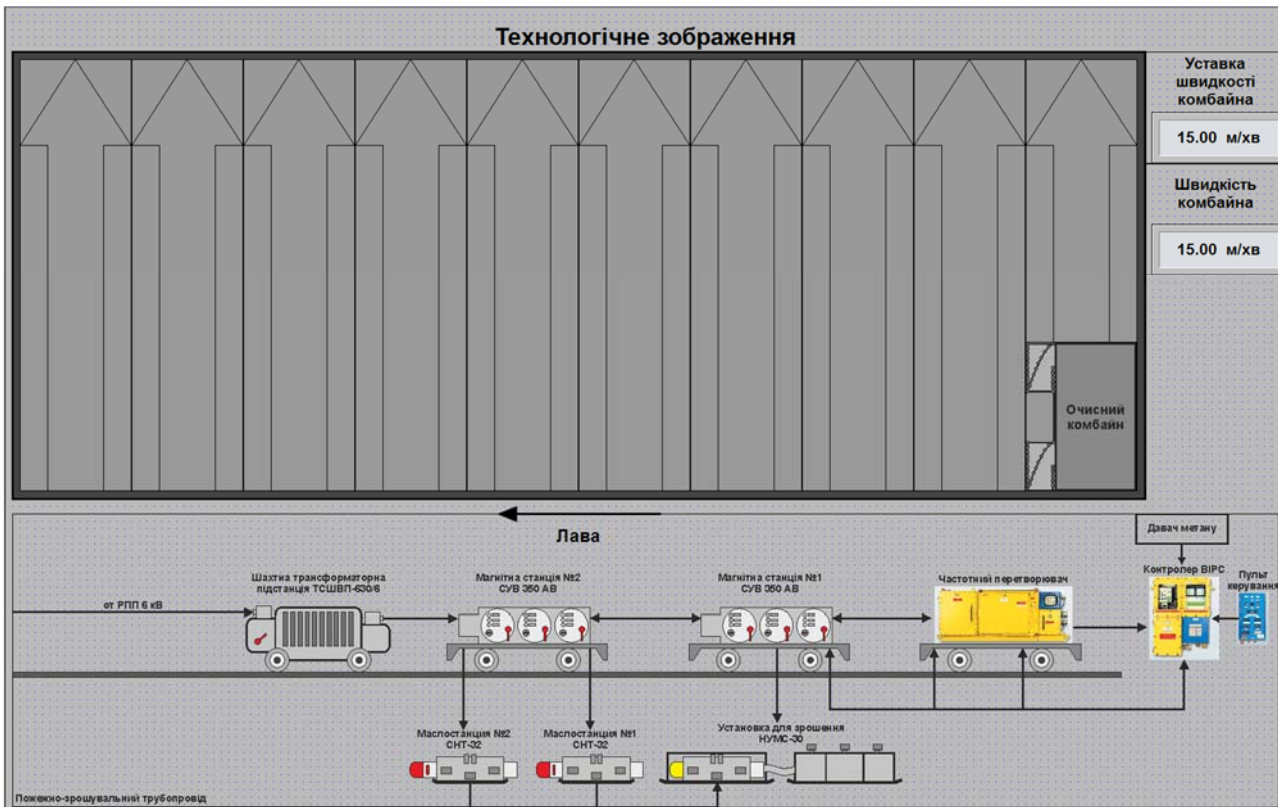


Система управління швидкістю очисного комбайна УКД-200-250

Технічні характеристики комбайна

Застосування по потужності пласта	850 - 1300 мм	в т.ч. : приводи виконавчих органів	220 кВт
Продуктивність:		привід винесеної системи подачі	2x55 кВт
при опірності вугілля різанню 120 кН / м	5,5 т/хв	Номинальна напруга	660 В
при опірності вугілля різанню 240 кН / м	4,4 т/хв	Максимальна робоча швидкість подачі	5 м/хв
при опірності вугілля різанню 360 кН / м	3,3 т/хв	Максимальна тягове зусилля системи подачі	200 кН
Сумарна номінальна потужність приводів	330 кВт	Маса	14400 кг





