

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

Пояснювальна записка

до дипломної роботи

магістра

(освітньо-кваліфікаційний рівень)

на тему «Розробка і дослідження системи управління стрічковим
транспортером з розпізнаванням маркування об'єктів методами технічного
зору»

ХАІ.301.362а.22О.151.00 9652039 ПЗ

Виконав: студент 2 курсу, групи 362а

Галузь знань 15 «Автоматизація та
приладобудування»

Спеціальність

151 “Автоматизація та комп’ютерно-
інтегровані технології”

Освітня програма “Комп’ютерні системи
технічного зору”

Скрипниченко М.В.

(прізвище та ініціали студента)

Керівник доц. Джуглаков В.Г.

(прізвище та ініціали)

Рецензент Суліма Є.А.

(прізвище та ініціали)

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів
Кафедра систем управління літальних апаратів (№301)
Рівень вищої освіти магістр
Галузь знань 15 «Автоматизація та приладобудування»
(шифр і назва)
Спеціальність 151 «Автоматизація та комп'ютерно- інтегровані технології»
(шифр і назва)
Освітня програма «Комп'ютерні системи технічного зору»
(назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри

к.т.н., доц. Костянтин ДЕРГАЧОВ

« » 2022 року

ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Скрипниченко Микиті Валентиновичу
(прізвище, ім'я, по батькові)

1. Тема роботи «Розробка і дослідження системи управління стрічковим транспортером з розпізнаванням маркування об'єктів методами технічного зору»

керівник роботи доцент Джулгаков В.Г.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від 17.11. 2022 року № 1602-уч

2. Строк подання студентом роботи: 09 грудня 2022 року

3. Вихідні дані до роботи Опис стенду «Стрічковий транспортер», вимоги до системи

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Вступ, оцінка проблеми управління стрічковим транспортером, опис об'єкту автоматичного управління, математичний опис об'єкту автоматичного управління, синтез системи автоматичного управління, конструкторська частина, дослідницька частина, дослідницько-експериментальна частина, економічна частина, заключення, список використаних джерел

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) 10 слайдів: опис об'єкту автоматичного управління, дослідження ОАУ, формування структури та дослідження САУ, формування коригувального пристрою, дослідження скоригованої системи, Вплив елементів нелінійності та розрахунок дискретного регулятора, конструкторська частина, розробка системи класифікації знаків маркування, напівнатурне моделювання, економічна частина.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	доцент Джулгаков В.Г.	19.09.2022	09.12.2022
2	доцент Джулгаков В.Г.	19.09.2022	09.12.2022
3	доцент Джулгаков В.Г.	19.09.2022	09.12.2022
4	к.т.н. доцент Краснов Л.О.	19.09.2022	09.12.2022
5	к.т.н. доцент Краснов Л.О.	19.09.2022	09.12.2022
6	к.е.н. доцент Попов О.С.	19.09.2022	09.12.2022

Нормоконтроль _____ Джулгаков В.Г. « 16 » грудня__2022 р.
 (підпис) (ініціали та прізвище)

7. Дата видачі завдання _____ 19.09.2022 _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1.	Початок переддипломної практики	12.09.2022	
2.	Формулювання теми роботи. Розробка технічного завдання	19.09.2021	
3.	Математичний опис системи управління. Аналіз і синтез системи управління. Проведення експериментальних досліджень	28.10.2022	Залік з переддипломної практики
4.	Конструкторська частина роботи. Дослідницька частина роботи. Експериментально-практична частина. Економічне обґрунтування розробки. Розробка питань охорони праці і безпеки в надзвичайних ситуаціях	28.11.2022	
5.	Оформлення розрахунково-пояснювальної записки і графічного матеріалу	09.12.2022	
6.	Попередній захист роботи. Рецензування роботи	16.12.2022	
7.	Захист роботи	19.12.2022	

Студент _____ Скрипниченко М.В.
 (підпис) (ім'я, прізвище.)

Керівник роботи _____ Джулгаков В.Г.
 (підпис) (ім'я, прізвище.)

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра систем управління літальних апаратів (№301)

«ЗАТВЕРДЖУЮ»

Завідуючий кафедрою 301

к.т.н., с.н.с., доцент

_____ Костянтин ДЕРГАЧОВ

«__» _____ 2022 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на дипломне проектування

Скрипниченко Микити Валентиновича

1 Тема роботи: «Розробка і дослідження системи управління стрічковим транспортером з розпізнаванням маркування об'єктів методами технічного зору»

затверджена наказом по університету від «17» листопада 2022 р. № 1602-уч _____.

2 Строк здачі студентом закінченої роботи « 09 » _____ грудня _____ 2022 р.

3 Область застосування розробки: Промислові стрічкові транспортери

4 Початкові дані для розроблювальної системи

4.1 Призначення і мета створення системи: Управління стрічковим транспортером з розпізнаванням маркування засобами технічного зору

4.2 Загальні відомості Система управління має бути розроблена для лабораторного макету стрічкового транспортера. Діапазон швидкості руху стрічки - від 2 до 50 мм/с.

У складі стенду реалізовані механізми переміщення відеокамери паралельно з рухом

стрічки. Завданням локального контролера є стабілізація швидкості руху стрічки при навантаженнях і синхронного руху відеокамер. Завданням комп'ютера (промислового контролера) є аналіз зображень та ідентифікація маркування

5 Технічні вимоги до каналів системи управління

5.1 Питання, що підлягають розробці: формування математичної моделі ОАУ, вибір виконавчого пристрою, розробка функціональної і структурної схем САУ, синтез ПАУ, аналіз якості управління, формування контролера, розробка системи розпізнавання маркування, напівнатурне моделювання роботи системи

5.2 Режим роботи системи (безперервний, циклічний, одноразової дії):
безперервний

5.3 Показники якості системи управління: допустима стала помилка (0.1 %), час перехідного процесу (до 0.5 секунд), перерегулювання (до 5 %), запаси стійкості по фазі (не менше 40 градусів) та по амплітуді (не менше 10 дБ)

5.4 Вимоги до приладового складу системи: контролер AT89S8252, оптичні енкодери

5.5 Вимоги до взаємозамінності блоків: відсутні

6 Умови експлуатації системи

6.1 Кліматичні вимоги до експлуатації (температура середовища, у якій буде працювати система управління, її вологість, вміст хімічно активних компонентів і т.ін.)

а) вологість 35%

б) температура середовища +5.. +40 градусів за Цельсієм

в) хімічно активні компоненти відсутні

6.2 Механічні вимоги (вібрація, тряска, можливі перекося, удари, нахили і т.ін.):
можливі вібрація, тряска

6.3 Наявність перешкод (електричні наведення радіоперешкоди, магнітні впливи):
відсутні

6.4 Електричні параметри системи (напруга джерел живлення, потужність, стабільність, частота): напруга живлення 25 В , потужність 20 Вт, постійний струм

7 Додаткові функції, реалізовані системою (сигналізація про несправності, реєстрація необхідної інформації, самоконтроль самої системи і т.ін.):

Дана система не має додаткових функцій

8 Обсяг виконуваних розроблювачем робіт

8.1 Етапи проведення роботи: Вербальний та математичний опис об'єкту управління. II етап: Розробка системи управління та розрахунок регулюючого пристрою . III етап: Розробка системи класифікації маркування з використанням засобів технічного зору. IV етап : Експериментальне дослідження САУ

8.2 Обсяг розробки по кожному етапу: _____

1) Стан проблеми (3 ст)

2) Синтез системи автоматичного управління (46 ст)

3) Конструкторська частина (4 ст)

4) Дослідницька частина (10 ст)

5) Експериментально-дослідницька частина (4 ст)

6) Економічна частина (11 ст)

9 Параметри устаткування системи: визначаються в процесі проектування

10 Вимоги безпеки: визначаються в процесі проектування

11 Дослідницька частина: Дослідження засобів розпізнавання та класифікації маркування об'єктів з використанням засобів технічного зору

12 Експериментально-практична частина: Напівнатурне моделювання роботи системи автоматичного управління з розпізнаванням маркування об'єктів

13 Економічна частина

13.1 Розробити (розрахувати, одержати): розрахувати собівартість системи управління

13.2 Умови і вимоги: одиничне виробництво, розрахунок проводиться за статтями калькуляції

13.3 Очікуваний результат: повна собівартість виробу грн. з урахуванням усіх витрат

14 Перелік графічних матеріалів із зазначенням форматів: 10 слайдів формату А4: опис об'єкту автоматичного управління, дослідження ОАУ, формування структури та дослідження САУ, формування коригувального пристрою, дослідження скоригованої системи, Вплив елементів нелінійності та розрахунок дискретного регулятора, конструкторська частина, розробка системи класифікації знаків маркування, напівнатурне моделювання, економічна частина.

Керівник роботи

Джулгаков Віталій Георгійович
(П.І.Б.)

« » _____ 2022 р.

Прийняв до виконання

Скрипниченко Микита Валентинович
(П.І.Б. студента)

« » _____ 2022 р.

Погоджено з питань:

конструкції

Джулгаков Віталій Георгійович
(П.І.Б.)

« » _____ 2022 р.

дослідницької частини

Краснов Леонід Олександрович
(П.І.Б.)

« » _____ 2022 р.

економіки

Попов Олександр Сергійович
(П.І.Б.)

« » _____ 2022 р.

РЕФЕРАТ

Пояснювальна записка містить: 107 ст., 48 рис., 27 табл., 3 додатків, 11 джерел.

Тема роботи: розробка і дослідження системи управління стрічковим транспортером з розпізнаванням маркування об'єктів методами технічного зору.

Мета роботи: розробка системи управління стрічковим транспортером з розпізнаванням маркування об'єктів методами технічного зору.

Предмет роботи: система управління стрічковим транспортером.

У дипломній роботі було розроблено систему управління стрічковим транспортером.

Результатом роботи є розроблена система управління стрічковим транспортером.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ

САУ – система автоматичного управління;

ОАУ – об'єкт автоматичного управління;

ТЗ – технічний зір;

ЛАЧХ - логарифмічно амплітудно частотна характеристика;

ПАУ – пристрій автоматичного управління;

ПЕОМ – персональна електронно обчислювальна машина;

ТБ – точка безбитковості.

ЗМІСТ

ВСТУП	12
1 СТАН ПРОБЛЕМИ	13
1.1 Оцінка проблеми використання стрічкових транспортерів.....	13
1.2 Огляд джерел	14
1.3 Патентний аналіз	15
1.4 Постановка задачі проектування	15
2 СИНТЕЗ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ	16
2.1 Опис функціонування ОАУ	16
2.2 Функціональна схема ОАУ	18
2.3 Формування вимог та вибір контролеру.....	19
2.4 Формування математичної моделі контуру управління швидкістю стрічки транспортеру.....	22
2.5 Лінеаризація математичної моделі та визначення передавальних функцій контуру управління швидкістю стрічки транспортеру	25
2.6 Машинне моделювання контуру управління швидкістю стрічки	28
2.7 Формування математичної моделі контуру керування швидкістю камери	30
2.8 Лінеаризація математичної моделі контуру керування камерою	32
2.9 Машинне моделювання контуру керування швидкістю камери	34
2.10 Постановка задачі синтезу	36
2.11 Дослідження характеристик нескоригованої САУ	40
2.12 Синтез коригуючого пристрою.....	43
2.13 Дослідження скоригованої САУ	51
2.14 Дослідження САУ з елементами нелінійності	56
2.15 Синтез цифрового коригуючого пристрою САУ	58
2.16 Висновки	62
3 КОНСТРУКТОРСЬКА ЧАСТИНА	63
3.1 Задачі контролеру і визначення складу вхідної інформації	63
3.2 Розробка алгоритмічного забезпечення.....	63

3.4 Висновки	65
4 ДОСЛІДНИЦЬКА ЧАСТИНА	66
4.1 Задачі і методи проведення досліджень	66
4.2 Дослідження засобів визначення положення знаку на зображенні	67
4.2.1 Дослідження методу виділення контурів на зображенні	68
4.2.2 Перевірка контуру на округлість.....	69
4.2.3 Класифікація знаку.....	70
4.3 Результати досліджень.....	72
6.5 Висновки	74
5 ЕКСПЕРИМЕНТАЛЬНО-ДОСЛІДНИЦЬКА ЧАСТИНА	75
5.1 Задачі і засоби виконання експериментальної розробки	75
5.2 Опис лабораторної установки.....	75
5.3 Аналіз результатів проведення експериментів і впровадження розробки	76
5.5 Висновки	78
6 ЕКОНОМІЧНА ЧАСТИНА	79
6.1 Основні поняття і положення	79
6.2 Опис виробу.....	79
6.3 Сегментування ринку.....	79
6.4 Аналіз конкурентоспроможності.....	80
6.5 Розрахунок собівартості і ціни виробу	83
6.6 Розрахунок точки беззбитковості.....	88
8.7 Висновки	89
ЗАКЛЮЧЕННЯ.....	90
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	91
ДОДАТОК А.....	93
ДОДАТОК В.....	99

ВСТУП

У сучасному світі поступово збільшуються об'єми споживацької продукції, які, перед тим як потрапити до рук людини, проходять низку технологічних процесів, починаючи від перетворення сировини на найпростіші складові компоненти, наприклад радіоелементи, потім складання та компоновка товару, упаковка, доставка. Між всіма етапами виробництва продукту, проходить переміщення об'єктів, між глобальними етапами, такими як виробництво складових вузлів на одному підприємстві та компоновка продукту на іншому, логістична задача вирішується різними видами транспорту, на локальному рівні, такому як сортування посилок у відділенні пошти, чи переміщення продуктів між різними етапами технологічних процесів виготовлення, існує декілька варіантів вирішення проблеми, починаючи від ручної переноски співробітниками, що займає багато часу, має більшу вірогідність пошкодити об'єкти транспортування та витрачає оплачувані людино-години, і закінчуючи системою дронів, що транспортує об'єкти між різними точками уникаючи різноманітних перешкод, проте є дорогою та складною у налаштуванні.

Одним з засобів вирішення задачі переміщення об'єктів на виробництві є стрічковий транспортер, до його переваг можна віднести відносну дешевизну у порівнянні з аналогами та простоту налаштування й експлуатації системи. Проте ручне управління таким транспортером через людський фактор є малоефективним, тому стрічкові транспортери доповнюються системами автоматичного управління що дозволяють їм транспортувати вантаж з постійною, відповідною до заданого режиму швидкістю. У випадках коли одна система транспортерів використовується для різних видів об'єктів які потребують виконання різних дій з ними або розгалуження маршруту доставки необхідно класифікувати кожен об'єкт транспортування, одним з методів класифікації є використання засобів технічного зору, до переваг цього методу можна віднести швидкість класифікації, мінімальну необхідну взаємодію з вантажем та простоту доповнення транспортеру, оскільки достатньо установити камеру навпроти. У деяких випадках це може доповнюватися необхідністю слідування камери за вантажем.

У даній роботі буде розглядатися розробка системи автоматичного управління стрічковим транспортером що забезпечуватиме стабілізацію швидкості руху стрічки та слідування камери за вантажем, та доповнення її системою класифікації вантажу за знаками, зображеними на ньому.

1 СТАН ПРОБЛЕМИ

1.1 Оцінка проблеми використання стрічкових транспортерів

Основна функція стрічкового транспортеру полягає у переміщенні об'єктів уздовж стрічки, і хоча у найпростішому варіанті використання для виконання цієї функції відсутня необхідність керування транспортером, та достатньо подати живлення на привід обертового валу, стрічковий транспортер розповсюджений у різноманітних промислових системах де постійно необхідно контролювати швидкість переміщення вантажу, змінювати її та компенсувати зовнішні збурюючі впливи, оскільки навіть незначне відхилення від заданого алгоритму дії може призвести до значних проблем та відхилень у цілому алгоритмі виробництва та непридатності для використання цілих партій продукту.

Задача управління полягає у забезпеченні зміни керованої величини відповідно до заданого значення керуючого впливу, дотримуючись заданих вимог до якості функціонування системи. Керування стрічковим транспортером полягає у зміні лінійної швидкості стрічки, на якій знаходяться предмети транспортування, зміна швидкості стрічки забезпечується зміною швидкості обертання валу електродвигуну, що обертає один з валів транспортеру, електродвигун керується сигналом що створюється системою управління. Існують три фундаментальні підходи до реалізації системи управління:

1) за керуючим впливом – підхід полягає у корекції задаючого впливу, для впливу на об'єкт автоматичного управління (ОАУ), при якому вихідний сигнал буде відповідати заданим вимогам. Недоліком даного методу є неможливість компенсації збурюючих впливів;

2) за збурюючим впливом – підхід полягає у компенсації збурення, що надходить на ОАУ, шляхом виміру збурюючого впливу та відповідної корекції керуючого впливу. Недоліком даного методу є відсутність корекції керуючого впливу для отримання необхідного вихідного сигналу;

3) за відхиленням – підхід полягає у подачі на ОАУ сигналу відхилення, який являє собою різницю між керуючим впливом та поточним вихідним сигналом ОАУ, що забезпечує як відповідність вихідного сигналу керуючому так і компенсацію відхилення.

Для реалізації системи управління спираючись на принцип управління за відхиленням необхідно отримувати інформацію про наявне значення

керованої величини, яке можна отримувати, обробляючи вихідний сигнал з оптичного енкодеру що закріплено на вільному валу транспортеру, який приводиться до обертання лише полотном стрічки. Для вирішення задачі управління обрано саме принцип управління за відхиленням оскільки, хоча середовище на виробництві гарно контрольоване та дозволяє уникнути більшості можливих збурюючих впливів, сила ваги предмету, що рухає транспортер впливатиме на швидкість лінійного руху стрічки.

Інша проблема у промисловому використанні полягає у ідентифікації об'єктів що рухає транспортер, ідентифікація необхідна оскільки один транспортер може використовуватися для переміщення різних типів предметів, для яких може бути необхідним використовувати сортування, чи виконання різних операцій з необхідністю, наприклад, зміни швидкості транспортеру. У якості одного з методів ідентифікації можна виділити технічний зір, оскільки він вимагає мінімуму взаємодії з об'єктом ідентифікації, та при використанні його у промисловості достатньо просто забезпечити постійне освітлення, ракурс на об'єкт, та відсутність різноманітних перешкод що заважають, та погіршують якість використання систем технічного зору на вулиці, також у цих умовах відсутнє значне обмеження по живленню, як, наприклад, у мобільних роботах, що дозволить забезпечити більші обчислювальні потужності, для швидшої ідентифікації.

У випадку коли об'єкти ідентифікації суттєво відрізняються достатньо спиратися на їх силует та колір, проте введення умовних позначень нанесених на об'єкти зробить створювану систему ідентифікації більш універсальною, та спростить внесення до неї нових об'єктів, для яких не потрібно буде проводити дослідження розпізнавання їх форми та кольору, а просто внести до системи нове умовне позначення.

1.2 Огляд джерел

Враховуючи широку розповсюдженість різноманітних видів транспортерів, вирішувана у роботі проблема не є унікальною, тому з метою спрощення вирішення поставленої задачі проведено аналіз існуючих рішень у технічній літературі. Розглянута література наведена у додатку А.

У роботі [4] розглянуті методи синтезу систем автоматичного управління, у роботі [5] розглянуті засоби та шляхи моделювання САУ й ОАУ, у роботі [6] розглянуті засоби проектування систем управління та засоби визначення

маркування методами технічного зору, у роботі [7] розглянута структура стрічкових транспортерів та засоби керування.

1.3 Патентний аналіз

Ознайомлення з існуючими патентами систем управління транспортерів допоможе краще зрозуміти поставлену задачу та дізнатися про додаткові проблеми, що можуть виникнути у процесі проектування. Результати патентного пошуку представлені у додатку Б.

У роботі [8] розглянута розроблена система управління стрічковим транспортером відмінної конструкції, у роботі [9] розглянуто розроблену систему управління накопичувальним транспортером, у роботі [10] розглянуті засоби аналізу та ідентифікації властивостей упаковки з використанням методів технічного зору у промисловій робототехніці.

1.4 Постановка задачі проектування

Задача проектування полягає у розробці системи автоматичного управління лінійною швидкістю руху стрічки стрічкового транспортера, що відповідає наступним показникам якості функціонування системи:

- 1) $t_{\text{пп}} \leq 0.5\text{с}$ – час перехідного процесу;
- 2) $\sigma \leq 5\%$ - перерегулювання;
- 3) $\varepsilon_{\text{ст}} \leq 0.1\%$ - стала похибка;
- 4) $\varphi_3 \geq 40^\circ$ - запас стійкості за фазою;
- 5) $L_3 \geq 10\text{дБ}$ – запас стійкості за амплітудою.

У якості збурюючого впливу у системі виступає вплив сили ваги предмету, що рухається транспортером, на натяг стрічки транспортеру.

Система доповнюється програмним додатком для класифікації маркування об'єктів транспортування, виходячи з класифікації

2 СИНТЕЗ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ

2.1 Опис функціонування ОАУ

Стрічкові транспортери широко розповсюджені у виробництвах та системах зберігання, через можливість постійного переміщення вантажів у суттєвих кількостях, проте їх використання обмежується тим що транспортери доволі громіздкі та стаціонарні. У загальному вигляді стрічковий транспортер складається з приводного барабану, до якого приєднано привід, який може бути електродвигуном під'єднаним до барабану напряду чи через редуктор, в залежності від характеристик двигуна та вимог до транспортеру, пасивного валу, який знаходиться на протилежному кінці транспортеру від приводного валу, роликів транспортеру, які підтримують стрічку при значній довжині транспортеру, та стрічки, яка може бути виготовлена з різних матеріалів та мати різну структуру в залежності від розміру, ваги та типу вантажу що необхідно транспортувати.

У роботі буде проводитися синтез систему автоматичного управління для стенду “Стрічковий транспортер”, що імітує роботу типового транспортного механізму протягування стрічки. Стенд складається з двох валів зі стрічкою, один вал приводиться до руху електродвигуном з редуктором, інший – пасивний. На обох валах встановлені оптичні енкодери, що забезпечують інформацію про обертання валів, та, відповідно, рух стрічки, також на пасивному валу встановлено тахогенератор, який повертає постійну напругу, пропорційно до швидкості обертання валу. Електродвигун що приводить до руху стрічку керується постійним струмом у діапазоні 0-27В, що подається з підсилювача потужності. Паралельно до стрічки транспортера встановлено пілон, з рухомою відеокамерою, до руху камера приводиться окремим електродвигуном постійного струму з редуктором, який керується постійним струмом напругою $\pm 27\text{В}$. Відеокамера з'єднана з ПЕОМ через інтерфейс USB для подальшої обробки сигналу з камери засобами технічного зору.

На платформі встановлено блок узгоджувальних підсилювачів (БУП), який необхідний для узгодження сигналів з датчиків, до необхідного мікроконтролеру формату, та об'єднує порти усіх вхідних та вихідних сигналів до одного блоку. Компонувочна схема стенду та його вхідні й вихідні сигнали наведені на рис. 2.1.