Хронологічний список подій

Header
Chronological Event List
Typ: SysListView32
ID: 10013

Інформаційний список тривог

Header
Alarm Message List
Typ: SysListView32
ID: 10022

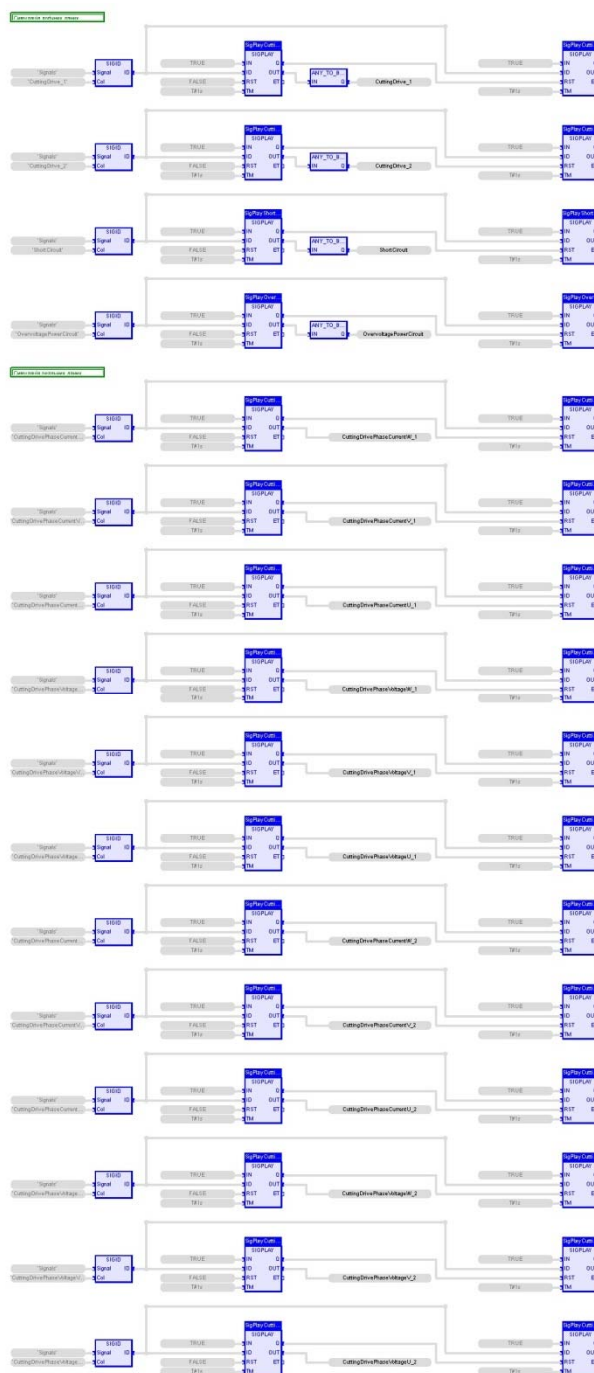
ДОДАТОК Г ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ТЕСТУВАННЯ

Г.1 Програмне забезпечення моніторингу

Г.1.1 Список програм

[FBD] SimulationObjectMonitoring (* Симуляція енергетичного стану видобучного ко мбайна *)

Г.1.2 Основна програма SimulationObjectMonitoring



SigPlayCuttingDrive_1
 CuttingDrive_1 (* Привід різання №1 *)
 SigPlayCuttingDrive_2
 CuttingDrive_2 (* Привід різання №2 *)

```

SigPlayOvervoltagePowerCircuit
OvervoltagePowerCircuit (* Перевантаження відходящих силових мереж *)
SigPlayShortCircuit
ShortCircuit (* Коротке замикання *)
SigPlayCuttingDrivePhaseCurrentW_1
CuttingDrivePhaseCurrentW_1 (* Привід різання №1 струм фази W *)
SigPlayCuttingDrivePhaseCurrentV_1
CuttingDrivePhaseCurrentV_1 (* Привід різання №1 струм фази V *)
SigPlayCuttingDrivePhaseCurrentU_1
CuttingDrivePhaseCurrentU_1 (* Привід різання №1 струм фази U *)
SigPlayCuttingDrivePhaseVoltageW_1
CuttingDrivePhaseVoltageW_1 (* Привід різання №1 напруга фази W *)
SigPlayCuttingDrivePhaseVoltageV_1
CuttingDrivePhaseVoltageV_1 (* Привід різання №1 напруга фази V *)
SigPlayCuttingDrivePhaseVoltageU_1
CuttingDrivePhaseVoltageU_1 (* Привід різання №1 напруга фази U *)
SigPlayCuttingDrivePhaseVoltageU_2
CuttingDrivePhaseVoltageU_2 (* Привід різання №2 напруга фази U *)
SigPlayCuttingDrivePhaseVoltageV_2
CuttingDrivePhaseVoltageV_2 (* Привід різання №2 напруга фази V *)
SigPlayCuttingDrivePhaseVoltageW_2
CuttingDrivePhaseVoltageW_2 (* Привід різання №2 напруга фази W *)
SigPlayCuttingDrivePhaseCurrentU_2
CuttingDrivePhaseCurrentU_2 (* Привід різання №2 струм фази U *)
SigPlayCuttingDrivePhaseCurrentV_2
CuttingDrivePhaseCurrentV_2 (* Привід різання №2 струм фази V *)
SigPlayCuttingDrivePhaseCurrentW_2
CuttingDrivePhaseCurrentW_2 (* Привід різання №2 струм фази W *)

```

Г.1.3 Список змінних

Global variables

```

CuttingDrivePhaseVoltageU_2 (* Привід різання №2 напруга фази U *)
REAL
Embedded: <Symbol>
CuttingDrivePhaseVoltageU_1 (* Привід різання №1 напруга фази U *)
REAL
Embedded: <Symbol>
CuttingDrivePhaseCurrentU_2 (* Привід різання №2 струм фази U *)
REAL
Embedded: <Symbol>
CuttingDrivePhaseCurrentU_1 (* Привід різання №1 струм фази U *)
REAL
Embedded: <Symbol>
CuttingDrive_1 (* Привід різання №1 *)
BOOL
Embedded: <Symbol>
ShortCircuit (* Коротке замикання *)
BOOL
Embedded: <Symbol>
CuttingDrivePhaseVoltageW_2 (* Привід різання №2 напруга фази W *)
REAL
Embedded: <Symbol>
CuttingDrivePhaseVoltageW_1 (* Привід різання №1 напруга фази W *)
REAL
Embedded: <Symbol>
CuttingDrivePhaseCurrentW_2 (* Привід різання №2 струм фази W *)
REAL
Embedded: <Symbol>
CuttingDrivePhaseCurrentW_1 (* Привід різання №1 струм фази W *)
REAL
Embedded: <Symbol>

```

```

CuttingDrive_2 (* Привід різання №2 *)
  BOOL
  Embedded: <Symbol>
OvervoltagePowerCircuit (* Перевантаження відходящих силових мереж *)
  BOOL
  Embedded: <Symbol>
CuttingDrivePhaseVoltageV_2 (* Привід різання №2 напруга фази V *)
  REAL
  Embedded: <Symbol>
CuttingDrivePhaseVoltageV_1 (* Привід різання №1 напруга фази V *)
  REAL
  Embedded: <Symbol>
CuttingDrivePhaseCurrentV_2 (* Привід різання №2 струм фази V *)
  REAL
  Embedded: <Symbol>
CuttingDrivePhaseCurrentV_1 (* Привід різання №1 струм фази V *)
  REAL
  Embedded: <Symbol>
SimulationObjectMonitoring
SigPlayCuttingDrive_1
  SigPlay
SigPlayCuttingDrive_2
  SigPlay
SigPlayShortCircuit
  SigPlay
SigPlayOvervoltagePowerCircuit
  SigPlay
SigPlayCuttingDrivePhaseCurrentW_1
  SigPlay
SigPlayCuttingDrivePhaseCurrentV_1
  SigPlay
SigPlayCuttingDrivePhaseCurrentU_1
  SigPlay
SigPlayCuttingDrivePhaseVoltageW_1
  SigPlay
SigPlayCuttingDrivePhaseVoltageV_1
  SigPlay
SigPlayCuttingDrivePhaseVoltageU_1
  SigPlay
SigPlayCuttingDrivePhaseCurrentW_2
  SigPlay
SigPlayCuttingDrivePhaseCurrentV_2
  SigPlay
SigPlayCuttingDrivePhaseCurrentU_2
  SigPlay
SigPlayCuttingDrivePhaseVoltageW_2
  SigPlay
SigPlayCuttingDrivePhaseVoltageV_2
  SigPlay
SigPlayCuttingDrivePhaseVoltageU_2
  SigPlay

```

Г.1.4 Сигнали

Time	CD1	CD2	SC	OP	PCW1	PCV1	PCU1	PVW1	PVV1	PVU1	PCW2	PCV2	PCU2	PVW2	PVV2	PVU2
T#0s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
T#5s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T#10s	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
T#10s	1	0	0	0	100	101	99	650	657	655	0	0	0	0	0	0
T#15s	1	0	0	0	38	39	36	650	657	655	0	0	0	0	0	0
T#20s	1	0	0	0	39	40	39	660	661	657	0	0	0	0	0	0
T#25s	1	0	0	0	53	53	53	655	657	651	0	0	0	0	0	0