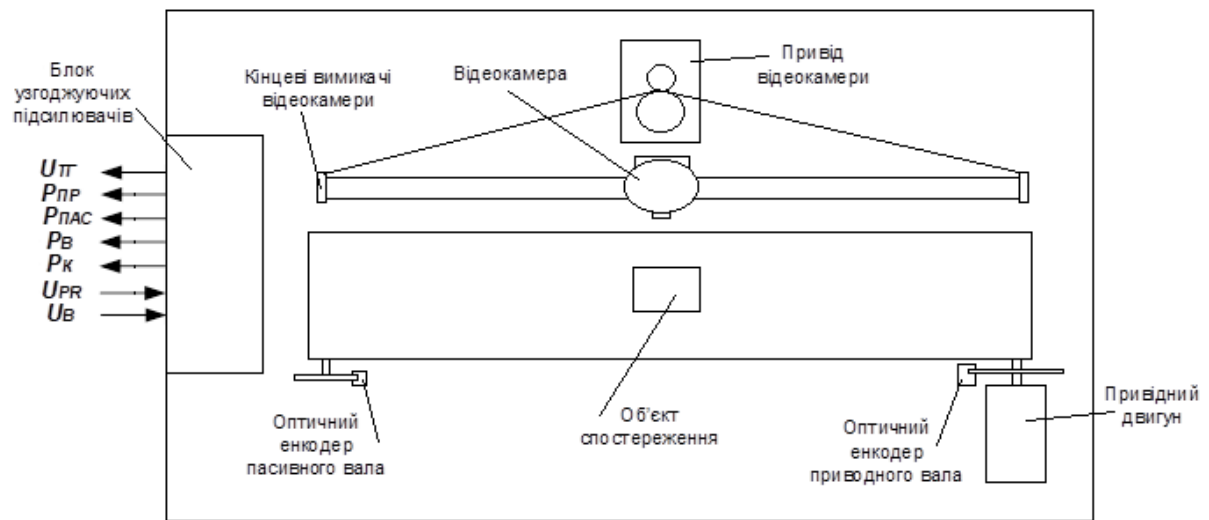


Рисунок 2.1 – Компонувочна схема ОАУ

На рис. 2.1 прийняті наступні умовні позначення:

$U_{ТГ}$ – Напряга постійного струму тахогенератора, пропорційна швидкості обертання пасивного вала. Аналоговий сигнал у діапазоні 0..5В;

$P_{пр}$ – Імпульси від оптичного енкадера на привідному валу. Потенціальні імпульси (логічний 0 – 0 В, логічна 1 – +5 В);

$P_{пас}$ – Імпульси від оптичного енкадера на пасивному валу, Потенціальні імпульси (логічний 0 – 0 В, логічна 1 – +5 В);

$P_{в}$ – Імпульси від оптичного енкадера, пропорційні швидкості руху відеокамери . Потенціальні імпульси (логічний 0 – 0 В, логічна 1 – +5 В);

$P_{к}$ – Імпульс при досягненні відеокамерою граничного положення. Імпульс (логічний 0 – 0 В, логічна 1 – +5 В);

$U_{пр}$ – Сигнал керування приводом транспортеру. Аналоговий сигнал в дьяпазоні 0..5В;

$U_{в}$ – Сигнал керування приводом відеокамери. Аналоговий сигнал у дьяпазоні -5..5В.

Обробка сигналів зворотнього зв'язку та генерація керуючих сигналів, на основі сигналів з датчиків, буде виконуватися на мікроконтролерному модулі, що надаватиме можливість зчитувати імпульси енкадерів та кінцевих датчиків камери завдяки зовнішнім перериванням, зчитувати сигнал з тахогенератору завдяки модулю АЦП та створювати керуючий сигнал для двигунів завдяки модулям ЦАП. В свою чергу на мікроконтролерний модуль буде передаватися задана швидкість стрічки транспортеру. Обробка сигналів з камери засобами

технічного зору виконується на ПЕОМ яка й передає на мікроконтролерний модуль задане значення швидкості стрічки транспортеру.

2.2 Функціональна схема ОАУ

Для подальшої розробки САУ створена функціональна схема ОАУ з мікроконтролерним модулем, створена схема наведена на рис. 2.2.

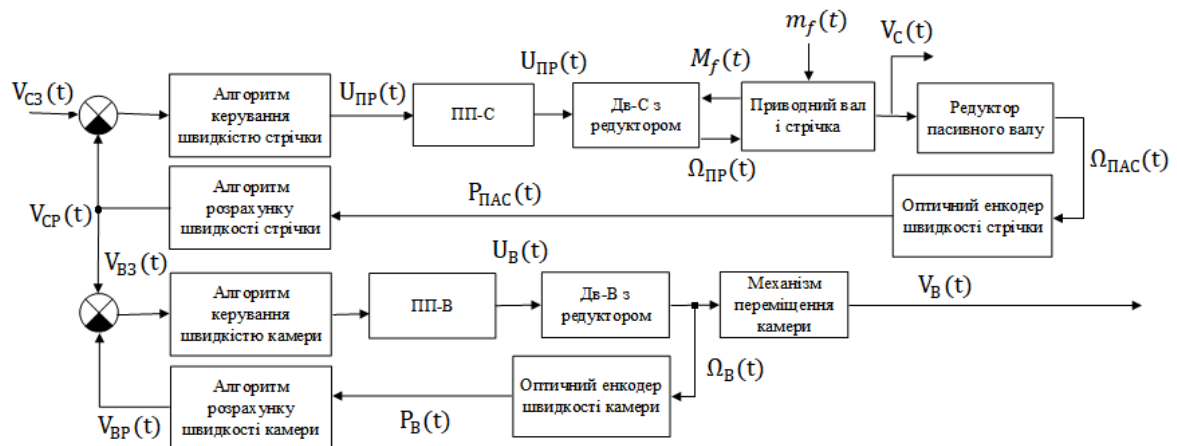


Рисунок 2.2 – Функціональна схема ОАУ з мікроконтролерним модулем

На рис. 2.2 прийняті наступні позначення:

$V_{C3}(t)$ – задана швидкість стрічки, В;

$V_C(t)$ – фактична швидкість стрічки, В;

$V_{CP}(t)$ – розрахована швидкість стрічки, В;

$\Omega_{PP}(t)$ – кутова швидкість приводу стрічки, $\frac{\text{об}}{\text{с}}$;

$\Omega_{ПАС}(t)$ – кутова швидкість пасивного вала, $\frac{\text{об}}{\text{с}}$;

Дв-С – двигун привода стрічки;

$V_{B3}(t)$ – задана швидкість руху відеокамери, В;

$V_B(t)$ – фактична швидкість руху відеокамери, В;

$V_{BP}(t)$ – розрахована швидкість руху відеокамери, В;

$\Omega_B(t)$ – кутова швидкість привода відеокамери, $\frac{\text{об}}{\text{с}}$;

Дв-В – двигун привода відеокамери;

$R_{ПАС}(t)$ – імпульси від енкодера на пасивному валу, В;

$R_B(t)$ – імпульси від енкодера на приводному валу відеокамери, В;

$U_{PP}(t)$ – сигнал керування приводом стрічки, В;

$U_B(t)$ – сигнал керування приводом відеокамери, В;

$m_f(t)$ – збурюючий вплив ваги вантажу, кг;

$M_f(t)$ – Збурюючий момент на палу двигуна.

2.3 Формування вимог та вибір контролеру

Оскільки ОАУ вже має в своєму складі вимірювальні пристрої що надають достатню, для виконання поставленої задачі управління, інформацію про функціонування об'єкту, необхідно обрати у якості виконавчого пристрою мікроконтроллер, у складі якого є достатня кількість вхідних та вихідних портів для вхідних та вихідних сигналів, а саме 2 виходи ЦАП для керування приводами ОАУ, 1 вхід АЦП для зчитування сигналу з тахогенератору та 2 обробчики зовнішніх переривань для зчитування логічних імпульсів з енкодерів. Також оскільки до системи висунута вимога про необхідний час перехідного процесу $t_{\text{пн}} \leq 1\text{с}$ необхідно обрати виконавчий пристрій швидкодія якого значно перевищує вимогу до системи. Виходячи з поставлених умов обрано декілька варіантів виконавчих пристроїв, серед яких обрано найкращий методом вагових коефіцієнтів, варіанти виконавчих пристроїв наведені у табл. 2.1.

У доступному ціновому діапазоні мікроконтроллерів відсутні варіанти з більш ніж одним виходом ЦАП, тому у мікроконтролерному модулі буде використано додатковий зовнішній двоканальний модуль ЦАП необхідний для керування двигунами контуру стрічки транспортера та контуру камери. Також, для коректної роботи, у складі мікроконтролерного модулю необхідні фільтр живлення, перетворювач інтерфейсу RS-232 для обміну даними з ПЕОМ, часозадавальний ланцюг (ЧЗЛ), схема скидання, індикатор режиму роботи системи, кнопка ввімкнення та буфери вихідного керуючого сигналу двигунів. Функціональна схема мікроконтролерного модулю наведена на рис. 2.3.

Таблиця 2.1 – Визначення контролеру методом вагових коефіцієнтів

Параметр	Коефіцієнт	Модель							
		AT89S8252	AT89C2051	ATmega8515	ATTINY12	AT89C55	AT89LP2052	AT89C4051	PIC12F1051
Флеш-пам'ять, Кб	0,1	8	2	8	1	20	4	4	1
Енерго-незалежна пам'ять, Кб	0,1	2	0,5	0,5	0,0625	0,5	0,5	0,25	0,125
Тактова Частота, МГц	0,1	24	24	16	4	24	20	24	16
Енерго-залежна пам'ять, Кб	0,1	0,25	0,125	0,5	0,125	0,25	0,25	0,25	0,0625
Цифрові порти	0,1	32	15	35	5	32	15	15	6
Таймери/Лічильники	0,3	3	2	2	1	3	2	2	2
Зовнішні переривання	0,3	9	6	3	1	8	2	6	1
АЦП	0,3	0	0	0	1	0	0	1	4
ЦАП	0,3	0	0	0	0	0	0	0	1
Ціна, \$	-0,7	1,8	0,95	3,2	1,6	6,9	2,1	1,1	0,82
Разом	1	8,965	5,897	5,26	0,79	6,15	3,7	6,28	4,15

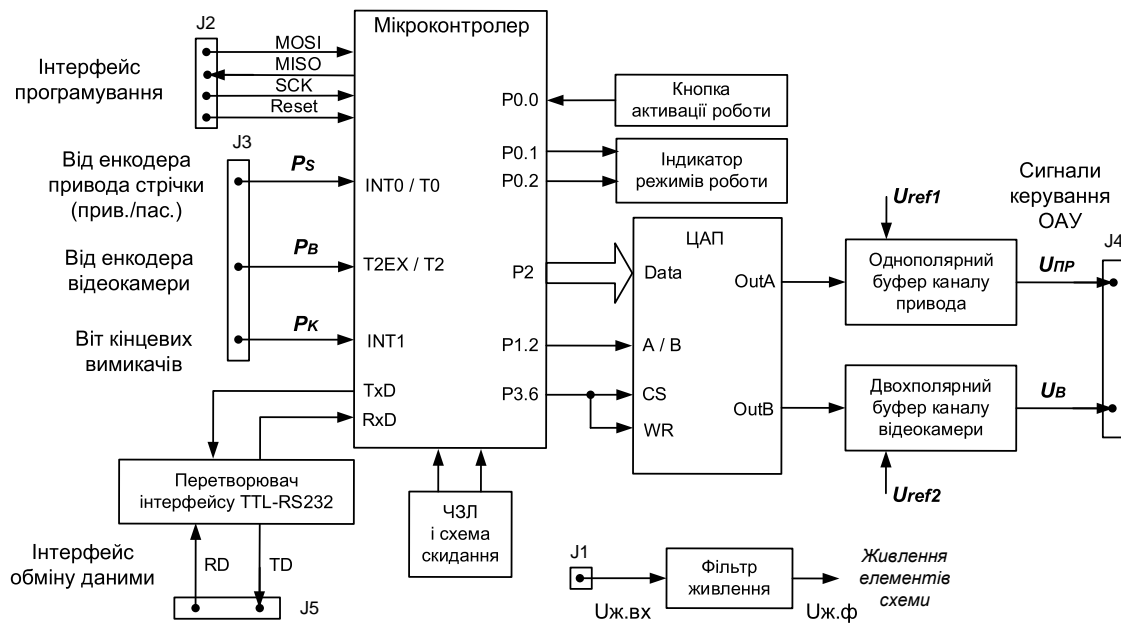


Рисунок 2.3 – Функціональна схема мікроконтролерного модуля

На рис. 2.3 прийняті наступні позначення:

J1-5 – Роз'єми модулю;

Uref1, Uref2 – Опорна напруга ЦАП, В;

Uпр – Напруга керування приводом стрічки, В;

Uв – Напруга керування приводом камери, В;

Uж.вх – вхідна напруга живлення модулю, В;

Uж.ф – відфільтрована напруга живлення, В.

У якості основного мікроконтролеру обрано AT89S8252 з наступними характеристиками:

- 1) 8 Кбайт системної флеш-пам'яті з 1000 циклів перезапису та SPI інтерфейсом для програмування;
- 2) 2 Кбайти енергонезалежної флеш-пам'яті з 100000 циклами перезапису;
- 3) напруга живлення контролеру 4-6В;
- 4) тактова частота контролеру 0-24 МГц;
- 5) 256 байт енергозалежної пам'яті;
- 6) 32 налаштовуваних цифрових портів вводу/виводу;
- 7) 3 16-бітних таймера/лічильника;
- 8) можливість обробки 9 джерел переривання.

Цих характеристик, у складі всього модулю виконавчого пристрою, достатньо для виконання поставленої задачі управління транспортером.

2.4 Формування математичної моделі контуру управління швидкістю стрічки транспортеру

Кінематична модель роботи стрічкового транспортеру наведена на рис. 2.4.

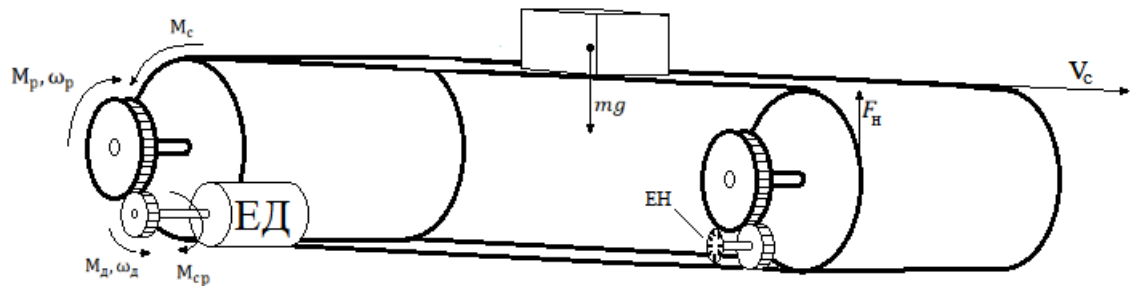


Рисунок 2.4 – Кінематична схема роботи транспортеру

На рис. 2.4 прийняті наступні позначення:

ЕД – Електродвигун;

ЕН – Оптичний енкодер;

M_d – Обертовий момент двигуна, Н*м;

ω_d – Швидкість обертання валу двигуна, $\frac{\text{об}}{\text{хв}}$;

M_p – Обертовий момент на виході редуктора, Н*м;

ω_p – Швидкість обертання на виході редуктора, $\frac{\text{об}}{\text{хв}}$;

M_c – Момент навантаження стрічки, Н*м;

M_{cp} – Момент навантаження на виході редуктора, Н*м;

F_H – сила натягнення стрічки, Н;

mg – вага вантажу, Н;

V_c – Лінійна швидкість стрічки транспортера, мм/с.

Метою розробки САУ є стабілізація лінійної швидкості стрічки транспортеру, стрічка приводиться до руху обертанням одного з валів транспортеру, який, в свою чергу, приводиться до руху обертанням валу електродвигуна через редуктор:

$$\begin{cases} V_c = \omega_p * 2\pi * R; \\ \omega_p = \frac{\omega_d}{i}; \\ \dot{\omega}_d = \frac{M_d - M_c}{J}. \end{cases} \quad (2.1)$$

де: i – Коефіцієнт редуктора;

J – момент інерції валу електродвигуна.

Момент навантаження на валу двигуна, що виступає у якості збурювального впливу, залежить від ваги вантажу розподіленій по стрічці :

$$\begin{cases} M_{cp} = \frac{M_c}{i}; \\ M_c = F_H * R; \\ F_H = 0.5 * l * q_{гр} * c. \end{cases} \quad (2.2)$$

де: R – Радіус валу транспортеру, м;

l – довжина стрічки транспортеру, м;

$q_{гр}$ – Розподілена маса вантажу, $\frac{кг}{м}$;

$c=0.3$ – коефіцієнт супротиву руху стрічки на опорах.

Розподілена маса вантажу розраховуються за наступною формулою:

$$q_{гр} = \frac{m}{0.36 * l} \quad (2.3)$$

де: m – маса вантажу на стрічці, кг;

m_c – маса м²стрічки транспортеру;

w – ширина стрічки транспортеру.

Кутова швидкість обертання валу двигуна залежить від параметрів електродвигуна, та при використанні умови врівноваження падіння напруги на активному опорі якоря ЕДС самоіндукції та противо-ЕДС, розраховується за наступними формулами:

$$\begin{cases} L_{\text{я}} \dot{i}_{\text{я}} + R_{\text{я}} i_{\text{я}} + e_{\text{я}} = U; \\ e_{\text{я}} = k_{\text{ем}} = \frac{U_{\text{н}} - i_{\text{я}} R_{\text{я}}}{\omega_{\text{дн}}}. \end{cases} \quad (2.4)$$

де : $L_{\text{я}}$ – Індуктивність якірної обмотки, Гн;

$i_{\text{я}}$ – Струм у якірній обмотці, А;

$R_{\text{я}}$ – Опір якірної обмотки, Ом;

$e_{\text{я}}$ – противо-ЕДС якоря;

U – Напруга живлення двигуна, В;

$k_{\text{ем}}$ – Електромагнітний коефіцієнт двигуна;

$\omega_{\text{дн}}$ – Номінальна швидкість обертання двигуна, $\frac{\text{об}}{\text{с}}$;

$U_{\text{н}}$ – Номінальна напруга живлення двигуна, В.

В режимі роботи двигуна без навантаження момент супротиву на валу вважається нульовим: :

$$J \dot{\omega}_{\text{д}} = k_{\text{ем}} i_{\text{я}}. \quad (2.5)$$

Підставляючи це у рівняння 2.4 отримуємо:

$$\frac{L_{\text{я}} J}{k_{\text{ем}}} * \ddot{\omega}_{\text{д}} + \frac{R_{\text{я}} J}{k_{\text{ем}}} * \dot{\omega}_{\text{д}} + k_{\text{ем}} \omega = U. \quad (2.6)$$

Після врахування у рівнянні (2.6) електромагнітного коефіцієнту електродвигуна воно приймає наступний вигляд:

$$\frac{L_{\text{я}} R_{\text{я}} J}{R_{\text{я}} k_{\text{ем}}^2} * \ddot{\omega}_{\text{д}} + \frac{R_{\text{я}} J}{k_{\text{ем}}^2} * \dot{\omega}_{\text{д}} + \omega = \frac{U}{k_{\text{ем}}}. \quad (2.7)$$

Вводяться наступні позначення:

$T_{\text{м}} = \frac{R_{\text{я}} J}{k_{\text{ем}}^2}$ – електромеханічна стала часу двигуна;

$T_{\text{я}} = \frac{L_{\text{я}}}{R_{\text{я}}}$ – електромеханічна стала часу якоря.

Вираз 2.7 набуває вигляд:

$$T_{\text{я}} T_{\text{м}} * \ddot{\omega}_{\text{д}} + T_{\text{м}} * \dot{\omega}_{\text{д}} + \omega = \frac{U}{k_{\text{ем}}}. \quad (2.8)$$

2.5 Лінеаризація математичної моделі та визначення передавальних функцій контуру управління швидкістю стрічки транспортеру

З метою лінеаризації математичної моделі системи до неї вводяться параметри стрічкового транспортеру та електродвигуна, дані параметри наведені у табл. 2.2.

Таблиця 2.2 – параметри транспортеру та електродвигуна

Характеристика	Значення
Номінальна потужність, Вт	2,67
Номінальна напруга, В	27
Номінальна частота обертання, об/хв	5400
Номінальний струм на якорі, А	0,11
Момент інерції якоря, кг·м ²	1,615*10 ⁻³
Індуктивність якоря, Гн	0,16
Опір обмотки якоря, Ом	30
Радіус барабану, м	0.021
Маса барабану, кг	0.5
Довжина стрічки, м	1.1
Коефіцієнт редуктора	20

Математична модель ОАУ після урахування параметрів приймає наступний вигляд::

$$\begin{cases} 0.005 * \ddot{\omega}_{\text{д}} + 0.7 * \dot{\omega}_{\text{д}} + \omega_{\text{д}} = 3.8 U; \\ V_{\text{с}} = 6.6 \omega_{\text{д}}; \\ M_{\text{ср}} = 0.435m * 10^{-3}. \end{cases} \quad (2.9)$$

Створена структурна сема ОАУ, зображена на рис. 2.5, дана схема необхідна для подальшого аналізу функціонування ОАУ та розробки САУ.

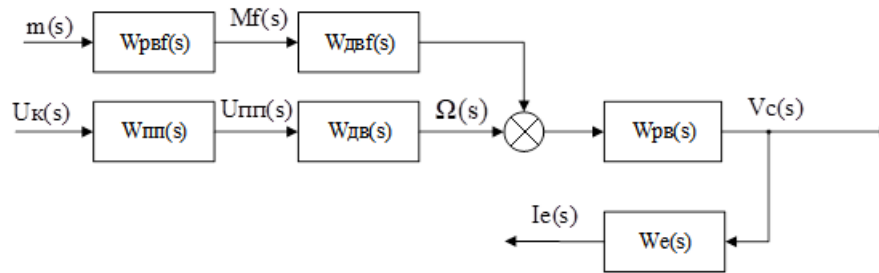


Рисунок 2.5 – Структурна схема ОАУ

На рис. 2.5 прийнято наступні позначення:

$W_{пп}(s) = \frac{U_{пп}(s)}{U_k(s)}$ – Передавальна функція підсилювача потужності;

$W_{дв}(s) = \frac{\Omega(s)}{U_{пп}(s)}$ – Передавальна функція електродвигуна за керуючим впливом, $\frac{\text{об}}{\text{В}\cdot\text{с}}$;

$W_{рвf}(s) = \frac{M_f(s)}{m(s)}$ – Передавальна функція валу та редуктора по збурюючому впливу, $\frac{\text{Н}\cdot\text{м}}{\text{кг}}$;

$W_{двf}(s) = \frac{\Omega(s)}{M_f(s)}$ – Передавальна електродвигуна за збурюючим впливом, $\frac{\text{об}}{\text{Н}\cdot\text{м}\cdot\text{с}}$;

$W_{рв}(s) = \frac{V_c(s)}{\Omega(s)}$ – Передавальна функція валу та редуктора $\frac{\text{мм}\cdot\text{с}}{\text{об}\cdot\text{с}}$;

$W_e(s) = \frac{I_e(s)}{V_c(s)}$ – Передавальна функція оптичного енкодера $\frac{\text{В}\cdot\text{с}}{\text{мм}}$;

$m(s)$ – зображення сигналу ваги вантажу;

$M_f(s)$ – Зображення сигналу збурюючого моменту на валу двигуна;

$U_k(s)$ – Зображення сигналу керуючої напруги;

$U_{пп}(s)$ – Зображення сигналу напруги підсилювача потужності;

$\Omega(s)$ – Зображення сигналу кутової швидкості валу двигуна;

$V_c(s)$ – Зображення сигналу лінійної швидкості стрічки;

$I_e(s)$ – Зображення сигналу вихідної напруги енкодера.

Виходячи з отриманих рівнянь, що описують роботу ОАУ, отримані передавальні функції компонентів ОАУ шляхом перетворення Лапласа. Через досить велику швидкодію підсилювач потужності вважається безінерційним елементом та описується лише коефіцієнтом підсилення:

$$W_{\text{пп}}(s) = \frac{U_{\text{пп}}(s)}{U_k(s)} = 5.4 \quad (2.10)$$

Передавальна функція електродвигуна за керуючим впливом:

$$W_{\text{дв}}(s) = \frac{\Omega(s)}{U_{\text{пп}}(s)} = \frac{3.63}{0.002s^2 + 0.242s + 1} \frac{\text{об}}{\text{В} * \text{с}} \quad (2.11)$$

Передавальна функція валу та редуктору, що передають момент супротиву від ваги вантажу на вал двигуна

$$W_{\text{рвф}}(s) = \frac{M_f(s)}{m(s)} = 0.435 * 10^{-3} \frac{\text{Н} * \text{м}}{\text{кг}} \quad (2.12)$$

Передавальна функція двигуна за вагою вантажу, що створює момент супротиву на валу двигуна:

$$W_{\text{двф}}(s) = \frac{\Omega(s)}{M_f(s)} = \frac{-433.22}{0.002s^2 + 0.242s + 1} \frac{\text{об}}{\text{Н} * \text{м} * \text{с}} \quad (2.13)$$

Передавальна функція валу та редуктора що передають обертовий момент з валу електродвигуна на приводний вал транспортеру:

$$W_{\text{рв}}(s) = \frac{V_c(s)}{\Omega(s)} = 6.6 \frac{\text{мм} * \text{с}}{\text{об} * \text{с}} \quad (2.14)$$

Передавальна функція оптичного енкодера на пасивному валу , який перетворює швидкість обертання валу на частоту логічних імпульсів приймається рівною одиниці, оскільки розрахунок швидкості обертання проводиться контролером САУ де це враховується:

$$W_e(s) = \frac{I_e(s)}{V_c(s)} = 1 \frac{\text{В} * \text{с}}{\text{мм}} \quad (2.15)$$

2.6 Машинне моделювання контуру управління швидкістю стрічки

На основі отриманих передавальних функцій елементів ОАУ сформована машинна модель руху стрічки транспортеру у середовищі Matlab Simulink з метою дослідження статичних та динамічних характеристик об'єкту, створена машинна модель зображена на рис. 2.6.

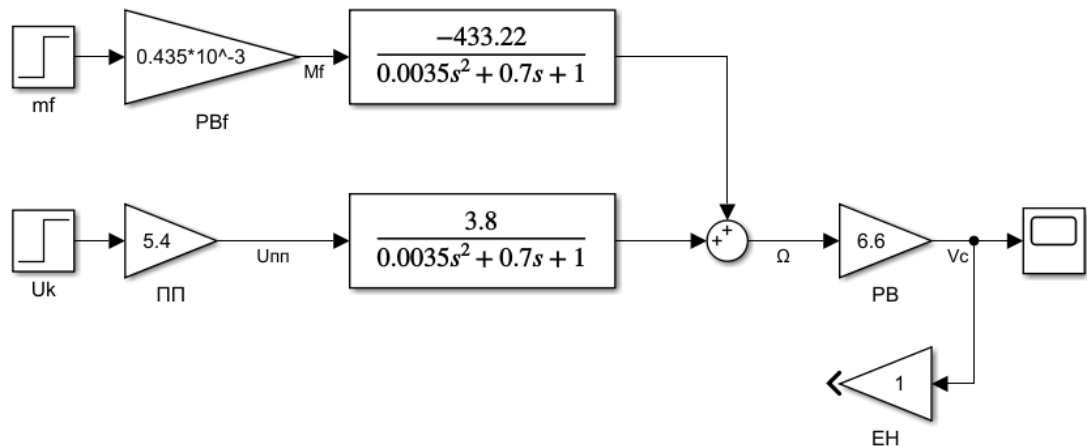


Рисунок 2.6 – Машинна модель ОАУ

Шляхом моделювання роботи системи отримані динамічні характеристики ОАУ у вигляді перехідних процесів за лінійною швидкістю стрічки транспортера за керуючим та збурювальним впливами та частотні характеристики системи, перехідні процеси наведені на рис. 2.7 та рис. 2.8, частотні характеристики системи наведені на рис. 2.9.

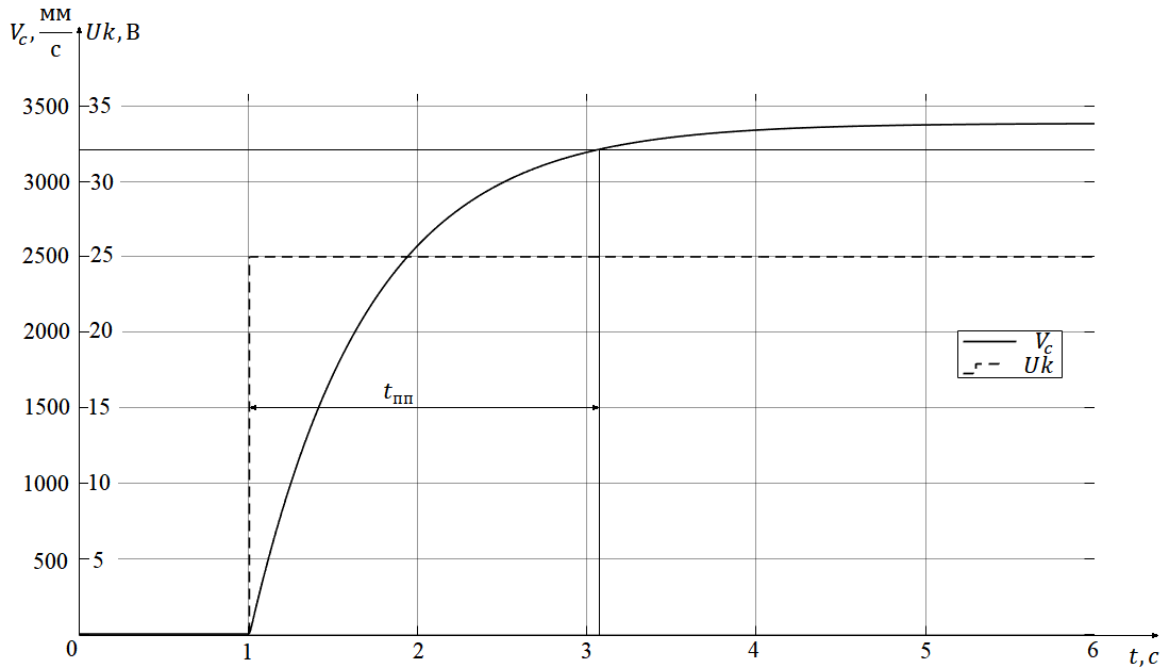


Рисунок 2.7 – Графік перехідного процесу ОАУ за керуючим впливом

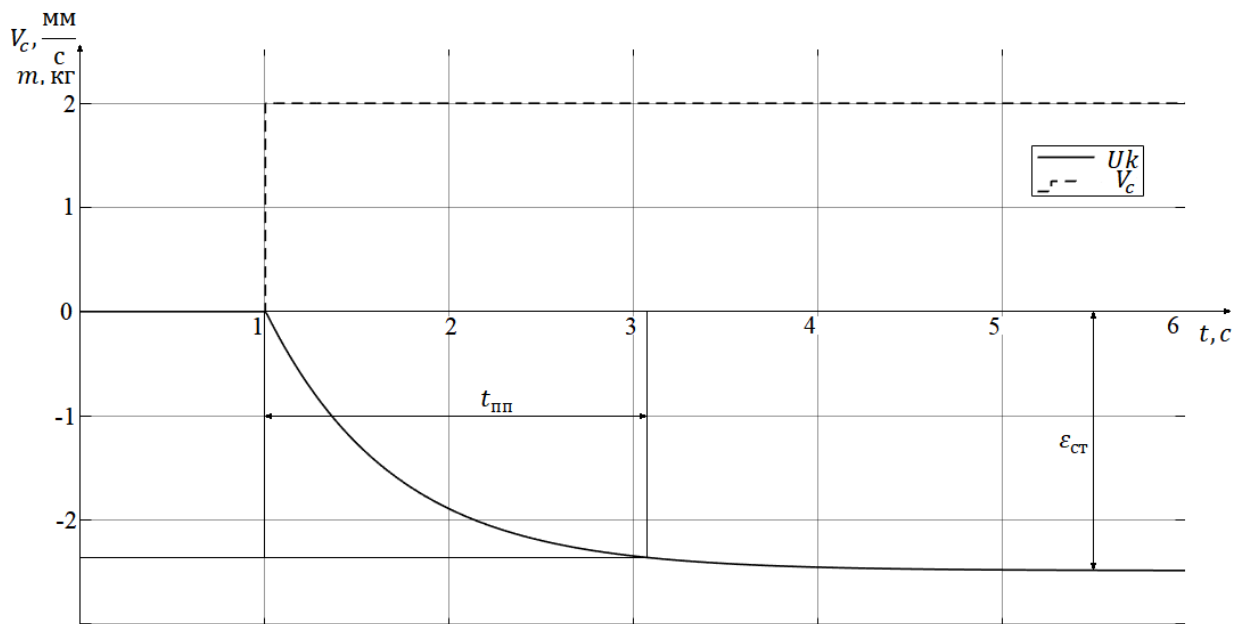


Рисунок 2.8 – Графік перехідного процесу ОАУ за збурюючим впливом

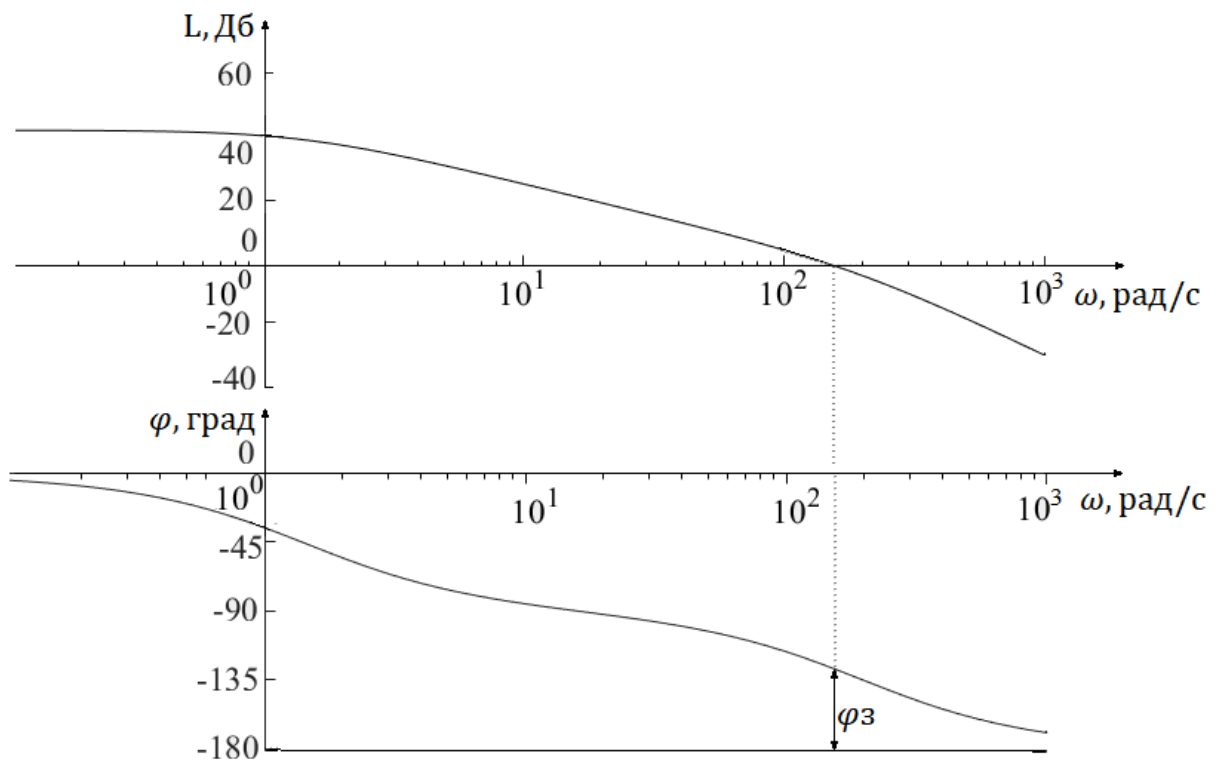


Рисунок 2.9 – Частотні характеристики ОАУ

Показники якості об'єкту управління зв контуром керування швидкістю стрічки, визначені з отриманих динамічних та частотних характеристик системи, надані у табл. 2.3.

Таблиця 2.3 – Показники якості ОАУ

№ з/п	Тип впливу	$t_{пп}, c$	$\sigma, \%$	$\varepsilon_{ст}, \%$	$L_3, Дб$	$\varphi_3, ^\circ$
1	Керуючий вплив	2.15	0	13408	∞	52.7
2	Збурюючий вплив	2.15	0	9.8	-	-

2.7 Формування математичної моделі контуру керування швидкістю камери

Кінематична схема контуру керування швидкістю камери наведена на рис. 2.10

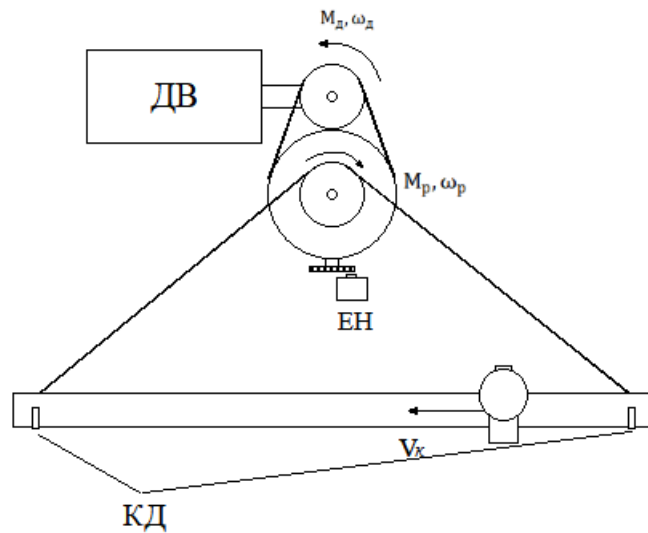


Рисунок 2.10 – Кінематична схема контуру камери

На рис. 2.10 прийняті наступні позначення:

ЕД – Електродвигун;

ЕН – Оптичний енкодер;

КД – Кінцеві датчики положення камери;

M_d – Обертовий момент двигуна, Н*м;

ω_d – Швидкість обертання валу двигуна, $\frac{\text{об}}{\text{хв}}$;

M_p – Обертовий момент на виході редуктора, Н*м;

ω_p – Швидкість обертання на виході редуктора, $\frac{\text{об}}{\text{хв}}$;

V_k – Лінійна швидкість стрічки транспортера, мм/с.

Метою розробки системи управління є стабілізація лінійної швидкості камери, у якості керуючого впливу виступає швидкість руху стрічки транспортеру, що дозволить камері рухатися паралельно вантажу та постійно отримувати інформацію про його стан. Робота даного контуру розглядається лише за керуючим впливом оскільки у контрольованому середовищі використання стенду не передбачається істотних збурюючих впливів на рух камери що можуть призвести до виникнення похибки у роботі системи. Камера приводиться до руху електродвигуном який через редуктор під'єднано до приводу камери:

$$\begin{cases} V_k = \omega_p * K_{\Pi} * 2\pi R_{\Pi}; \\ \omega_p = \frac{\omega_d}{i}; \\ \dot{\omega}_d = \frac{M_d}{J}. \end{cases} \quad (2.16)$$

де: i – Коефіцієнт редуктора;

K_{Π} - Коефіцієнт приводу камери;

R_{Π} – Радіус валу приводу камери;

J – момент інерції валу електродвигуна.

У контурі керування швидкістю камери використовується та сама модель двигуна що й у контурі керування швидкістю стрічки, тому використовується раніше визначена математична модель двигуна.

2.8 Лінеаризація математичної моделі контуру керування камерою

Виходячи з відомих параметрів установки математична модель контуру керування швидкістю камери має наступний вигляд::

$$\begin{cases} 0.005 * \ddot{\omega}_d + 0.7 * \dot{\omega}_d + \omega_d = 3.8 U; \\ \omega_p = \frac{\omega_d}{20}; \\ V_k = \omega_p * 75.36; \end{cases} \quad (2.17)$$

Розроблена структурна схема контуру керування швидкістю камери, схема зображена на рис. 2.11.

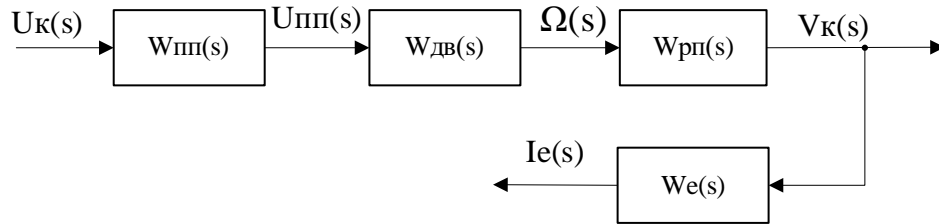


Рисунок 2.11 – Структурна схема контуру керування швидкістю камери

На рис. 2.11 прийнято наступні позначення:

$W_{пп}(s) = \frac{U_{пп}(s)}{U_k(s)}$ – Передавальна функція підсилювача потужності;

$W_{дв}(s) = \frac{\Omega(s)}{U_{пп}(s)}$ – Передавальна функція двигуна, $\frac{\text{об}}{\text{В*с}}$;

$W_{рп}(s) = \frac{V_c(s)}{\Omega(s)}$ – Передавальна функція редуктора та привода $\frac{\text{мм*с}}{\text{об*с}}$;

$W_e(s) = \frac{I_e(s)}{V_c(s)}$ – Передавальна функція оптичного енкодера $\frac{\text{В*с}}{\text{мм}}$;

$U_k(s)$ – Зображення сигналу керуючої напруги;

$U_{пп}(s)$ – Зображення сигналу напруги підсилювача потужності;

$\Omega(s)$ – Зображення сигналу кутової швидкості валу двигуна;

$V_k(s)$ – Зображення сигналу лінійної швидкості камери;

$I_e(s)$ – Зображення сигналу вихідної напруги енкодера.

Виходячи з математичної моделі отримані передавальні функції елементів контуру управління швидкістю камери, як і у контурі управління швидкістю стрічки підсилювач потужності вважається безінерційним:

$$W_{пп}(s) = \frac{U_{пп}(s)}{U_k(s)} = 5.4 \quad (2.18)$$

Передавальна функція електродвигуна:

$$W_{дв}(s) = \frac{\Omega(s)}{U_{пп}(s)} = \frac{3.63}{0.002s^2 + 0.242s + 1} \frac{\text{об}}{\text{В * с}} \quad (2.19)$$

Передавальна функція редуктора та приводу камери:

$$W_{\text{рп}}(s) = \frac{V_c(s)}{\Omega(s)} = 3.76 \frac{\text{мм} * \text{с}}{\text{об} * \text{с}} \quad (2.20)$$

Передавальна функція оптичного енкодера, який перетворює швидкість обертання на частоту логічних імпульсів приймається рівній одиниці, оскільки розрахунок швидкості проводиться програмно на контролері САУ:

$$W_e(s) = \frac{I_e(s)}{V_c(s)} = 1 \frac{\text{В} * \text{с}}{\text{мм}} \quad (2.21)$$

2.9 Машинне моделювання контуру керування швидкістю камери

Для визначення динамічних та частотних характеристик об'єкту використовується машинне моделювання його роботи у середовищі Matlab Simulink, машинна модель наведена на рис. 2.12.

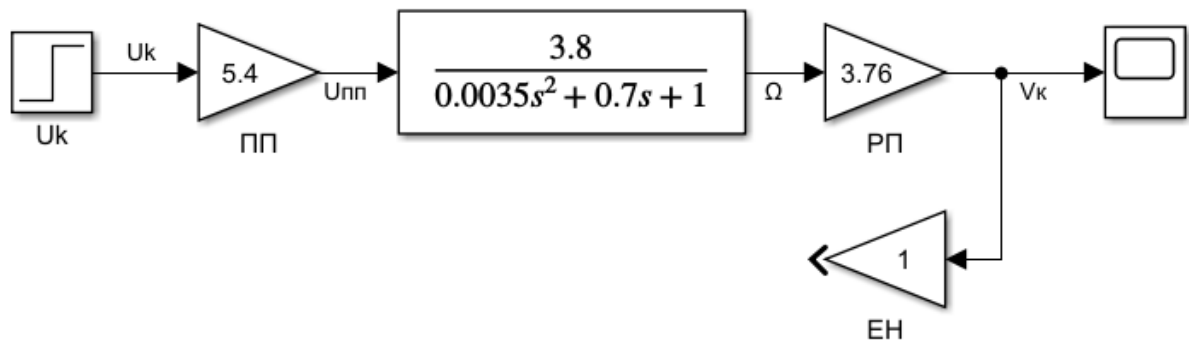


Рисунок 2.12 – Машинна модель контуру керування швидкістю камери

У результаті моделювання роботи системи отримані динамічна характеристика у вигляді графіку перехідного процесу та частотні характеристики системи, графік перехідного процесу системи за керуючим впливом наведено на рис. 2.13, частотні характеристики системи зображені на рис. 2.14.

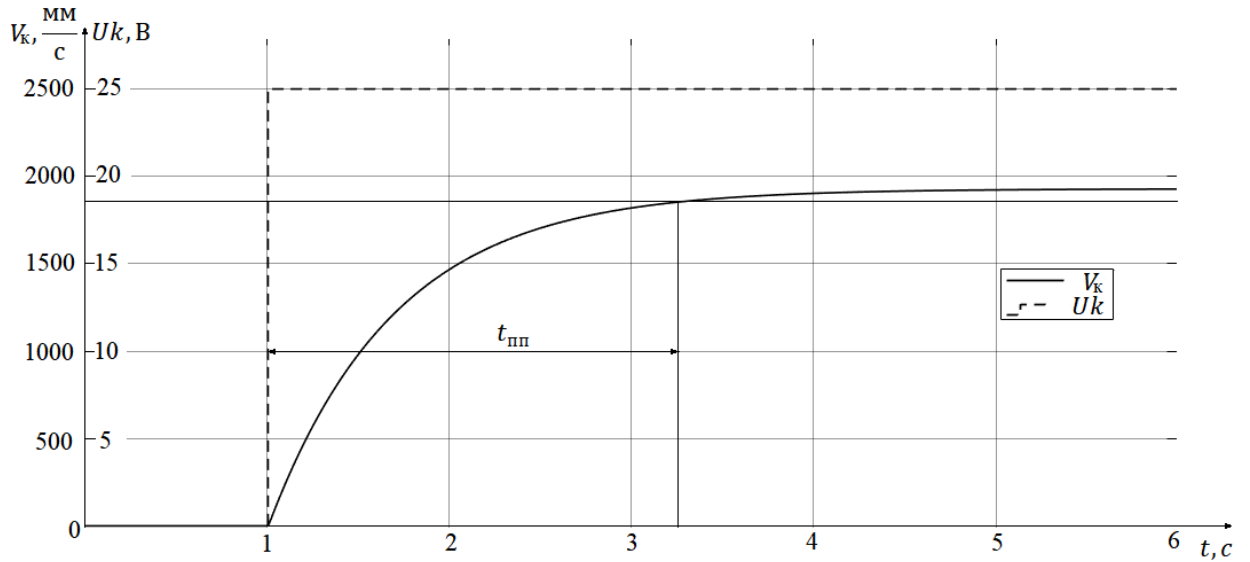


Рисунок 2.13 – Графік перехідного процесу контуру керування швидкістю камери

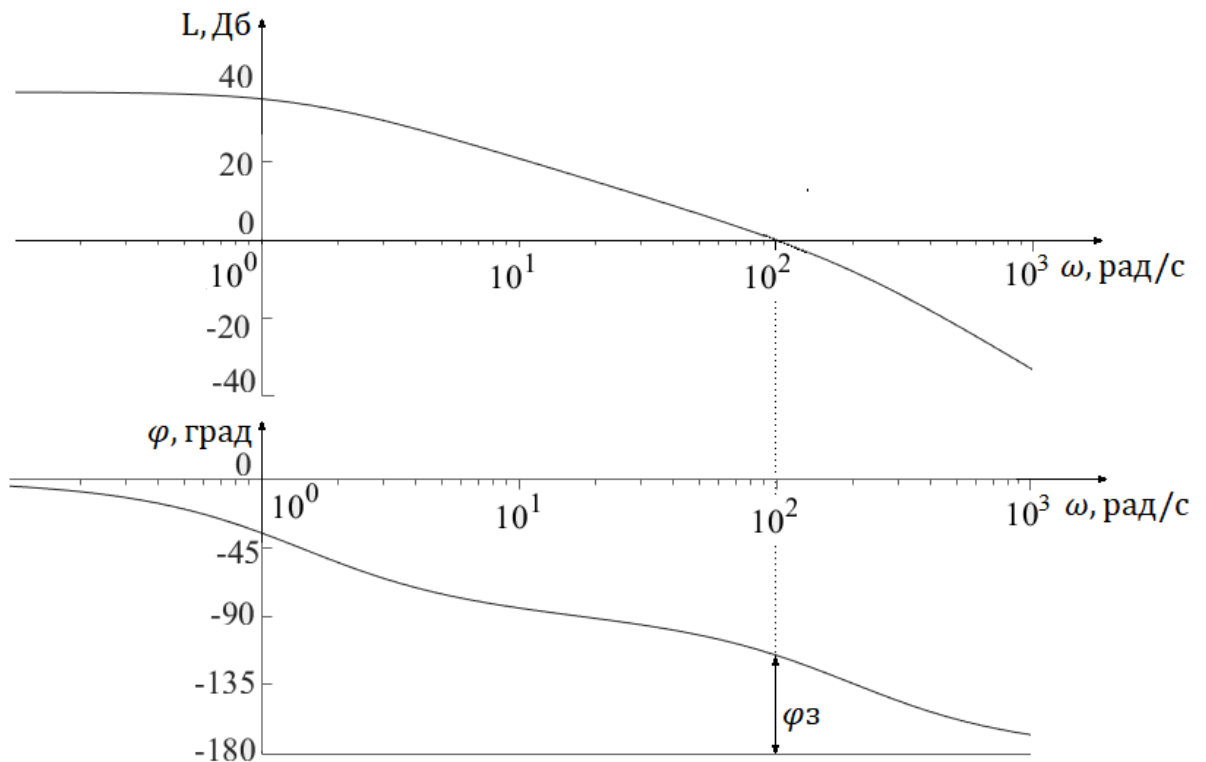


Рисунок 2.14 – Частотні характеристики контуру керування швидкістю камери

Показники якості функціонування контуру, визначені виходячи з динамічних та частотних характеристик, надані у табл. 2.4.