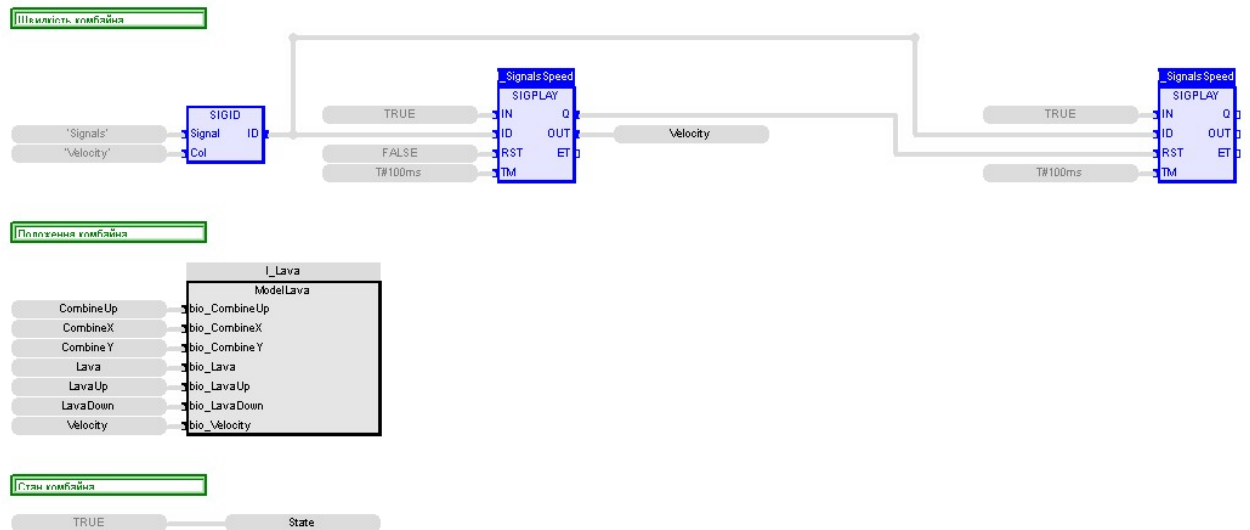
Г.2 Програмне забезпечення моніторингу

Г.2.1 Список програм

[FBD] Main (* Основна програма *)

[ST] ModellLava (* Модель лави *)

Г.2.2 Основна програма Main



l_Lava (ModellLava)

Lava (* Значення виборки лави *)

CombineUp (* Напрямок руху комбайну *)

CombineX (* Позиція комбайна *)

CombineY (* Позиція комбайна *)

Velocity (* Швидкість комбайна *)

LavaUp (* Флаг відображення лави *)

LavaDown (* Флаг відображення лави *)

State (* Стан комбайна *)