Таблиця 2.4 – Показники якості контуру керування швидкістю камери

Тип впливу	$t_{пп}, c$	$\sigma, \%$	$\epsilon_{ст}, \%$	$L_3, Дб$	$\varphi_3, ^\circ$
Керуючий вплив	2.18	0	7608	∞	52.7

2.10 Постановка задачі синтезу

Задача синтезу системи автоматичного управління полягає у доповненні існуючого ОАУ створеним пристроєм автоматичного управління, отримані після доповнення ОАУ системи автоматичного управління за контурами керування швидкістю стрічки та керування швидкістю камери мають відповідати наступним вимогам::

- 1) $t_{пп} \leq 0.5c$ – час перехідного процесу;
- 2) $\sigma \leq 5\%$ - перерегулювання;
- 3) $\epsilon_{ст} \leq 0.1\%$ - стала похибка;
- 4) $\varphi_3 \geq 40^\circ$ - запас стійкості за фазою;
- 5) $L_3 \geq 10дБ$ – запас стійкості за амплітудою.

2.10 Формування функціональної та структурної схеми САУ

Створена функціональна схема САУ за контуром керування швидкістю стрічки виходячи з принципу управління за відхиленням, схема зображена на рис. 2.15.

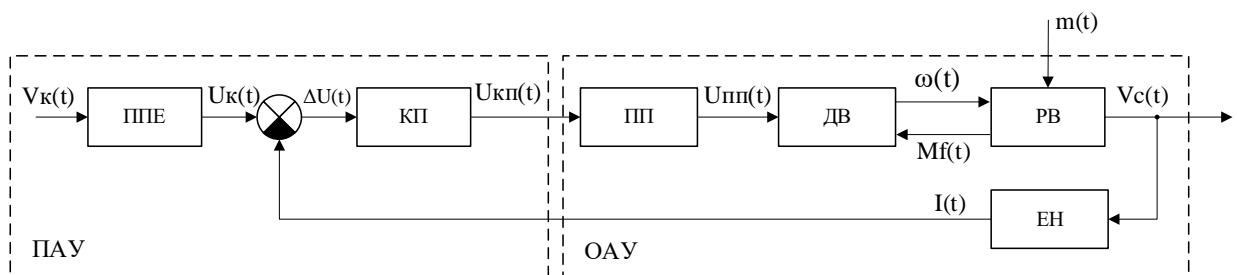


Рисунок 2.15 – Функціональна схема САУ за контуром швидкості стрічки

На рис. 2.15 прийнято наступні позначення:

ППЕ – Перетворюючий пристрій;

КП – Коригуючий пристрій;

ПП – Підсилювач потужності;

ДВ – Двигун;

РВ – Редуктор та вал транспортеру;

ЕН – Оптичний енкодер;

$V_k(t)$ – Керуючий сигнал лінійної швидкості, $\frac{\text{мм}}{\text{с}}$;

$U_k(t)$ – Керуюча напруга, В;

$\Delta U(t)$ – Напруга відхилення, В;

$U_{\text{кп}}(t)$ – Напруга коригуючого пристрою, В;

$U_{\text{пп}}(t)$ – Напруга підсилювача потужності, В;

$\omega(t)$ – Кутова швидкість валу електродвигуна, $\frac{\text{об}}{\text{с}}$;

$V_c(t)$ – Лінійна швидкість стрічки транспортеру, $\frac{\text{мм}}{\text{с}}$;

$m(t)$ – Вага вантажу транспортеру, кг;

$M_f(t)$ – Збурюючий момент від стрічки на валу двигуна, Н*м;

$I(t)$ – Сигнал з оптичного енкодера, В.

Виходячи з функціональної схеми САУ контуру управління швидкістю стрічки створено структурну схему САУ, схема зображена на рис. 2.16.

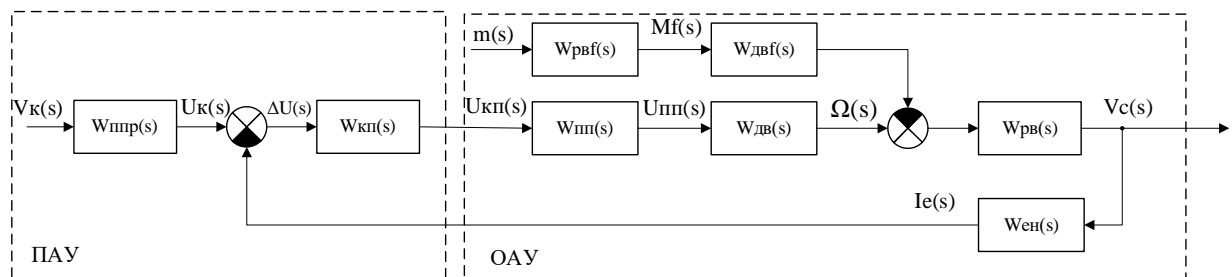


Рисунок 2.16 – Структурна схема САУ за контуром швидкості стрічки

На рис. 2.16 прийняті наступні позначення:

$W_{\text{ппп}}(s) = \frac{U_k(s)}{V_k(s)} = 1$ – Передавальна функція перетворюючого пристрою,

$\frac{\text{В} \cdot \text{с}}{\text{мм}}$,

$W_{\text{кп}}(s) = \frac{U_{\text{кп}}(s)}{\Delta U(s)}$ – Передавальна функція коригуючого пристрою;

$W_{\text{пп}}(s) = \frac{U_{\text{пп}}(s)}{U_k(s)} = 5.4$ – Передавальна функція підсилювача потужності;

$W_{дв}(s) = \frac{\Omega(s)}{U_{пп}(s)} = \frac{3.8}{0.0035s^2 + 0.7s + 1}$ – Передавальна функція двигуна за керуючим впливом, $\frac{\text{об}}{\text{В*с}}$;

$W_{рвф}(s) = \frac{M_f(s)}{m(s)} = 0.435 * 10^{-3}$ – Передавальна функція валу та редуктора по збурюючому впливу, $\frac{\text{Н*м}}{\text{кг}}$;

$W_{двф}(s) = \frac{\Omega(s)}{M_f(s)} = \frac{-433.22}{0.0035s^2 + 0.7s + 1}$ – Передавальна двигуна за збурюючим впливом, $\frac{\text{об}}{\text{Н*м*с}}$;

$W_{рв}(s) = \frac{V_c(s)}{\Omega(s)} = 6.6$ – Передавальна функція валу та редуктора $\frac{\text{мм*с}}{\text{об*с}}$;

$W_e(s) = \frac{I_e(s)}{V_c(s)} = 1$ – Передавальна функція оптичного енкодера $\frac{\text{В*с}}{\text{мм}}$;

$V_k(s)$ – Зображення керуючого значення лінійної швидкості;

$U_k(s)$ – Зображення сигналу керуючої напруги;

$\Delta U(s)$ – Зображення сигналу напруги відхилення;

$m(s)$ – Зображення сигналу ваги вантажу;

$M_f(s)$ – Зображення сигналу збурюючого моменту на валу двигуна;

$U_{пп}(s)$ – Зображення сигналу напруги підсилювача потужності;

$\Omega(s)$ – Зображення сигналу кутової швидкості валу двигуна;

$V_c(s)$ – Зображення сигналу лінійної швидкості стрічки;

$I_e(s)$ – Зображення сигналу вихідної напруги енкодера.

Виходячи з принципу управління за відхиленням сформовано функціональну схему САУ за контуром управління швидкістю камери, схема зображена на рис. 2.17.

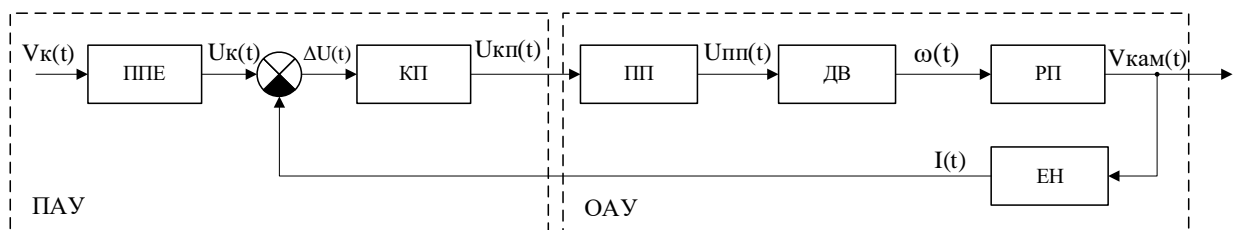


Рисунок 2.17 – Функціональна схема САУ за контуром швидкості камери

На рис. 2.17 прийнято наступні позначення:

ППЕ – Перетворюючий пристрій;

КП – Корируючий пристрій;

ПП – Підсилювач потужності;

ДВ – Двигун;

РП – Редуктор та привод камери;

ЕН – Оптичний енкодер;

$V_k(t)$ – Керуючий сигнал лінійної швидкості, $\frac{\text{мм}}{\text{с}}$;

$U_k(t)$ – Керуюча напруга, В;

$\Delta U(t)$ – Напруга відхилення, В;

$U_{\text{кп}}(t)$ – Напруга коригуючого пристрою, В;

$U_{\text{пп}}(t)$ – Напруга підсилювача потужності, В;

$\omega(t)$ – Кутова швидкість валу електродвигуна, $\frac{\text{об}}{\text{с}}$;

$V_{\text{кам}}(t)$ – швидкість камери, $\frac{\text{мм}}{\text{с}}$;

$I(t)$ – Сигнал з оптичного енкодера, В.

Виходячи з функціональної схеми САУ за контуром управління швидкістю камери створена структурна схема за контуром швидкості камери, схема зображена на рис. 2.18.

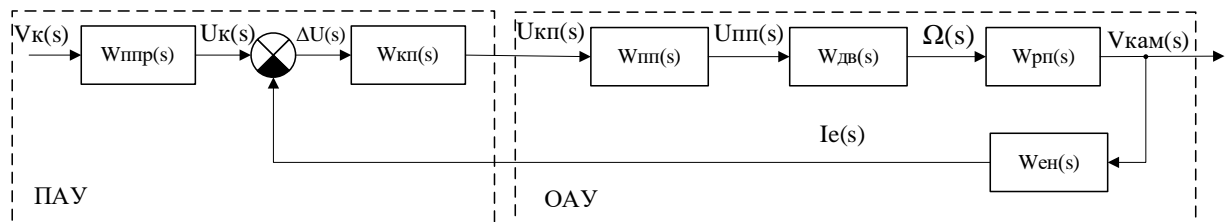


Рисунок 2.18 – Структурна схема САУ контуру керування швидкістю камери

На рис. 2.18 прийняті наступні позначення:

$W_{\text{ппр}}(s) = \frac{U_k(s)}{V_k(s)} = 1$ – Передавальна функція перетворюючого пристрою,

$\frac{\text{В} \cdot \text{с}}{\text{мм}}$,

$W_{\text{кп}}(s) = \frac{U_{\text{кп}}(s)}{\Delta U(s)}$ – Передавальна функція коригуючого пристрою;

$W_{\text{пп}}(s) = \frac{U_{\text{пп}}(s)}{U_k(s)} = 5.4$ – Передавальна функція підсилювача потужності;

$W_{\text{дв}}(s) = \frac{\Omega(s)}{U_{\text{пп}}(s)} = \frac{3.8}{0.0035s^2 + 0.7s + 1}$ – Передавальна функція двигуна за

керуючим впливом, $\frac{\text{об}}{\text{В} \cdot \text{с}}$;

$W_{\text{рп}}(s) = \frac{V_c(s)}{\Omega(s)} = 3.76$ – Передавальна функція редуктора та приводу камери $\frac{\text{мм} \cdot \text{с}}{\text{об} \cdot \text{с}}$;

$W_e(s) = \frac{I_e(s)}{V_c(s)} = 1$ – Передавальна функція оптичного енодера $\frac{\text{В} \cdot \text{с}}{\text{мм}}$;

$V_k(s)$ – Зображення керуючого значення лінійної швидкості;

$U_k(s)$ – Зображення сигналу керуючої напруги;

$\Delta U(s)$ – Зображення сигналу напруги відхилення;

$U_{\text{пн}}(s)$ – Зображення сигналу напруги підсилювача потужності;

$\Omega(s)$ – Зображення сигналу кутової швидкості валу двигуна;

$V_{\text{кам}}(s)$ – Зображення сигналу лінійної швидкості стрічки;

$I_e(s)$ – Зображення сигналу вихідної напруги енодера.

2.11 Дослідження характеристик нескоригованої САУ

Для визначення динамічних характеристик системи проводиться машинне моделювання роботи у середовищі Matlab Simulink, машинна модель системи зображена на рис. 2.19, графіки перехідних процесів за керуючим та збурюючим впливами наведені на рис. 2.20 та 2.21.

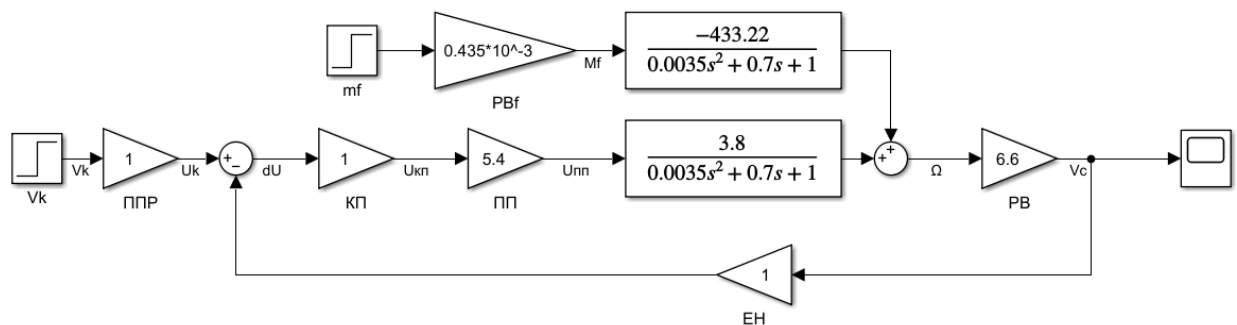


Рисунок 2.19 – Машинна модель САУ за контуром швидкості стрічки

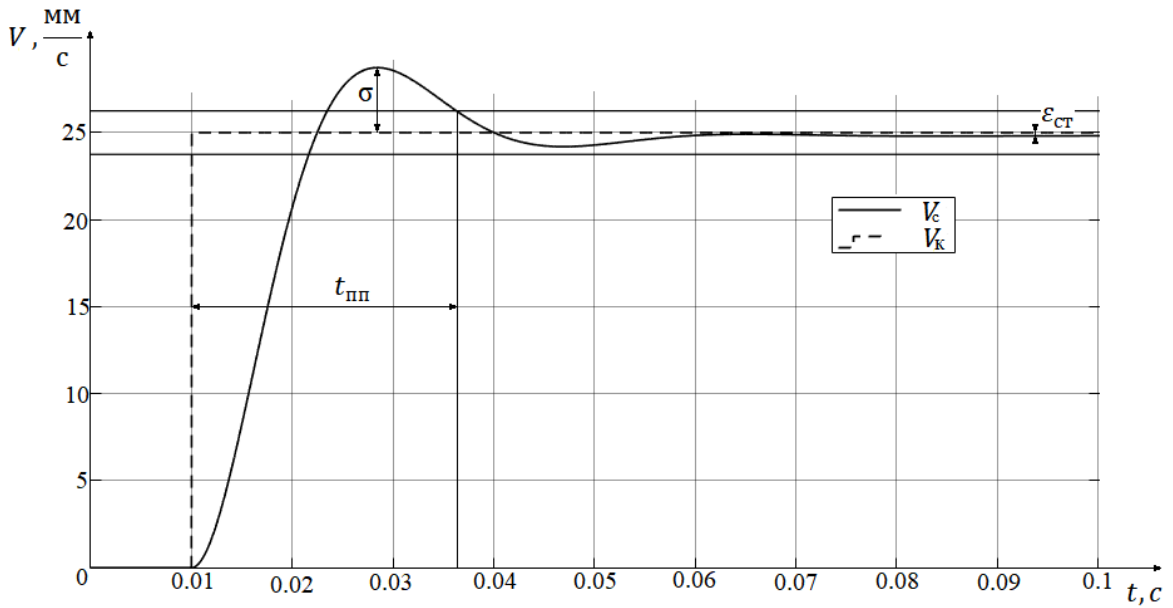


Рисунок 2.20 – Графік перехідного процесу САУ за керуючим впливом

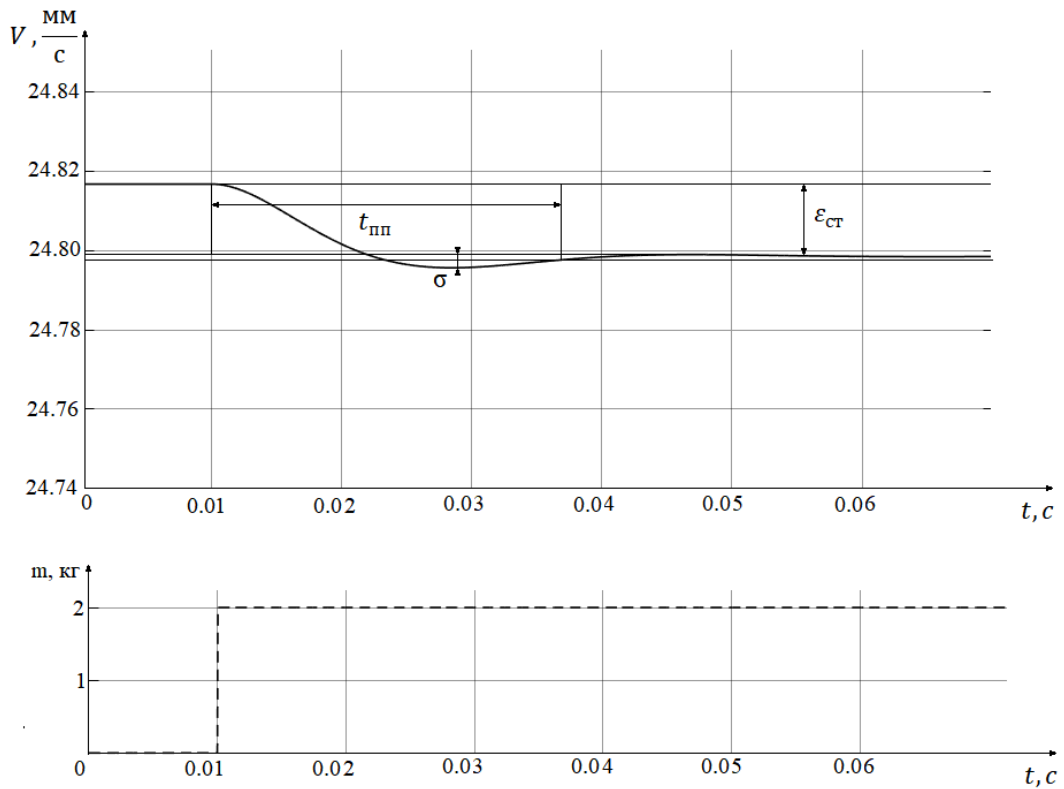


Рисунок 2.21 – Графік перехідного процесу системи за збурюючим впливом

Спираючись на отримані динамічні характеристики системи визначені показники якості функціонування, вони наведені у табл. 2.5.

Таблиця 2.5 – Показники якості нескоригованої САУ за контуром швидкості стрічки

№ з/п	Тип впливу	Тип ПП	χ	$t_{пп}, c$	$\sigma, \%$	$\varepsilon_{ст}, \%$
1	Керуючий вплив	коливальний	1	0.026	15.7	0.32
2	Збурюючий вплив	коливальний	1	0.026	15.5	0.0072

Машинна модель контуру керування швидкістю камери зображена на рис. 2.22, графік перехідного процесу за керуючим впливом наведено на рис. 2.23.

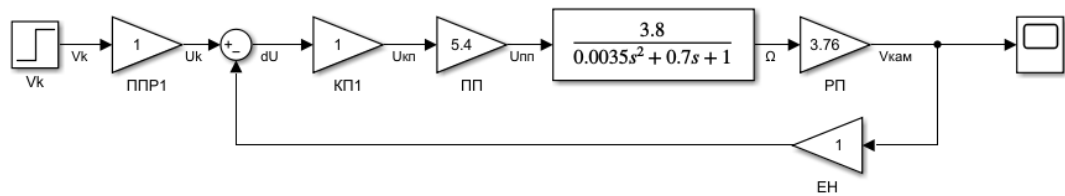


Рисунок 2.22 – Машинна модель САУ за контуром швидкості камери

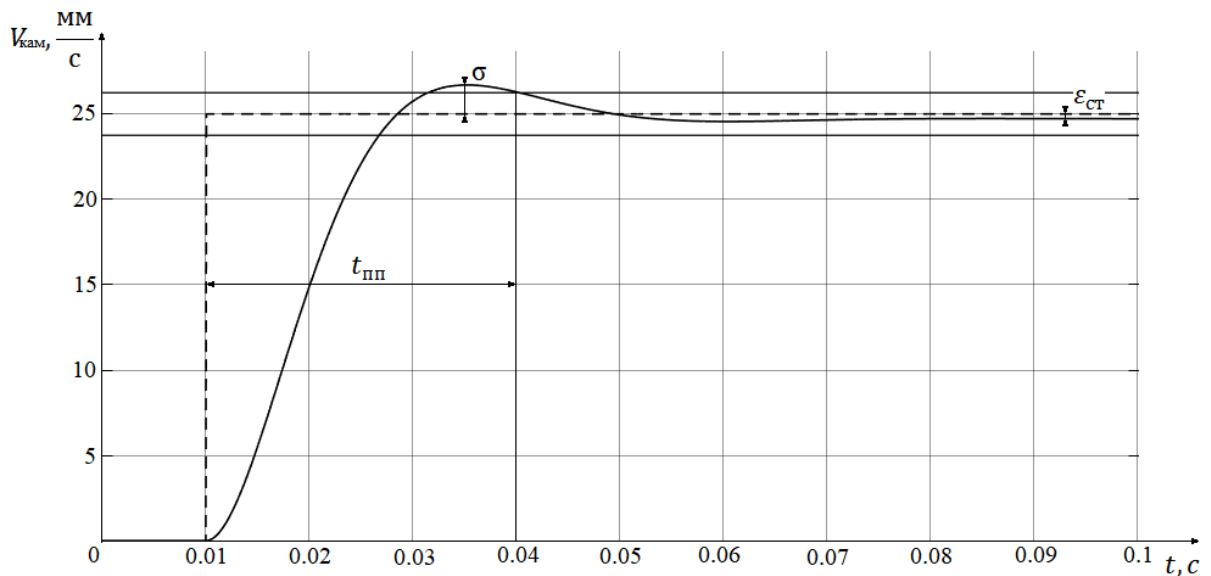


Рисунок 2.23 – Графік перехідного процесу за керуючим впливом

Визначені, виходячи з отриманих графіків перехідних процесів системи, показники якості наведені у табл. 2.6.

Таблиця 2.6 – Показники якості нескоригованої САУ контуру керування швидкістю камери

Тип впливу	Тип ПП	χ	$t_{пп}, c$	$\sigma, \%$	$\epsilon_{ст}, \%$
Керуючий вплив	коливальний	1	0.03	7.9	0.108

Показники якості сформованих САУ не відповідають висунутим у технічному завданні умовам, а саме величині перерегулювання та статичній похибці системи. Необхідно провести синтез коригуючого елемента системи щоб досягти необхідних показників якості функціонування системи.

2.12 Синтез коригуючого пристрою

Функція коригуючого пристрою полягає у перетворенні вхідного сигналу, який залежить від обраного принципу управління системи, на вихідний, що, при впливі на ОАУ, призведе до зміни вихідного сигналу системи, відповідно до необхідних показників якості. При створенні систем управління за принципом керування за відхиленням вхідним сигналом коригуючого пристрою є різниця між керуючим впливом та вихідним сигналом системи. Серед можливих варіантів реалізації регулятора обрано ПД-регулятор, оскільки, на відміну від синтезованого методом ЛАЧХ, або керування з використанням нейронних мереж чи інших, більш складних за алгоритмами регуляторів, ПД вимагає набагато менше обчислювальних операцій для однієї ітерації розрахунку вихідного сигналу, що, при обмежених обчислювальних потужностях контролеру САУ призведе до значного збільшення можливої швидкодії системи.

ПД регулятор представляє собою три паралельні ланки, пропорційну інтегральну та диференціальну, вихідний сигнал регулятора відповідно представляє собою суму вихідних сигналів цих ланок, їх передавальні функції:

$$\begin{aligned}
 W_{п}(s) &= K_{п}; \\
 W_{і}(s) &= \frac{K_{і}}{s}; \\
 W_{д}(s) &= K_{д}s.
 \end{aligned}
 \tag{2.22}$$

Пропорційна ланка представляє собою простий пропорційний коефіцієнт, його збільшення призводить до збільшення швидкодії системи, зменшення статично похибки та збільшення амплітуди коливань. Занадто велике збільшення коефіцієнту ланки може зробити систему нестійкою.

Інтегральна складова регулятора накопичує сигнал похибки пропорційно до свого коефіцієнту, що позбавляє систему статичної похибки, при цьому час зменшення похибки залежить від коефіцієнту ланки..

Диференціальна ланка створює вихідний сигнал пропорційно до швидкості зміни вхідного сигналу, що дозволяє значно збільшити швидкодію системи через збільшення вихідного сигналу на початку перехідного процесу, як ступінчатого так і при лінійному зростанні сигналу, проте збільшення диференціального коефіцієнту призводить до збільшення амплітуди та частоти коливань чи перерегулювання..

Існують різні варіанти реалізації ПД регулятора, які відрізняються використаними ланками. У даній системі буде використано ПІ регулятор оскільки нескоригована система відповідає вимогам по швидкодії та не потребує додаткового форсування керуючого впливу яке збільшить перерегулювання.

Методи аналітичного розрахунку коефіцієнтів регулятора дають можливість розрахувати лише опорні значення, оскільки вони не спираються на конкретну структуру чи параметри системи, опорні значення коефіцієнтів регулятора вимагають подальшої оптимізації шляхом підбору. У даній роботі коефіцієнти регулятора розраховуються за методом Чиєна-Хронеса-Рєсвіка (CHR), цей метод обрано оскільки він спирається на критерій максимальної швидкодії при відсутності перерегулювання, коефіцієнти для ПІ регулятора розраховуються за наступними формулами:

$$\begin{aligned} K_p &= \frac{0.35}{a}; \\ K_I &= \frac{1.2L}{K} \end{aligned} \quad (2.23)$$

Де a – точка перетину дотичної до графіку перехідного процесу осі ординат;

L – точка перетину дотичної до графіку перехідного процесу осі абсцис;

K – пропорційний коефіцієнт замкненої системи.

Параметри необхідні для розрахунку коефіцієнтів регулятора методом СНР визначаються через графічний аналіз графіку перехідного процесу системи за ступінчатим впливом, графік перехідного процесу за керуючим впливом САУ контуру керування швидкістю стрічки наведено на рис. 2.24.

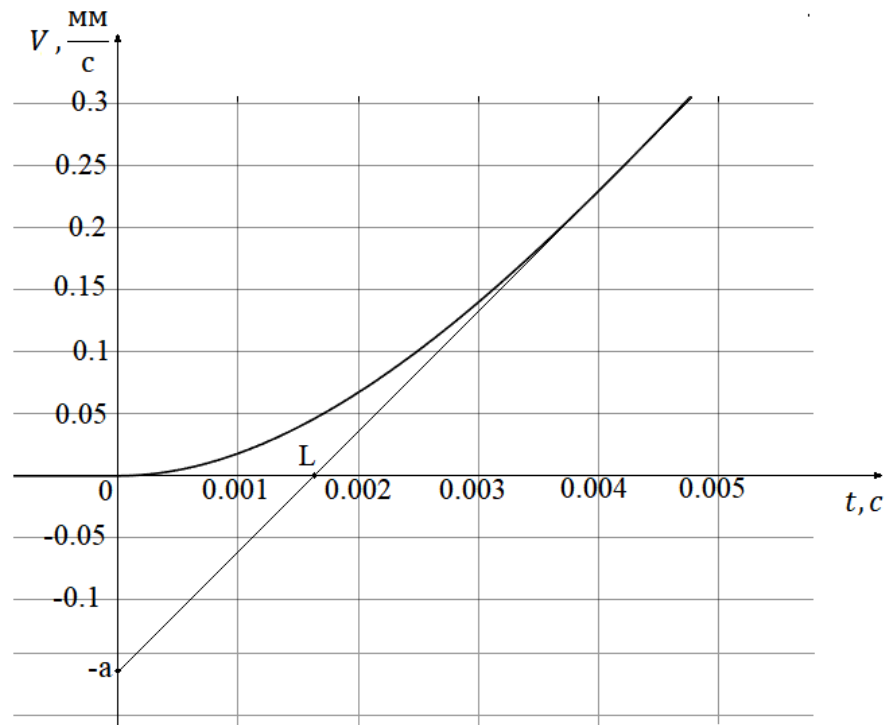


Рисунок 2.24 – Графік перехідного процесу САУ за контуром швидкості стрічки

Виходячи з графіків ПП САУ Зображених на рис. 2.24 та рис. 2.20 визначені опорні коефіцієнти ПІ регулятора у контурі керування швидкістю стрічки:

$$\begin{aligned}
 a_1 &= 0.165 \\
 L_1 &= 0.00168 \\
 K_1 &= 0.992 \\
 K_{п1} &= \frac{0.35}{a} = 2.12; \\
 K_{I1} &= \frac{1.2L}{K} = 0.002
 \end{aligned}
 \tag{2.24}$$

Після визначення опорних коефіцієнтів проводиться їх оптимізація, на першому етапі визначається оптимальне значення пропорційного коефіцієнту, графіки перехідних процесів системи з різними значеннями пропорційного коефіцієнту регулятора наведені на рис. 2.25, показники якості перехідних процесів наведені у табл 2.7.

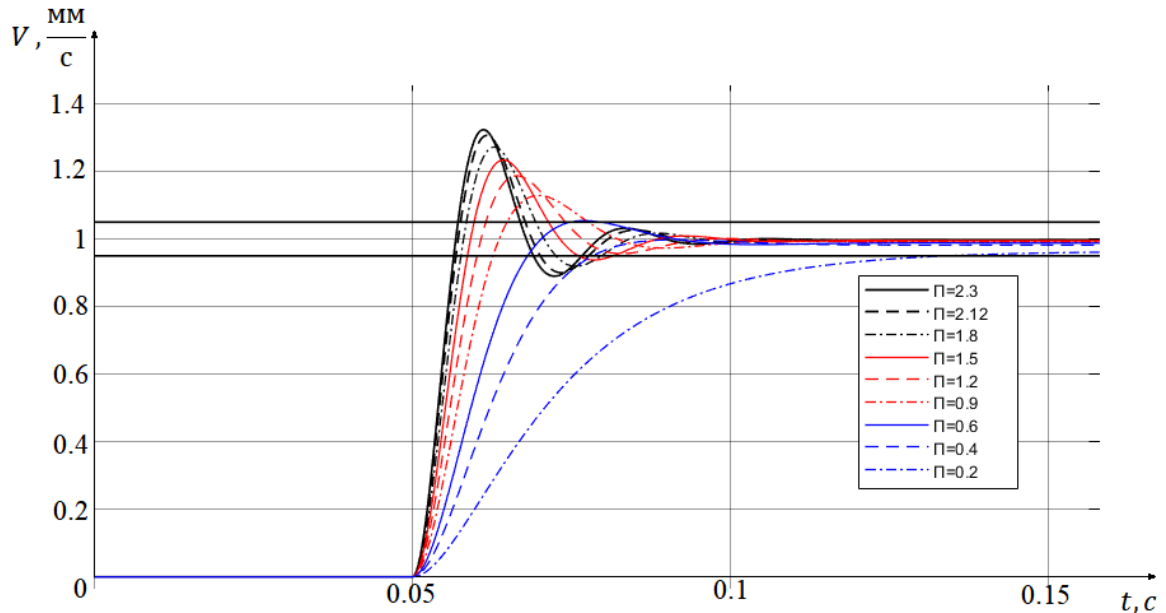


Рисунок 2.25 – Графіки перехідних процесів з різними значеннями пропорційного коефіцієнту регулятора

Таблиця 2.7 – Показники якості перехідного процесу з різними значеннями пропорційного коефіцієнту

K_p	$t_{пп}, c$	$\sigma, \%$	$\varepsilon_{ст}, \%$
2.3	0.027	32.3	0.32
2.12	0.028	30.4	0.35
1.8	0.03	27	0.41
1.5	0.032	23.3	0.49
1.2	0.024	18.7	0.62
0.9	0.027	12.8	0.081
0.6	0.03	5.4	1.21
0.4	0.028	1.5	1.83
0.2	0.089	0	3.6

Серед розглянутих коефіцієнтів обрано $K_{п1} = 0.4$ виходячи з максимальної швидкодії системи при дотримуванні вимог. Перехідні процеси оптимізації інтегральної складової регулятора наведено на рис. 2.26, показники якості перехідних процесів наведені у табл. 2.8.

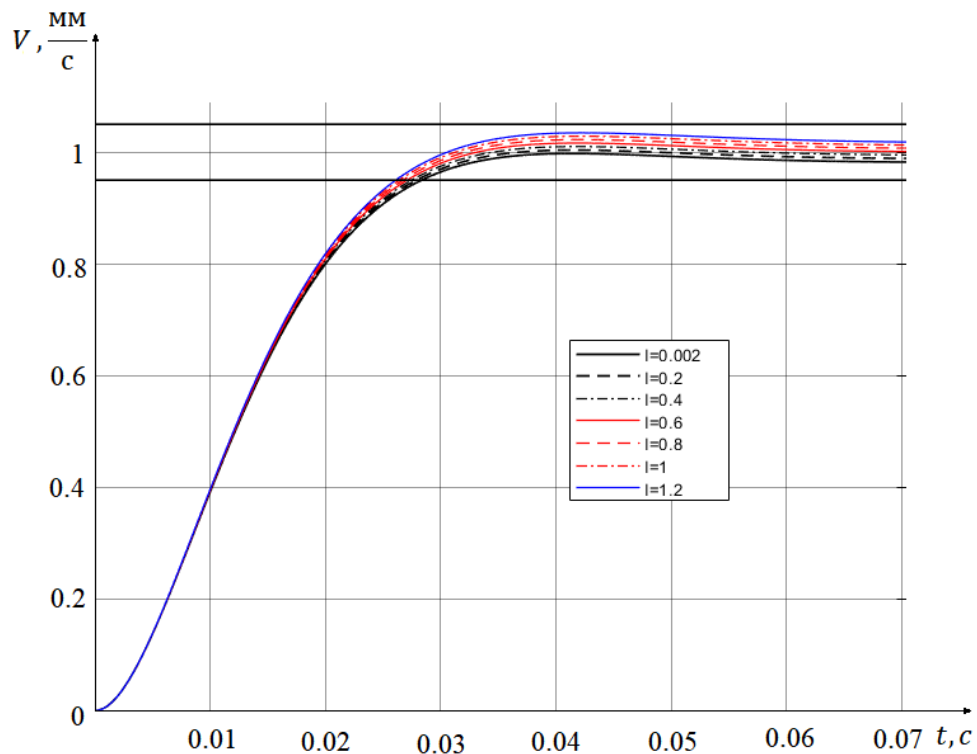


Рисунок 2.26 – Графіки перехідних процесів САУ контуру швидкості стрічки

Таблиця 2.8 – Показники якості перехідного процесу з різними значеннями інтегрального коефіцієнту

K_I	$t_{пп}, c$	$\sigma, \%$	Час зменшення похибки до $\varepsilon_{ст} < 0.1\% ,c$
0.002	0.028	1.6	218
0.2	0.028	1.57	11.131
0.4	0.027	1.47	4.815
0.6	0.027	1.49	0.382
0.8	0.027	1.48	1.225
1	0.027	1.48	1.307
1.2	0.027	1.47	1.226

Виходячи з отриманих графіків перехідних процесів системи з різними значеннями коефіцієнту інтегральної складової збільшення даного коефіцієнту призводить до зменшення часу, необхідного для компенсації статичної похибки до допустимого рівня, проте збільшення коефіцієнту також призводить до накопичення більшої інтегральної складової за час перехідного процесу, що призводить до збільшення перерегулювання. Виходячи з критерію найшвидшого зменшення статичної похибки до задовільного рівня, при умові дотримання вимог за перерегулюванням, обрано інтегральний коефіцієнт регулятора САУ контуру швидкості стрічки $K_{I1} = 0.6$.

Проведено визначення опорних коефіцієнтів регулятора САУ контуру швидкості камери, графік перехідного процесу САУ за керуючим впливом наведено на рис. 2.27.

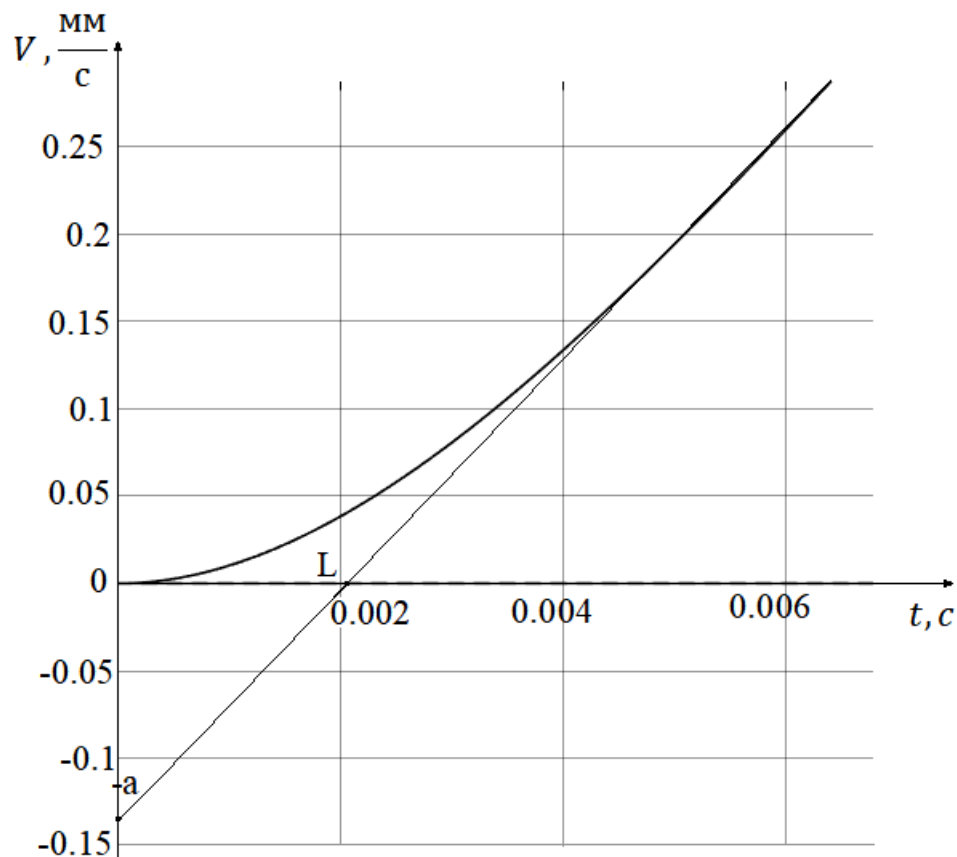


Рисунок 2.27 – Графік перехідного процесу САУ контуру швидкості камери за керуючим впливом

Виходячи з графіків ПП САУ Зображених на рис. 2.27 та рис. 2.21 визначені опорні коефіцієнти ПІ регулятора у контурі керування швидкістю камери:

$$\begin{aligned}
 a_2 &= 0.13 \\
 L_2 &= 0.0021 \\
 K_2 &= 0.987 \\
 K_{п2} &= \frac{0.35}{a} = 2.69; \\
 K_{I2} &= \frac{1.2L}{K} = 0.0025
 \end{aligned}
 \tag{2.25}$$

Спираючись на опорні коефіцієнти ПІ регулятора проведено оптимізацію пропорційної складової регулятора контуру управління швидкістю камери, графіки перехідних процесів підбору пропорційного коефіцієнту наведені на рис. 2.28, показники якості перехідних процесів наведені у табл. 2.9.

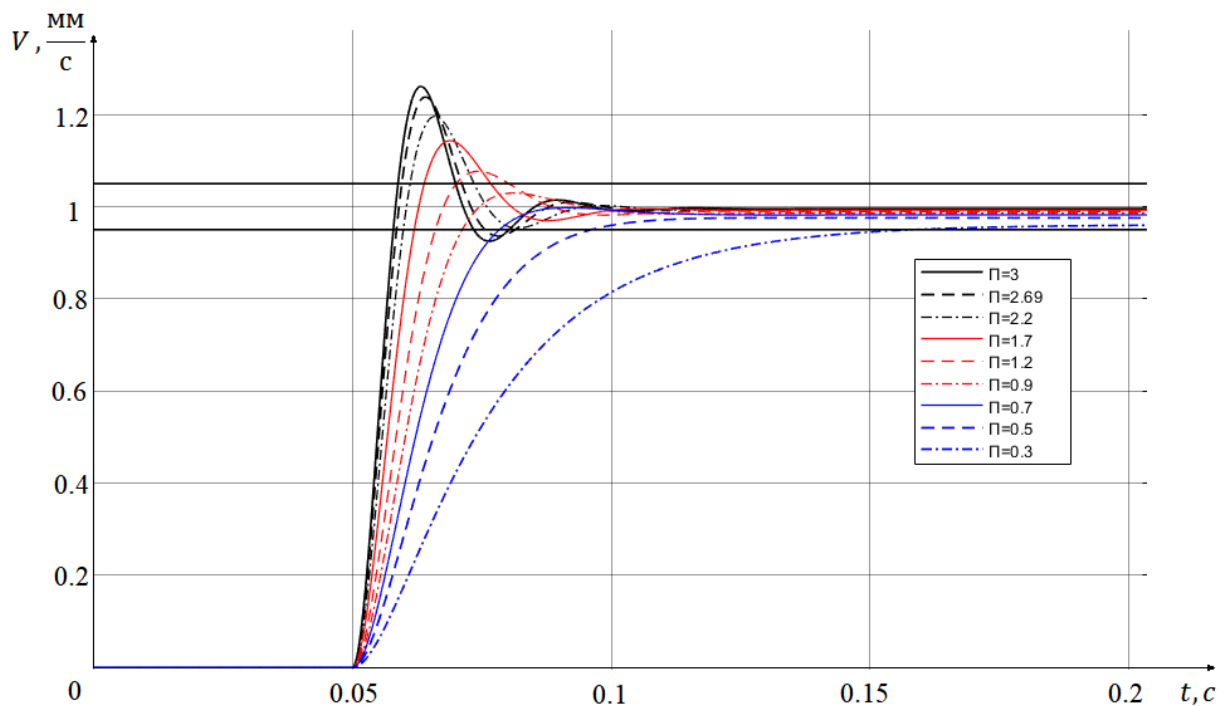


Рисунок 2.28 – Графіки перехідних процесів з різними значеннями пропорційного коефіцієнту регулятора

Таблиця 2.9 – Показники якості перехідного процесу з різними значення пропорційного коефіцієнту

K_{Π}	$t_{\text{пп}}, \text{с}$	$\sigma, \%$	$\varepsilon_{\text{ст}}, \%$
3	0.03	26	0.43
2.69	0.031	23.8	0.48
2.2	0.024	19.6	0.58
1.7	0.027	14.3	0.75
1.2	0.03	8.79	1.05
0.9	0.021	4.45	1.39
0.7	0.028	1.56	1.77
0.5	0.046	0	2.43
0.3	0.101	0	4.02

Серед розглянутих коефіцієнтів обрано $K_{\Pi 2} = 0.9$ виходячи з умови максимальної швидкодії при дотримуванні вимог за перерегулюванням. Перехідні процеси оптимізації інтегральної складової регулятора наведено на рис. 2.29, показники якості перехідних процесів наведені у табл. 2.10.

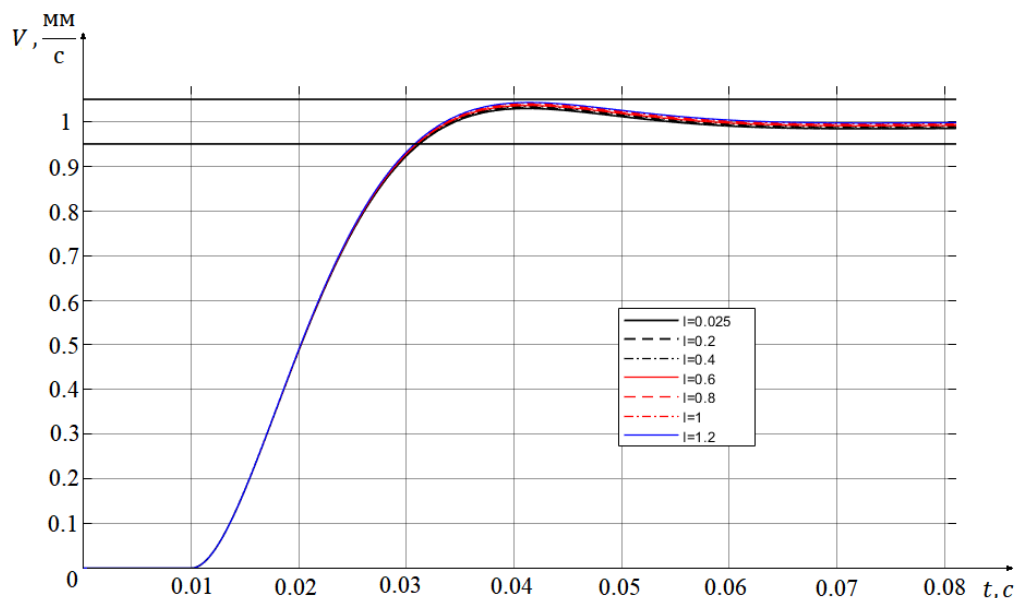


Рисунок 2.29 – Графіки перехідних процесів САУ контуру швидкості камери

Таблиця 2.10 – Показники якості перехідного процесу з різними значеннями інтегрального коефіцієнту

K_I	$t_{пп}, c$	$\sigma, \%$	Час зменшення похибки до $\epsilon_{ст} < 0.1\%$, c
0.0025	0.021	4.6	260
0.2	0.021	4.6	22
0.4	0.021	4.6	18
0.6	0.021	4.6	7.644
0.8	0.021	4.59	5.472
1	0.021	4.59	3.461
1.2	0.021	4.59	2.116

обрано інтегральний коефіцієнт регулятора САУ контуру швидкості стрічки $K_{I2} = 1.2$.