l_SignalsSpeed

Г.2.3 Користувачський функціональний блок ModellLava

```
#define d_LavaIndexMinimum 0
#define d_LavaIndexMaximum 9
#define d_LavaXMaximum 100
#define d_LavaXOne (d_LavaXMaximum / d_LavaIndexMaximum)

#define d_LavaXPixel 150.0
#define d_LavaXPixelMaximum d_LavaXPixel * ANY_TO_REAL(d_LavaIndexMaximum)
#define d_CombineXStep d_LavaXPixel * 100.0 / d_LavaXPixelMaximum
#define d_CombineXMove 0.1

#define d_LavaYPixel 580.0
#define d_CombineYPixel 200.0
#define d_CombineY d_CombineYPixel * 100.0 / d_LavaYPixel

#define d_VelocityMaximum 5.0
#define d_VelocityMaximumStep 0.05
#define d_VelocityCoefficient d_VelocityMaximumStep / d_VelocityMaximum

#define d_CombineYPixelMove 380
#define d_LavaCoefficient d_CombineYPixelMove / d_LavaYPixel

IF bio_CombineUp THEN
    bio_CombineY := bio_CombineY + bio_Velocity * d_VelocityCoefficient;
```



```

ELSE
    bio_CombineY := bio_CombineY - bio_Velocity * d_VelocityCoefficient;
END_IF;

IF bio_CombineY > 100.0 THEN
    bio_CombineY := 100.0;
ELSIF bio_CombineY < 0.0 THEN
    bio_CombineY := 0;
END_IF;

l_PositionX := ANY_TO_DINT(TRUNC(bio_CombineX)) / d_LavaXOne;
IF l_PositionX < d_LavaIndexMinimum THEN
    l_PositionX := d_LavaIndexMinimum;
ELSIF l_PositionX > d_LavaIndexMaximum THEN
    l_PositionX := d_LavaIndexMaximum;
END_IF;

FOR l_Index := d_LavaIndexMinimum TO l_PositionX DO
    bio_Lava[l_Index] := 0.0;
END_FOR;

FOR l_Index := l_PositionX TO d_LavaIndexMaximum DO
    bio_Lava[l_Index] := 100.0;
END_FOR;

IF bio_CombineUp = TRUE THEN
    bio_Lava[l_PositionX] := 100.0 - d_CombineY - bio_CombineY * d_LavaCoefficient;
    FOR l_Index := d_LavaIndexMinimum TO d_LavaIndexMaximum DO
        bio_LavaUp[l_Index] := TRUE;
        bio_LavaDown[l_Index] := FALSE;
    END_FOR;
ELSE
    bio_Lava[l_PositionX] := 100.0 - d_CombineY - (100.0 * d_LavaCoefficient - bio_CombineY * d_LavaCoefficient);
    FOR l_Index := d_LavaIndexMinimum TO d_LavaIndexMaximum DO
        bio_LavaUp[l_Index] := FALSE;
        bio_LavaDown[l_Index] := TRUE;
    END_FOR;
END_IF;

IF (bio_CombineUp = TRUE AND bio_CombineY = 100.0) OR (bio_CombineUp = FALSE AND bio_CombineY = 0.0) THEN
    bio_Velocity := 0.0;
    bio_CombineX := bio_CombineX + d_CombineXMove;
END_IF;

IF (ANY_TO_DINT(TRUNC(bio_CombineX)) / d_LavaXOne) > l_PositionX THEN
    bio_CombineX := bio_CombineX + d_CombineXMove;
    bio_CombineUp := NOT(bio_CombineUp);
END_IF;

bio_CombineUp (* Напряг руху комбайну *)
bio_CombineY (* Положення комбайна в лаві по осі Y *)
bio_Velocity (* Швидкість комбайна *)
l_PositionX (* Положення за осю X *)
bio_CombineX (* Положення комбайна в лаві по осі X *)
l_Index
bio_Lava (* Стан лави *)
bio_LavaUp (* Флаг відображення лави *)
bio_LavaDown (* Флаг відображення лави *)

```

Г.2.4 Список змінних

Global variables

```

State (* Стан комбайна *)
  BOOL
  Embedded: <Symbol>
  Profile = STRATON
CombineY (* Позиція комбайна *)
  REAL
  Embedded: <Symbol>
  Profile = STRATON
CombineX (* Позиція комбайна *)
  REAL
  Embedded: <Symbol>
  Profile = STRATON
Velocity (* Швидкість комбайна *)
  REAL
  Embedded: <Symbol>
  Profile = STRATON
Lava (* Значення виборки лави *)
  REAL [0..9]
  Embedded: <Symbol>
  Profile = STRATON
CombineUp (* Напрямок руху комбайну *)
  BOOL
  := TRUE
  Embedded: <Symbol>
  Profile = STRATON
LavaDown (* Флаг відображення лави *)
  BOOL [0..9]
  Embedded: <Symbol>
  Profile = STRATON
LavaUp (* Флаг відображення лави *)
  BOOL [0..9]
  Embedded: <Symbol>
  Profile = STRATON
Main
l_Lava
  _Modellava
l_SignalsSpeed
  SigPlay
Modellava
bio_CombineUp (* Напрямок руху комбайну *)
  BOOL
bio_CombineX (* Положення комбайна в лаві по осі X *)
  REAL
bio_CombineY (* Положення комбайна в лаві по осі Y *)
  REAL
bio_Lava (* Стан лави *)
  REAL [0..9]
bio_LavaUp (* Флаг відображення лави *)
  BOOL [0..9]
bio_LavaDown (* Флаг відображення лави *)
  BOOL [0..9]
bio_Velocity (* Швидкість комбайна *)
  REAL
l_Index
  DINT
l_PositionX (* Положення за осю X *)
  DINT

```

Г.2.5 Сигнали

Time	Velocity
T#0s	0
T#10s	0
T#10s	1.5
T#20s	1.5
T#20s	2.5
T#30s	2.5
T#30s	3.5
T#50s	3.5
T#50s	5.0
T#80s	5.0
T#80s	2.0
T#90s	2.0