2.13 Дослідження скоригованої САУ

З метою визначення динамічних та частотних характеристик скоригованої системи, доповненої ПІ регулятором, проведено машинне моделювання у середовищі Matlab Simulink, використана машинна модель зображена на рис. 2.30, графіки перехідних процесів системи за керуючим та збурюючим впливами наведені на рис. 2.31 та рис. 2.32, графіки частотних характеристик системи зображені на рис. 2.33.

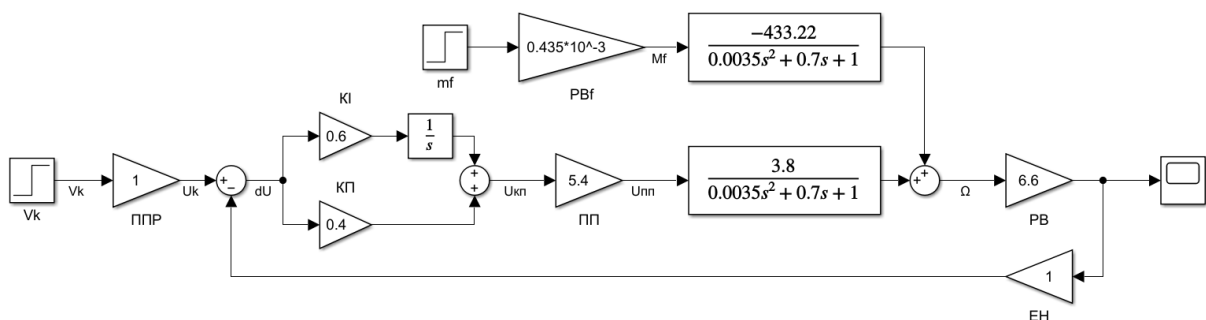


Рисунок 2.30 – Машинна модель скоригованої САУ контуру швидкості стрічки

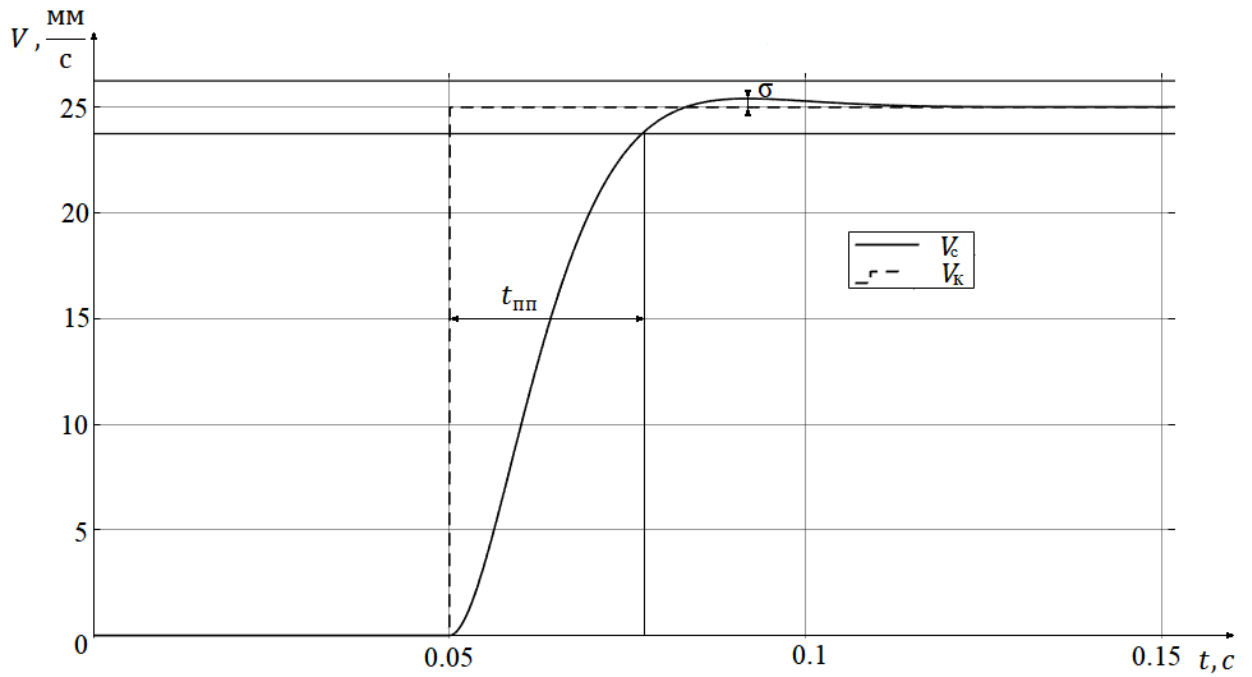


Рисунок 2.31 – Графік перехідного процесу скоригованої САУ контуру швидкості стрічки за керуючим впливом

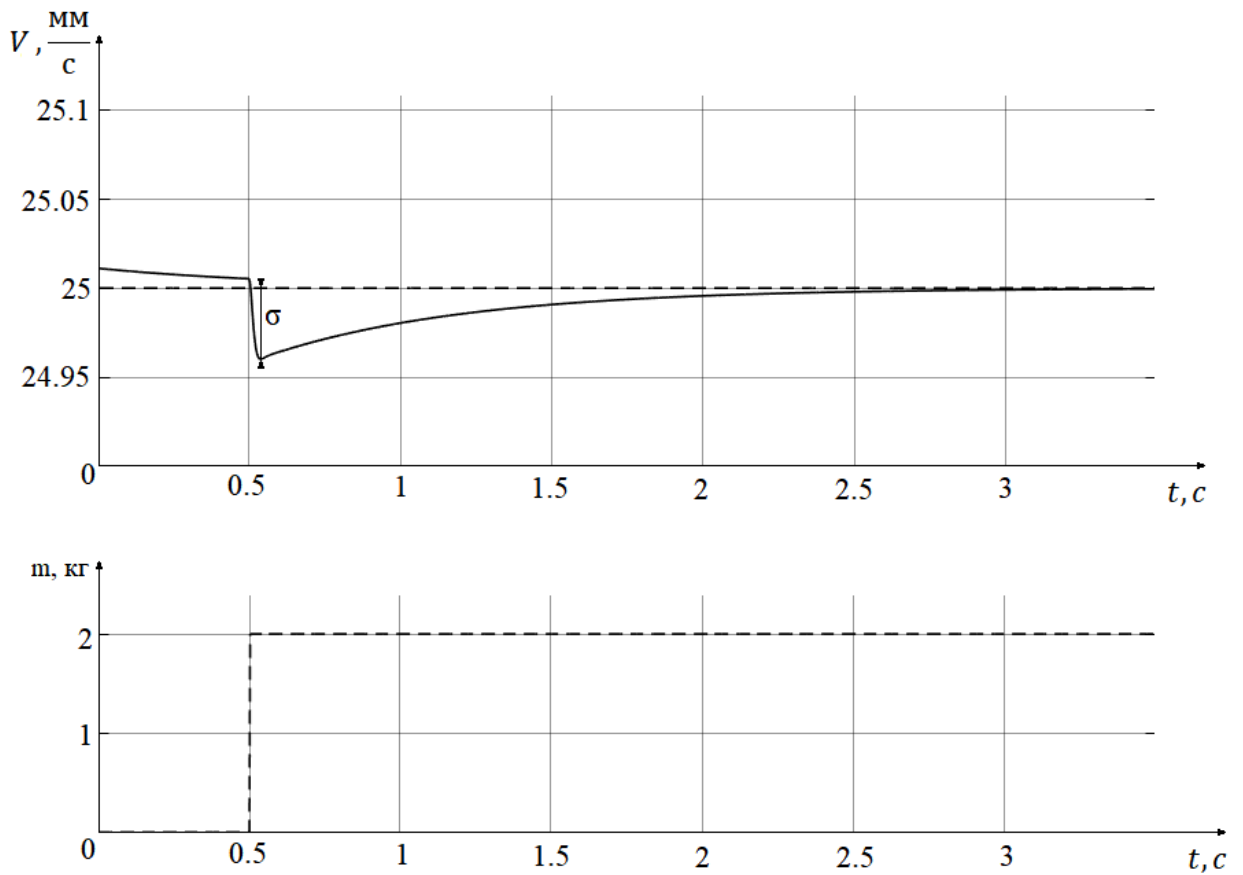


Рисунок 2.32 – Графік перехідного процесу скоригованої САУ контуру керування швидкістю стрічки за збурюючим впливом

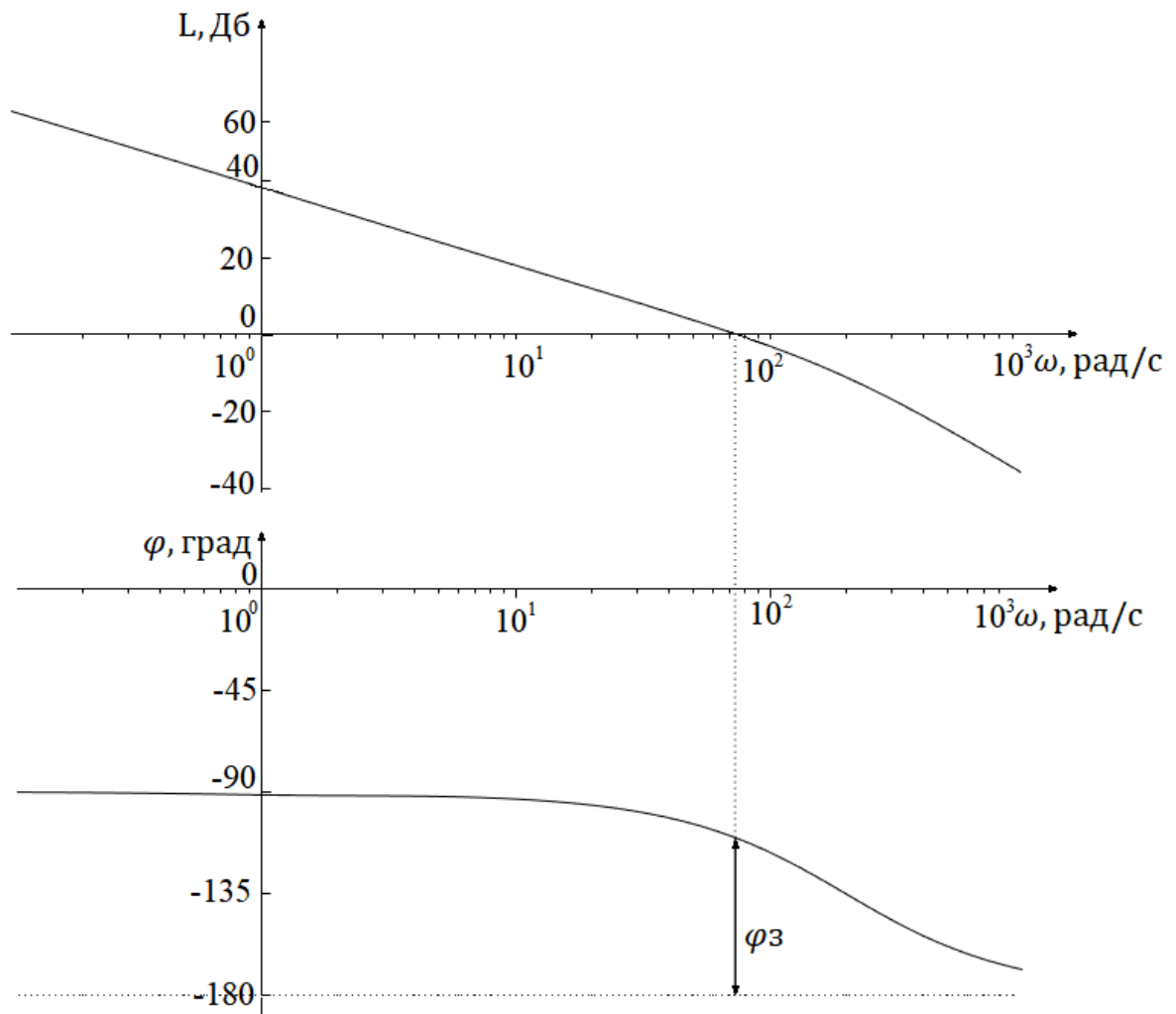


Рисунок 2.33 – Частотні характеристики скоригованої САУ контуру керування швидкістю стрічки

Показники якості функціонування системи, визначені шляхом аналізу динамічних та частотних характеристик, наведені у табл. 2.11.

Таблиця 2.11 – Показники якості скоригованої САУ контуру керування швидкістю стрічки

№ з/п	Тип впливу	$t_{пп}, c$	$\sigma, \%$	$\varepsilon_{ст}, \%$	$L_z, Дб$	$\varphi_z, ^\circ$
1	Керуючий вплив	0.027	1.49	0	∞	69.7
2	Збурюючий вплив	0	0.16	0	-	-

Проведено машинне моделювання роботи контуру управління швидкістю камери, машинна модель зображена на рис. 2.34, графік перехідного процесу системи за керуючим впливом наведено на рис. 2.35, частотні характеристики системи наведені на рис. 2.36.

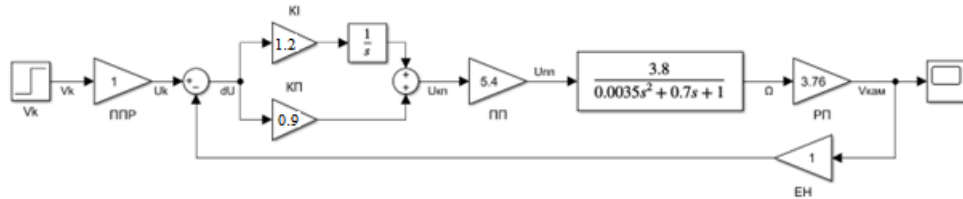


Рисунок 2.34 – Машинна модель скоригованої САУ контуру керування швидкістю камери

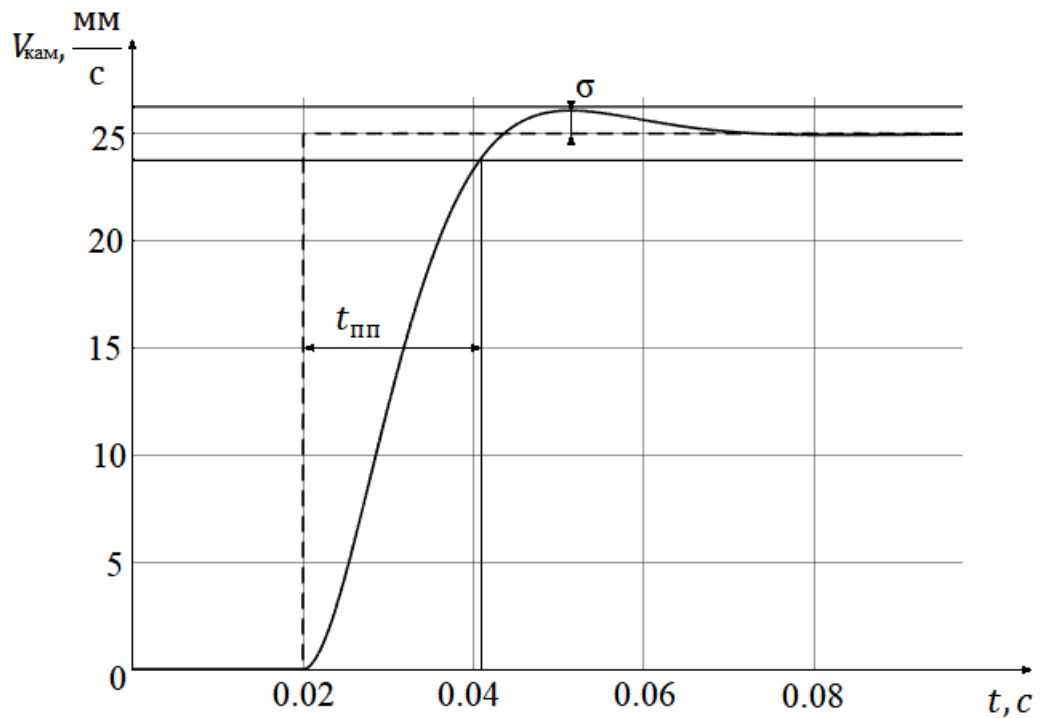


Рисунок 2.35 – Графік перехідного процесу скоригованої САУ контуру керування швидкістю камери за керуючим впливом

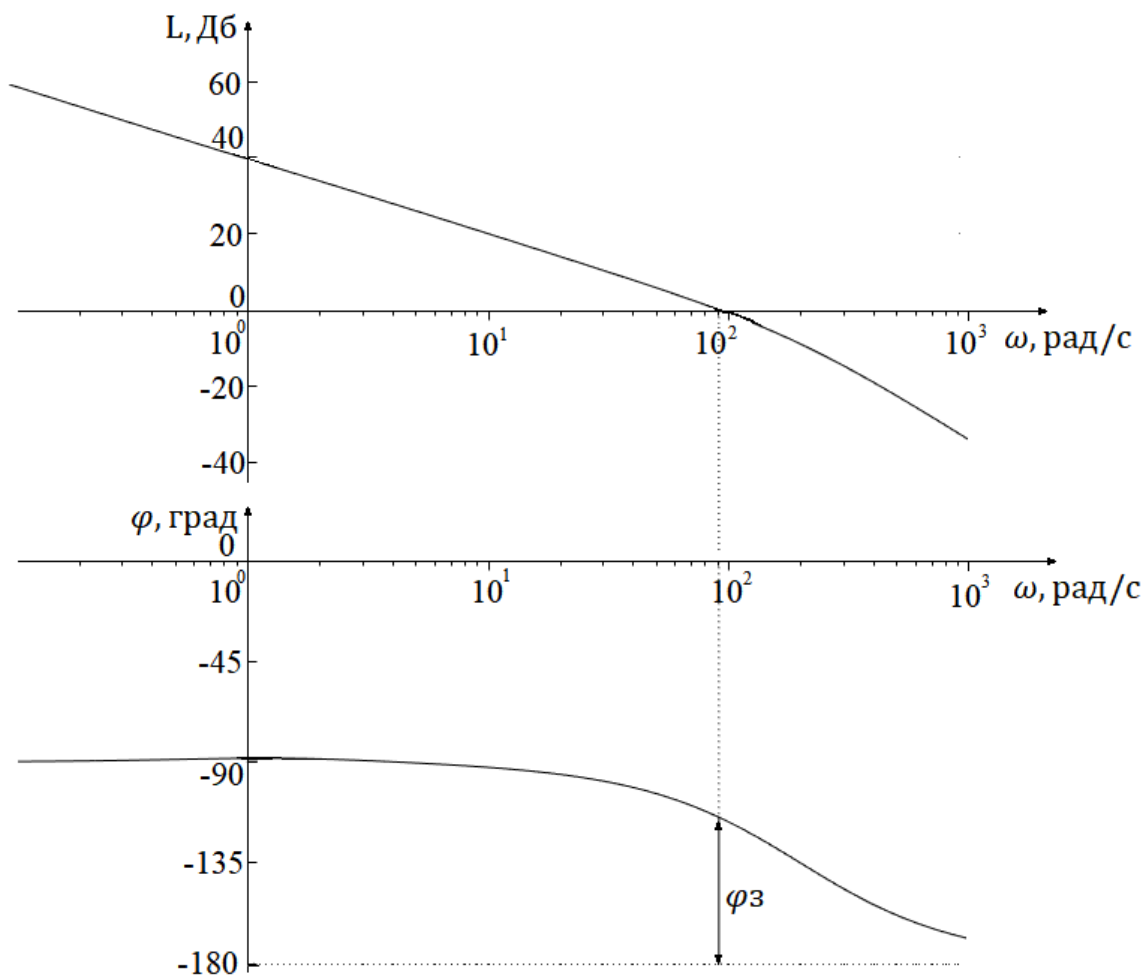


Рисунок 2.36 – Графік частотних характеристик скоригованої САУ контуру керування швидкістю камери

Показники якості контуру управління швидкістю камери скоригованої системи наведені у табл. 2.12.

Таблиця 2.12 – Показники якості скоригованої САУ контуру керування швидкістю камери

Тип впливу	$t_{пп}, c$	$\sigma, \%$	$\varepsilon_{ст}, \%$	$L_3, Дб$	$\varphi_3, ^\circ$
Керуючий вплив	0.022	4.6	0	∞	65.5

2.14 Дослідження САУ з елементами нелінійності

Лінійна модель ОАУ відповідає роботі реального об'єкту лише у робочій точці, відносно якої математична модель була лінеаризована. Характеристики реального ОАУ та його елементів є нелінійними, тому застосування до нього пристрою автоматичного управління розробленого виходячи з лінійної моделі ОАУ може не відповідати висунутим вимогам до якості функціонування.

Проведено моделювання роботи САУ з включенням нелінійних елементів з метою кращої імітації роботи реального об'єкту. До контуру керування швидкістю стрічки додано зону нечутливості 0..1В та зону насичення 0..27В до входу електродвигуна. Машинна модель САУ контуру керування швидкістю стрічки доповнена нелінійними елементами зображена на рис. 2.37, Графік перехідного процесу САУ за керуючим впливом наведено на рис. 2.38.

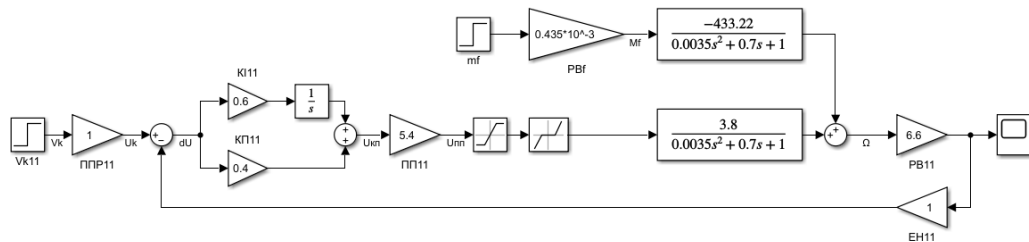


Рисунок 2.37 – Машинна модель САУ керування швидкістю стрічки з нелінійними елементами

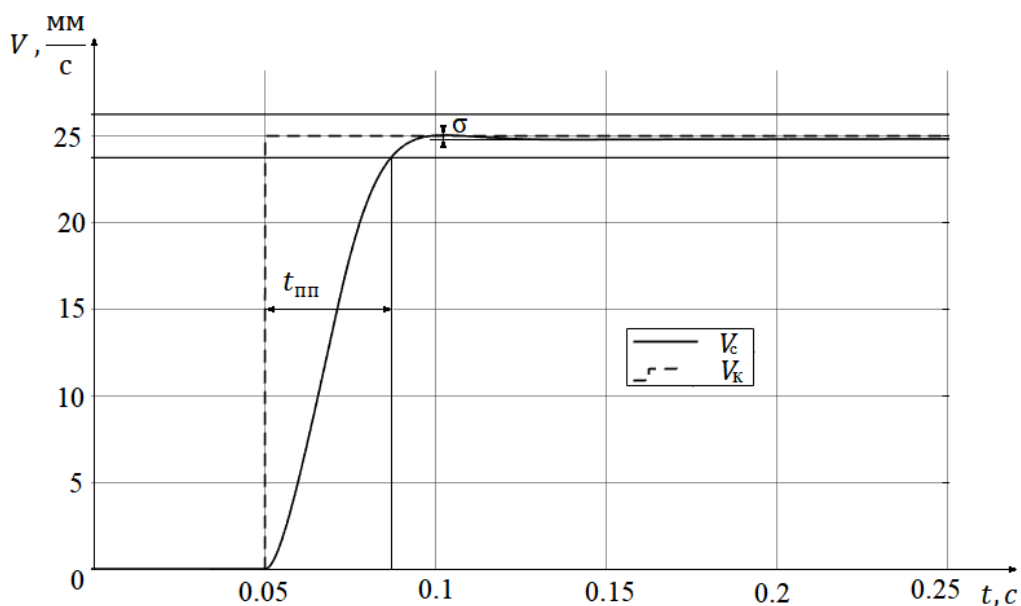


Рисунок 2.38 – Графік перехідного процесу САУ за контуром керування швидкістю стрічки з нелінійними елементами

До контуру керування швидкістю камери додано зону нечутливості $0..1В$ та зону насичення $-27..27В$ до входу електродвигуна. Машинна модель САУ контуру керування швидкістю камери з елементами нелінійності наведена на рис. 2.39, графік перехідного процесу системи з елементами нелінійності наведено на рис. 2.40, порівняння функціонування лінійних та з нелінійними елементами систем наведено у табл. 2.13.

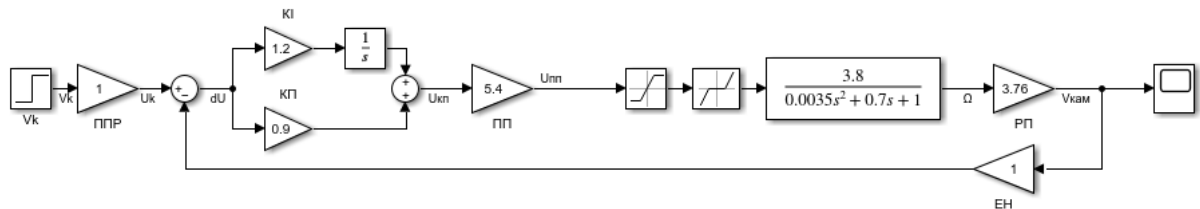


Рисунок 2.39 – Машинна модель САУ керування швидкістю камери з нелінійними елементами

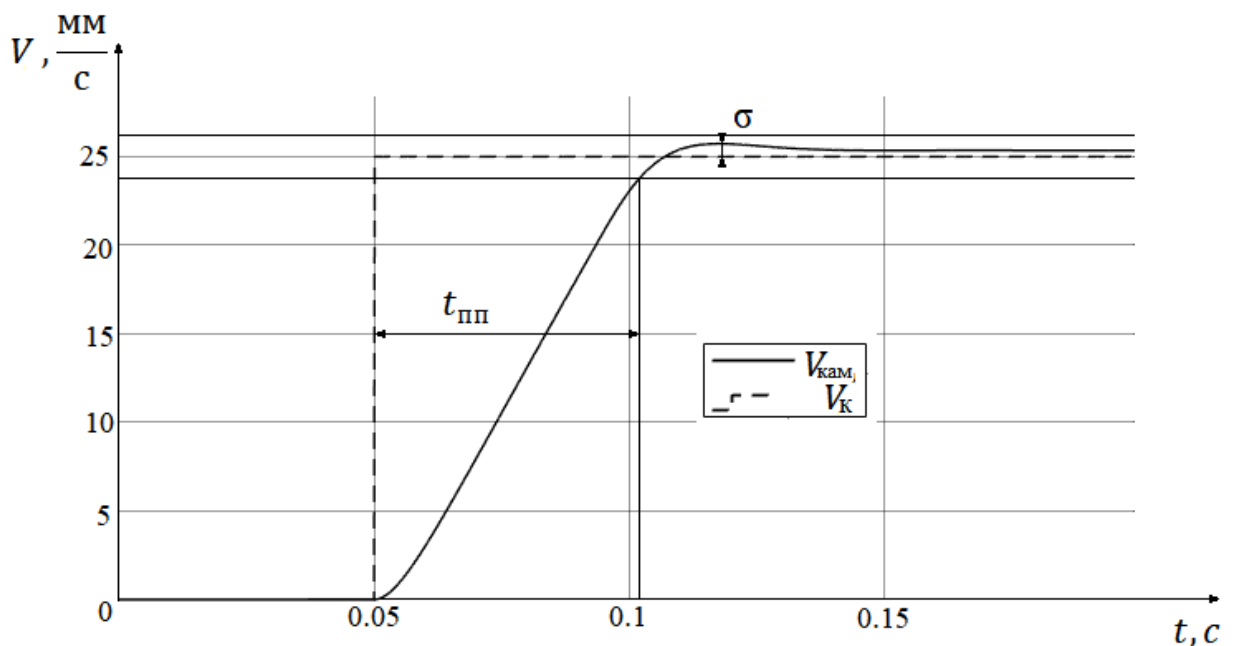


Рисунок 2.40 – Графік перехідного процесу САУ за контуром керування швидкістю камери з нелінійними елементами

Таблиця 2.13 – Порівняння характеристик систем

САУ	Тип САУ	$t_{пп}, c$	$\sigma, \%$	$\varepsilon_{ст}, \%$
Стрічки	Лінійна	0.027	1.49	0
	Нелінійна	0.037	1	0
Камери	Лінійна	0.022	4.6	0
	Нелінійна	0.052	2.92	0

Введення елементів нелінійності призвело до збільшення часу перехідного процесу оскільки вони обмежують максимальну напругу живлення двигуну, та , за рахунок зони нечутливості, зменшує цю напругу взагалі, що призводить до меншого прискорення та збільшення часу перехідного процесу, проте менша швидкість наростання сигналу відповідно зменшує перерегулювання, що зменшує навантаження на електродвигун як елемент системи.

2.15 Синтез цифрового коригуючого пристрою САУ

На дискретному мікроконтролерному модулі, що використано у даному проєкті, неможливо реалізувати неперервний лінійний алгоритм коригуючого пристрою, для коректного функціонування системи необхідно перетворити алгоритм коригуючого пристрою до дискретного виду. Виконавчий пристрій надає можливість налаштовувати період квантування коригуючого елемента тому необхідно визначити задовільні значення.

Виходячи з умови стійкості системи з дискретним регулятором, виходячи з ЛАЧХ, визначено максимальний період дискретності САУ контуру керування швидкістю стрічки, при якому вона залишатиметься стійкою:

$$T_{0 \max c} = \frac{1}{2\omega_{в c}} = 0.0079 \text{ с.} \quad (2.26)$$

де: $\omega_{в c} = 63.2 \frac{\text{рад}}{\text{с}}$ – власна частота безперервної системи, визначена виходячи з ЛАЧХ системи.

Для контуру керування швидкістю камери:

$$T_{0 \max k} = \frac{1}{2\omega_{BK}} = 0.0055 \text{ с.} \quad (2.27)$$

де: $\omega_{BK} = 90.52 \frac{\text{рад}}{\text{с}}$ – власна частота безперервної системи, визначена виходячи з ЛАЧХ системи.

Мінімальний період дискретності коригуючого елемента обумовлено показниками обраного котролеру:

$$T_{0 \min} = 0.00002 \text{ с.} \quad (2.28)$$

Проведено дослідження впливу періоду квантування коригуючого елемента на показники якості САУ контуру керування швидкістю, графіки залежності часу перехідного процесу системи та перерегулювання від періоду дискретності на рис. 2.41

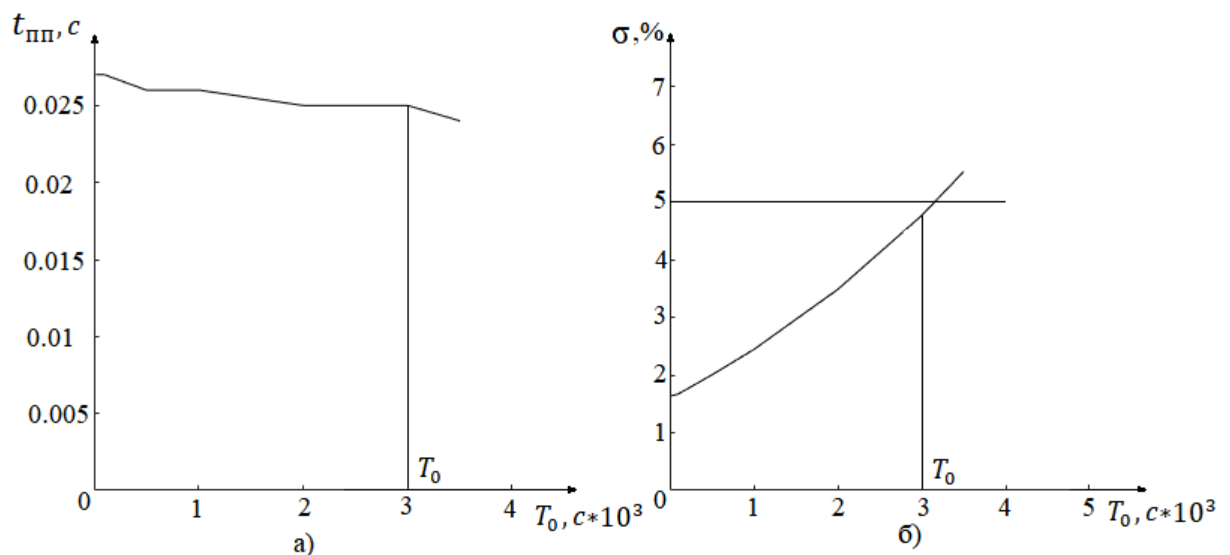


Рисунок 2.41 – а) графік залежності часу перехідного процесу системи від періоду дискретності, б) графік залежності перерегулювання системи від періоду дискретності

При збільшенні періоду дискретності першим було порушено обмеження на перерегулювання, виходячи з цього обрано значення $T_0 c = 0.003$ с яке задовольняє усім вимогам системи. Проведено дослідження САУ контуру керування швидкістю камери, графіки залежності показників якості системи від періоду квантування регулятора наведені на рис. 2.42.

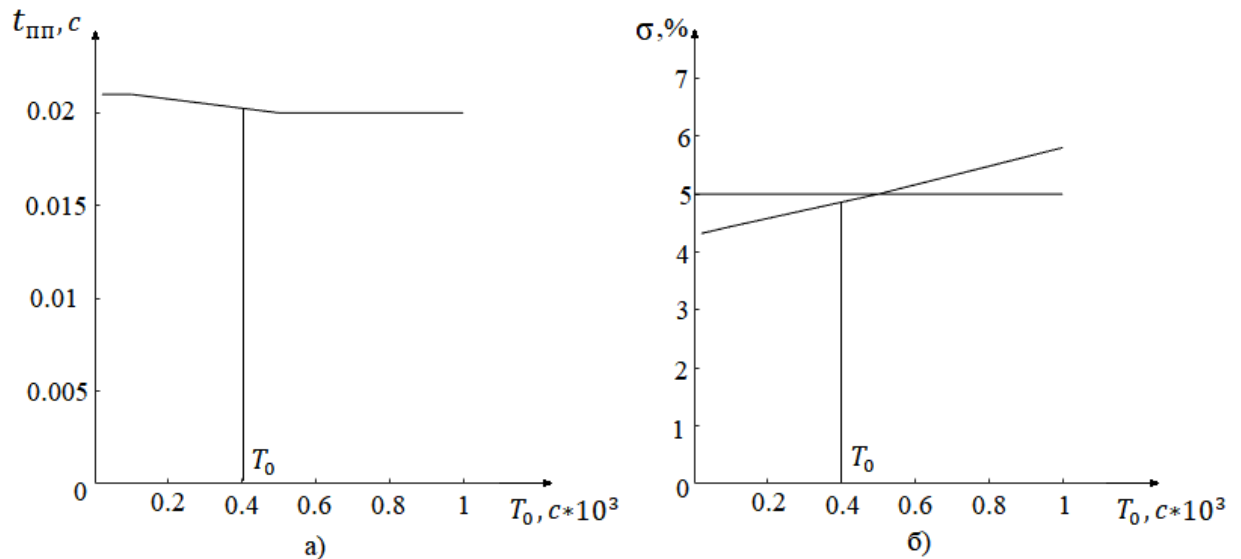


Рисунок 2.42 – а) графік залежності часу перехідного процесу системи від періоду дискретності, б) графік залежності перерегулювання системи від періоду дискретності

Виходячи з обмеження по перерегулюванню обрано найбільший допустимий період дискретності регулятора у контурі управління швидкістю камери $T_{0к} = 0.0004$ с.

Проведено машинне моделювання роботи систем з дискретними коригувальними пристроями, графік перехідного процесу САУ контуру керування швидкістю стрічки наведено на рис. 2.43, графік перехідного процесу САУ контуру керування швидкістю камери наведено на рис. 2.44, Показники якості систем надані у табл. 2.14.

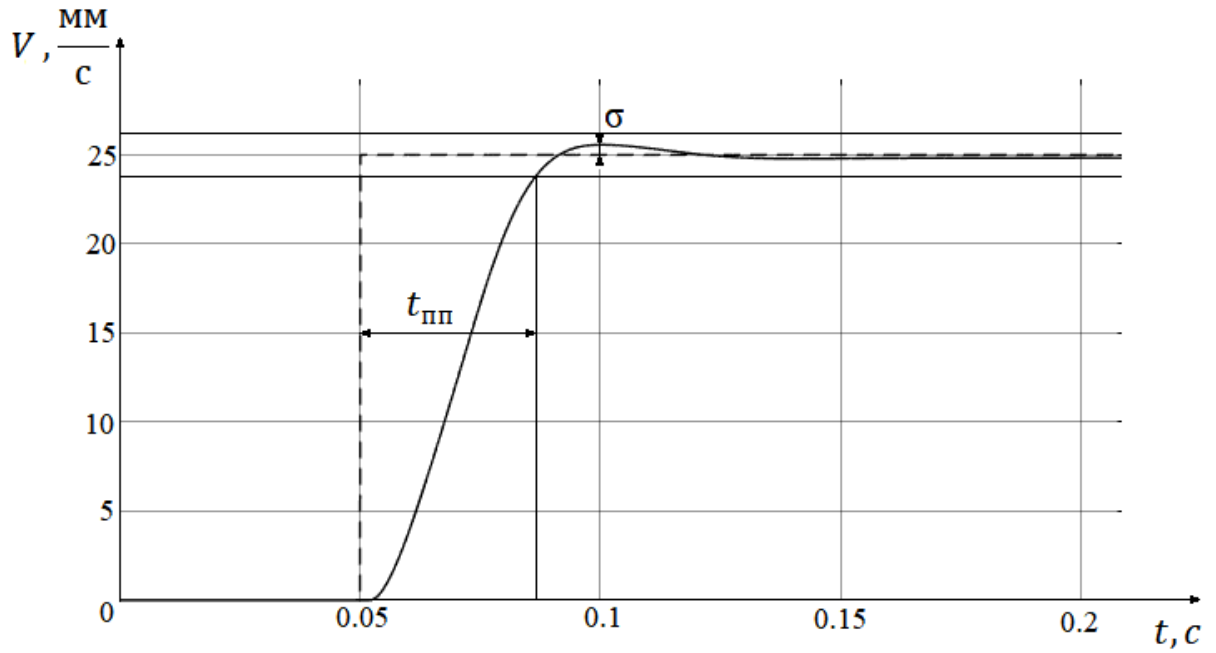


Рисунок 2.43 – Графік перехідного процесу САУ контуру керування швидкістю стрічки з дискретним регулятором

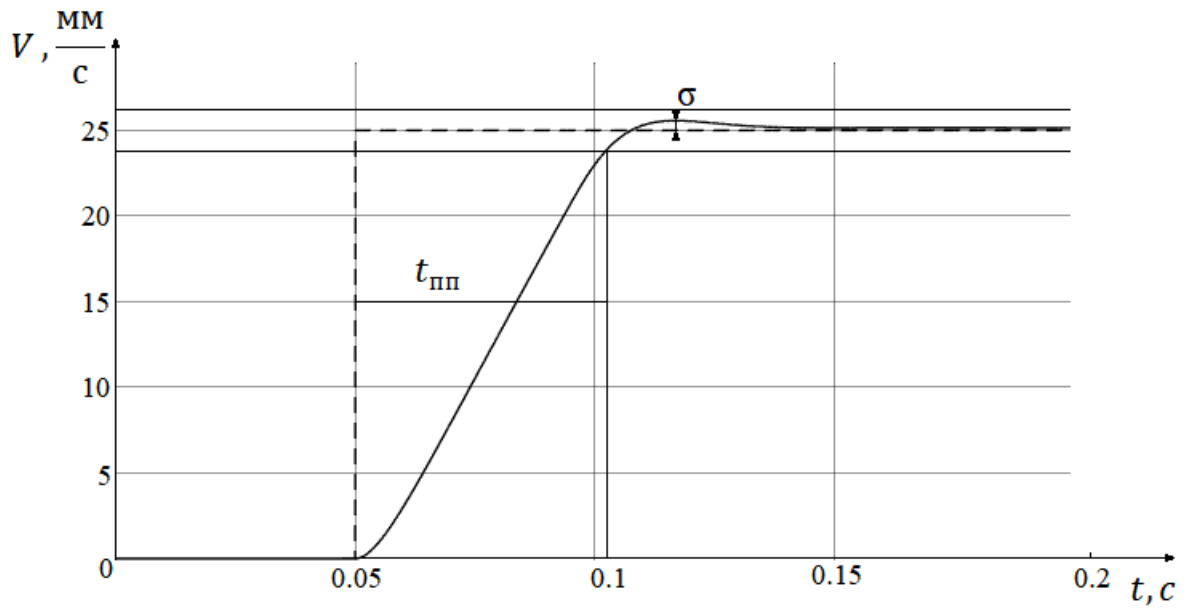


Рисунок 2.44 – Графік перехідного процесу САУ контуру керування швидкістю камери з дискретним регулятором

Таблиця 2.14 – Показники якості функціонування систем з дискретним регулятором

САУ	$t_{пп}, c$	$\sigma, \%$	$\varepsilon_{ст}, \%$
стрічки	0.037	1.48	0
камери	0.052	2.24	0

2.16 Висновки

У результаті виконання даного пункту дипломного проекту сформовану структуру САУ для контурів керування швидкістю стрічки та швидкістю камери ОАУ стенду “Стрічковий транспортер”, проведено машинне моделювання роботи системи та визначені її показники якості у нескоригованому вигляді. З метою покращення показників якості відповідно до вимог висунутих у технічному завданні проведено синтез коригувального елемента. У якості коригувального елемента обрано ПІ регулятор, через низьку кількість математичних розрахунків що необхідно виконати для визначення керуючого впливу, що суттєво при реалізації системи з мікроконтролером у якості виконавчого пристрою. Розраховано опорні коефіцієнти регулятора методом CHR та проведена їх подальша оптимізація. З метою наближення результатів моделювання роботи системи до реальних умов проведено моделювання роботи систем з додаванням нелінійних елементів що імітують зони насичення та нечутливості електродвигунів. Останнім етапом було перетворення лінійного коригувального елемента до дискретного вигляду, у якому він буде реалізований на виконавчому пристрої, було досліджено вплив зміни періоду квантування регулятора на показники якості та визначено його, виходячи з умови максимально допустимого часу квантування, який дозволить використовувати меншу частину обчислювальних потужностей виконавчого пристрою. Досліджено систему з цифровим коригуючим елементом, показники якості системи відповідають вимогам висунутим у технічному завданні.

3 КОНСТРУКТОРСЬКА ЧАСТИНА

3.1 Задачі контролеру і визначення складу вхідної інформації

Задача контролеру у складі даної системи у першу чергу полягає у виконанні ролі ОАУ, зчитуванні вихідних сигналів, що надає експериментальний стенд та керуючого впливу що формується на ПЕОМ, їх обробка для визначення відповідних значень фізичних величин та розрахунку відповідного керуючого впливу що подається на ОАУ.

Інформацію про швидкість обертання пасивного валу транспортеру та швидкість руху камери контролер отримує у вигляді імпульсів з оптичного енкодера, виходячи з кількості імпульсів можливо розраховувати положення камери чи стрічки транспортеру, а частота імпульсів пропорційна швидкості, також контролер отримує імпульси при досягненні камерою кінцевих положень на рейці. У якості керуючого впливу виступає число, що передається через послідовний порт з ПЕОМ, вихідним керуючим ОАУ сигналом є аналоговий рівень напруги на ЦАП виході контролеру. Вхідні та вихідні сигнали контролеру надані у табл. 3.1.

Таблиця 3.1 – Вхідні та вихідні сигнали контролеру

Сигнал	Напрямок	Тип
Швидкість стрічки	Вхідний	Логічні імпульси
Швидкість камери	Вхідний	Логічні імпульси
Кінцеве положення камери	Вхідний	Логічні імпульси
Задаюча швидкість	Вхідний	Цифровий пакет послідовного порту
Керуючий вплив стрічки	Вихідний	Аналоговий сигнал
Керуючий вплив камери	Вихідний	Аналоговий сигнал

3.2 Розробка алгоритмічного забезпечення

Оскільки контролер представляє собою лише цифрову обчислювальну машину, для виконання коректного Управління ОАУ до нього заноситься програма створена відповідно до його архітектури. Для відпрацювання контролеру відповідно до необхідного періоду квантування регулятора використовуються таймери контролеру що підіймають флаг необхідності

зміни значення вихідного сигналу після заданого проміжку часу. Блок-схема алгоритму програми контролера наведена на рис. 3.1.



Рисунок 3.1 – Блок-схема алгоритму програми контролера

3.3 Розробка структури цифрового контролера

Виходячи з вимог до сигналів контролера та необхідності програмувати його розроблено принципову електричну схему комунікацій контролера, схема зображена на рис. 3.2.

До складу мікроконтролерного модулю входить сам контролер AT89S8252, що забезпечує обробку сигналів системи, та розрахунок керуючого впливу, схеми TLV5638 що являє собою двоканальний ЦАП, необхідний для перетворення розрахованого значення керуючого впливу на напругу, часозадаючого ланцюгу який тактує роботу контролера з необхідною частотою та трьох роз'ємів для програмування контролера, комунікації з ПЕОМ через послідовний порт та комунікації з ОАУ.

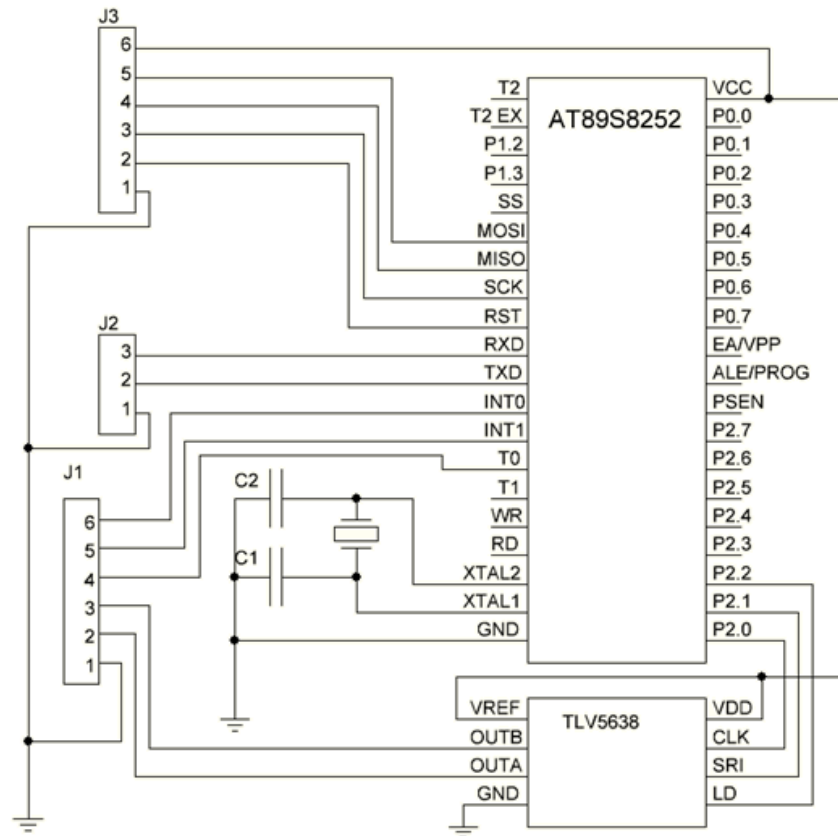


Рисунок 3.2 – Принципова електрична схема комунікацій контролера

3.4 Висновки

Виходячи з поставленої задачі та обраних апаратних засобів сформовано алгоритм роботи мікроконтролерного модулю для забезпечення функціонування САУ. Спираючись на вхідні та вихідні сигнали контролера та його особливості створена принципова електрична схема комунікацій контролера що дозволить виконувати управління ОАУ.

4 ДОСЛІДНИЦЬКА ЧАСТИНА

4.1 Задачі і методи проведення досліджень

Режим роботи транспортеру, у даному проекті, задається шляхом додавання умовних позначень на об'єкті транспортування, камера що постійно має знаходитися напроти вантажу, що забезпечується системою управління контуру швидкості камери, та постійно тримати у кадрі вантаж та знак на ньому надає відповідне зображення. Для виділення та ідентифікації знаку на вантажі необхідно провести аналіз зображення засобами технічного зору.

У даному розділі буде проводитися дослідження засобів попередньої обробки зображення, які дозволять збільшити ефективність визначення знаку на зображенні та зменшити витрати часу на це. Час ідентифікації знаку на зображенні має бути мінімальним оскільки система працює у реальному часі та значні затримки викликані визначенням знаку на вантажі зменшать її ефективність, яка відображається у кількості вантажу що може проходити через транспортер за певний проміжок часу.

У якості позначень даної системи виступають знаки зображені на рис. 4.1. У якості переваги саме цих зображень можливо виділити велику кількість тестової інформації у вільному доступі з різноманітними перешкодами для ідентифікації знаків, що дозволить якісніше налагодити систему.



Рисунок 4.1 – Умовні позначення для вантажу транспортеру

Серед спільних рис цих позначень можливо виділити круглу форму та її червоний чи синій однотонний колір, виходячи з цих ознак у подальшому буде визначатися положення цих знаків на зображенні.

Розробка системи технічного зору буде проводитися на мові програмування Python, основна обробка зображень буде проводитися використовуючи бібліотеку OpenCV, що має у своєму складі багато добре оптимізованих інструментів для роботи з цифровими зображеннями, це дозволить як значно зменшити час розробки системи так і збільшити її швидкодію.

4.2 Дослідження засобів визначення положення знаку на зображенні

Зображення з камери поступає у систему у кольоровому просторі RGB, це означає що для кожного пікселю зображення існують три цифрові характеристики, що відповідають інтенсивності складової відповідного кольору, червоного зеленого та синього, для кожної складової у форматі зображення з камери виділено один байт пам'яті, що відповідає діапазону значень 0..255. Саме зображення представляє собою матрицю пікселів розмірності $N \times M$ де кількість строк та стовпців задається використовуваною камерою.

Для ідентифікації знаку та переведення системи до відповідного режиму необхідно, в першу чергу, виділити межі цього знаку на зображенні. Дослідження методу визначення знаку буде проводитися з використанням зображення наведеного на рис. 4.2. Саме це зображення є гарним прикладом оскільки на ньому окрім самого знаку є багато інших об'єктів серед яких й такі що співпадають з кольором знаку.



Рисунок 4.2 – Початкове зображення дослідження

4.2.1 Дослідження методу виділення контурів на зображенні

В першу чергу, для визначення положення знаку, необхідно виділити його контур, контур представляє собою послідовність прилеглих одне до одного пікселів, що знаходяться на границі між різними значеннями на бінарному зображенні, контури визначаються завдяки функції бібліотеки OpenCV `findcontours()` що повертає масив де кожний елемент відповідає одному контуру та представляє собою масив з координат пікселів контуру на зображенні. При дослідженні методу бінарізації зображення у якості критерію буде виступати кількість знайдених контурів, оскільки зменшення кількості контурів які оброблюються у подальшому призведе до збільшення швидкодії системи.

Досліджено наступні методи бінарізації зображення:

- 1) Виділення границь методом `sobel`;
- 2) Виділення границь методом `prewitt`;
- 3) Виділення границь методом `canny`;
- 4) Бінарізація порогом за кольором.

Методи виділення границь застосовуються до монохромного зображення, у якому кожному пікселю відповідає лише одне значення, такі зображення представляються у сірому форматі де діапазон 0..255 значень пікселів відповідає поступовій зміні від чорного до білого кольору, а значення кожного пікселя, при переході від формату RGB, розраховується як середнє значення між трьома кольоровими складовими. Бінарізація зображення з використанням порогу за кольором не вимагає поєднання усіх каналів кольорів до одного, тому втрачається менше інформації про зображення, що дозволяє відсіяти можливі контури між пересіченнями кольорів, що не належать до знаку та їх подальша обробка призведе до затрат часу. Для більш зручного налаштування порогів бінарізації зображення переводиться з RGB формату до HSV, у якому використовуються наступні складові:

- 1) Hue – відтінок що задається у діапазоні 0..360
- 2) Saturation – насиченість кольору у діапазоні 0..100
- 3) Value – яскравість у діапазоні 0..100

Код для бінарізації зображень та підрахунку кількості контурів наведено у листингу В.3 додатку В, отримані у результаті обробки зображення наведено на рис. 4.3, кількість знайдених на зображенні контурів після обробки надана у табл. 4.1.

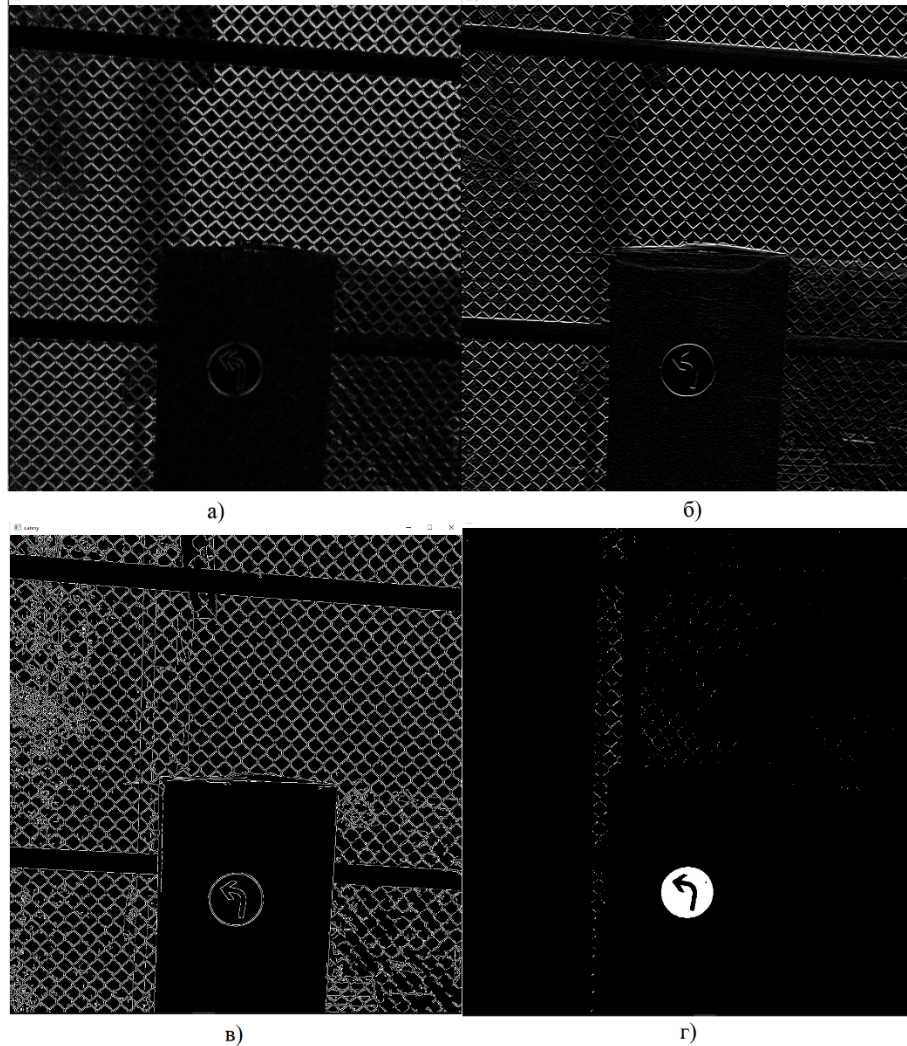


Рисунок 4.3 – Бінарізовані зображення методами
 а) sobel б) prewitt в) canny г) порогу

Таблиця 4.1 – Порівняння кількості контурів

Метод	sobel	prewitt	canny	порогу
Кількість	25148	2633	2871	765

Найефективнішим з точки зору подальшої обробки є метод бінарізації за порогом, також слід зазначити що він виконується швидше інших розглянутих методів.

4.2.2 Перевірка контуру на округлість

Після визначення наявних на зображенні контурів вони розглядаються у порядку зменшення площі контуру, оскільки наступний етап класифікації

знаку займає найбільший час обчислювань необхідно відсіяти можливі хибні зображення. Виходячи з того що усі знаки мають круглий контур. У реальному використанні алгоритму знак може знаходитися під кутом від камери чи бути нерівномірно освітлений, тому розроблено алгоритм перевірки на умовну округлість контуру, він полягає у наступних етапах:

- 1) Через точки контуру проводиться наближений до його форми еліпс
- 2) Перевіряється чи знаходиться точка еліпсу, на контурі, з допустимою похибкою, що залежить від розмірів контуру;
- 3) Якщо кількість точок еліпса наближених до контуру складає більше 75% то контур вважається умовно округлим.

4.2.3 Класифікація знаку

Після визначення можливого контуру знаку необхідно його класифікувати, визначивши конкретний знак та дії системи відповідні до нього. Класифікація знаку виконується за рахунок нейронної мережі створеної з використанням пакету Tensorflow від Google, у якості вхідних даних виступає монохромне зображення вирізане з початкового зображення камери відповідно до розмірів контуру та нормалізоване до розмірів зображень, а яких проводилося навчання мережі, вихідними даними є ймовірність класифікації зображення до одного з класів, результат з вірогідністю більше 90% вважається вірним.

Нейронна мережа побудована на основі архітектури LeNet-5 що широко розповсюджена у системах технічного зору для класифікації зображень невеликої розмірності, ця архітектура відрізняється значно меншою кількістю параметрів у порівнянні з більш сучасними аналогами, що дозволяє швидко обраховувати усі ланки мережі.

Архітектура LeNet-5 розроблена у 1998 році, та спиралася на значно менші обчислювальні потужності у порівнянні з сучасними, тому саме використання операцій згортання замість повністю пов'язаних шарів дозволило використовувати дану архітектуру до відносно великих зображень, стандартна структура нейронної мережі архітектури LeNet-5 приведена на рис. 4.4.

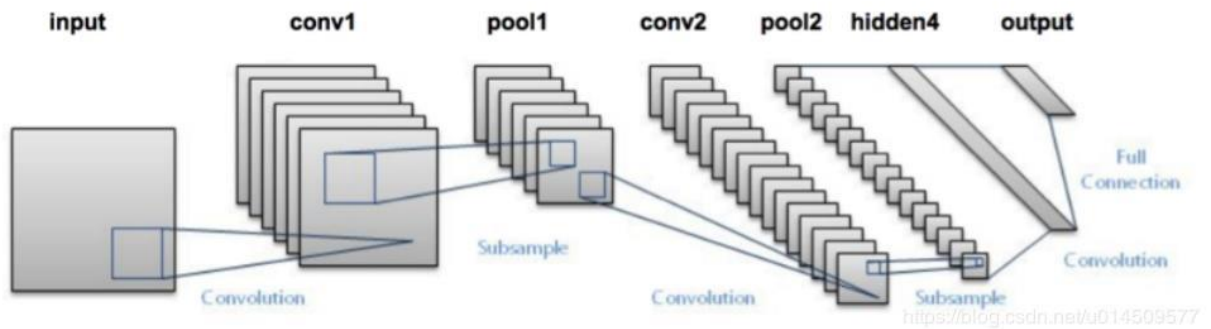


Рисунок 4.4 – структура мережі LeNet-5

Сутність даної архітектури полягає у попарному використанні шарів згортання (англ. Convolution) та поєднання (англ. Pooling). Шар згортання перебирає вхідну інформацію, що задана у вигляді матриці, маскою заданого розміру з заданими коефіцієнтами, що дозволяє виявляти на зображенні чи попередньому шарі певні риси, вихідне значення такого шару представляє собою матрицю меншої розмірності де кожен елемент відповідає сумі добутків значень попереднього шару на коефіцієнти матриці, після чого отримане значення перевіряється заданою функцією активації, якщо умова функції активації не виконується то значення змінюється на нуль, приклад наступної маски може виділяти ознаку діагональної лінії на зображенні:

$$K = \begin{bmatrix} -0.5 & -0.5 & 1 \\ -0.5 & 1 & -0.5 \\ 1 & -0.5 & -0.5 \end{bmatrix}.$$

Зазвичай для кожного шару згортання вхідна матриця оброблюється з використанням різних масок та видає кількість вихідних матриць відповідну до кількості масок.

Шар поєднання виконує функцію зменшення розміру вхідної матриці та виділення необхідної інформації, що зменшує необхідну кількість обчислень виконуваних у подальшому. При проходженні шару поєднання вхідна матриця перебирається маскою заданого розміру з кроком один, що виділяє значення на вихідну матрицю, зазвичай це пошук максимального значення у межах маски проте може застосовуватися й середнє чи мінімальне значення.

Після проходження всіх шарів згортання та поєднання значення набору отриманих матриць розглядаються як одномірний масив та проходять два повнозв'язний шари на виході яких отримується одномірний масив, кожен елемент якого відповідає одному з вихідних класів, та значення елемента

означає відповідність вхідного зображення кожному класу. Структура та параметри використаної у проект мережі наведені у табл. 4.2.

Таблиця 4.2 – Структура та параметри нейронної мережі

№	Тип шару	Розмірність виходу	Розмірність маски	Активація
0	Вхідне зображення	$28 \times 28 \times 1$	$5 \times 5 \times 1$	relu
1	Згортання	$24 \times 24 \times 32$	$5 \times 5 \times 1$	relu
2	Поєднання (max)	$12 \times 12 \times 32$	$2 \times 2 \times 1$	relu
3	Згортання	$8 \times 8 \times 2048$	$3 \times 3 \times 1$	relu
4	Поєднання (max)	$4 \times 4 \times 2048$	$2 \times 2 \times 1$	relu
5	Повнозв'язний	$256 \times 1 \times 1$	-	relu
6	Повнозв'язний	$10 \times 1 \times 1$	-	softmax

Після визначення структури мережі проводиться її навчання, суть якого полягає у зміні вагових коефіцієнтів масок, вагових коефіцієнтів повнозв'язних шарів та функцій активації, для досягнення найбільшої точності визначення на навчальній вибірці вхідних даних у яких заздалегідь відома належність до одного з вихідних класів. Навчання мережі проводилося з використанням наявних у пакеті Tensorflow засобів, у якості вхідних даних при навчанні виступала база з зображеннями знаків, використаних у роботі. Код програми що задавав структуру мережі та проводив навчання наведено у лістингу В.1 додатку В. У результаті навчання отримана нейронна мережа з точністю класифікації на валідаційному наборі даних 97%.

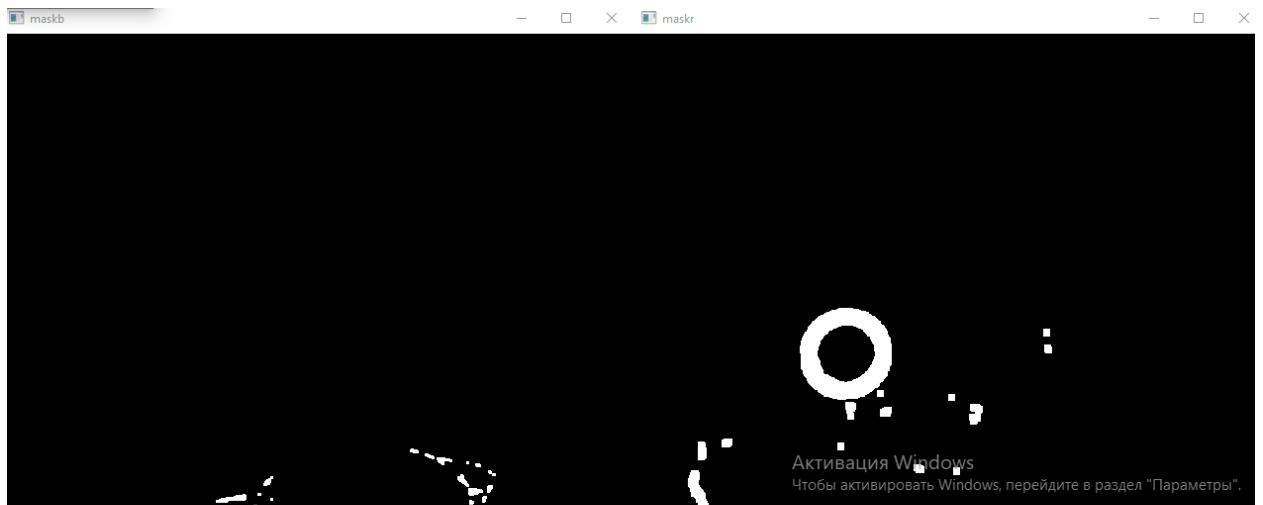
4.3 Результати досліджень

У результаті проведення досліджень визначено метод знаходження границь на зображенні що дозволяє знаходити меншу кількість контурів які проходять подальшу обробку та, відповідно, збільшити швидкодію системи. З використанням визначеного методу розроблено програмний додаток на мові програмування Python з основними використаними пакетами OpenCV та Tensorflow, що зчитує зображення з камери та проводить його багатоетапну обробку, на рис. 4.5 приведено зображення зчитане з камери.



Рисунок 4.5 – Зображення зчитане з камери

Після зчитування з камери зображення зображення бінарізується на дві маски за синім та червоним кольорами відповідно, отримані у результаті бінарізації маски зображені на рис. 4.6.



а)

б)

Рисунок 4.6 – Маски зображення а) синя б) червона

Після отримання масок на них визначаються контури, які розглядаються у порядку зменшення внутрішньої площини, відкидаються занадто малі та перевіряються на наближеність до овальної форми, визначений на даному зображенні можливий знак зображено на рис. 4.7.

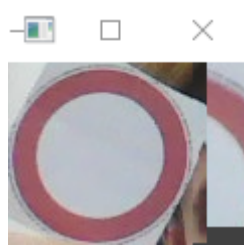


Рисунок 4.7 – Визначений можливий знак

Після визначення можливого знаку, він нормалізується до формату який приймає нейронна мережа та оброблюється нею, у результаті чого повертається інформація про схожість знаку на певний клас, при перевищенні порогу схожості у 90% знак вважається класифікованим, у інших випадках розглядається наступний контур, результат класифікації знаку наведено на рис. 4.8.

```

sign number [9]
1/1 [=====] - 0s 32ms/step
[1.18547108e-11 1.15114505e-08 6.90808591e-11 2.37633541e-11
 1.23549066e-12 6.86230961e-10 2.19043880e-12 2.38246679e-11
 2.21320900e-12 9.99998927e-01 1.10353437e-06 1.15837773e-09
 2.30199040e-10]
sign number [9]
  
```

Рисунок 4.8 – Результат класифікації знаку

На даному прикладі знак класифіковано коректно, відповідно можливо задати режим роботи системи. програмний код розробленого додатку наведено у лістингу В.2 додатку В.

6.5 Висновки

У результаті виконання даного розділу проведено дослідження засобів виділення контурів на зображенні та розроблено додаток що дозволяє класифікувати задані знаки на зображенні засобами технічного зору, що, в свою чергу, дозволяє задавати відповідний режим роботи системи автоматично, та призводить до підвищення її ефективності.

5 ЕКСПЕРИМЕНТАЛЬНО-ДОСЛІДНИЦЬКА ЧАСТИНА

5.1 Задачі і засоби виконання експериментальної розробки

Метою проведення експериментальних досліджень є перевірка працездатності системи або окремих її складових у наближених до реальної експлуатації умовах, де на об'єкт дослідження впливають всі фактори які могли бути пропущені чи відкинуті, як незначні, при побудові математичної моделі та подальшому моделюванні роботи системи. Експериментальне дослідження розробленої системи проведено у вигляді напівнатурного моделювання, даний спосіб моделювання дозволяє оцінити здатність окремого елемента системи працювати відповідно вимогам у умовах реальної експлуатації, в той час як робота інших компонентів системи моделюється.

Напівнатурне моделювання роботи системи класифікації знаків об'єктів транспортування дозволить перевірити коректність класифікації знаків, відсутність хибних розпізнавань знаків та стабільність генерації керуючого впливу відповідно до необхідного режиму роботи системи.

5.2 Опис лабораторної установки

Експериментальне дослідження буде проводитися з використанням ПЕОМ та камери з параметрами відповідними до камери лабораторного стенду “Стрічковий транспортер”, моделювання роботи контролеру для розрахунку керуючого впливу та реакції ОАУ на відповідний вплив буде проводитися у середовищі Matlab Simulink у режимі реального часу. З використанням додатку com0com на ПЕОМ створено два віртуальні СОМ-порти, які передають інформацію між собою, програма класифікації знаків надсилає до одного з них керуючий вплив з налаштуваннями відповідними реальній системі, з іншого порту інформація надходить до машинної моделі, в якій будуються графіки вихідних параметрів САУ, зображення машинної моделі САУ наведено на рис. 5.1.

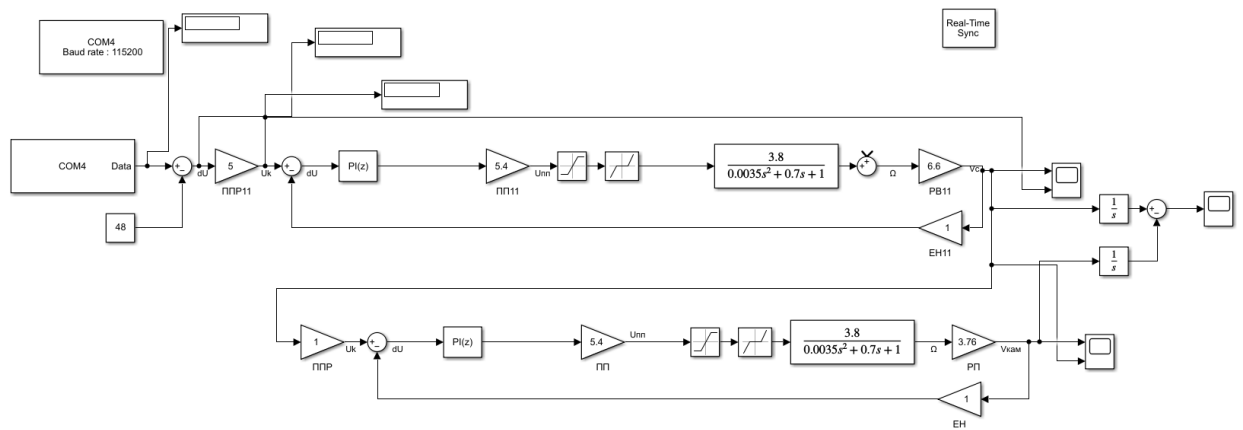


Рисунок 5.1 – Машинна модель САУ

Машинна модель зберігає інформацію про керуючий вплив, лінійну швидкість стрічки транспортера, лінійну швидкість руху камери, різницю між положенням камери та вантажу на стрічці.

5.3 Аналіз результатів проведення експериментів

Під час проведення експерименту у кадрі перебували знаки відповідні різним режимам роботи, інформація про це надана у таблиці 5.1.

Таблиця 5.1 – Стани системи при проведенні експериментального дослідження

№	Час переходу, с	Задана швидкість, мм/с
1	0	0
2	3	10
3	8	0
4	20	20
5	23	30
6	31	20
7	38	10
8	48	0

Отримані графіки параметрів системи наведені на рис. 5.2 – рис 5.4.

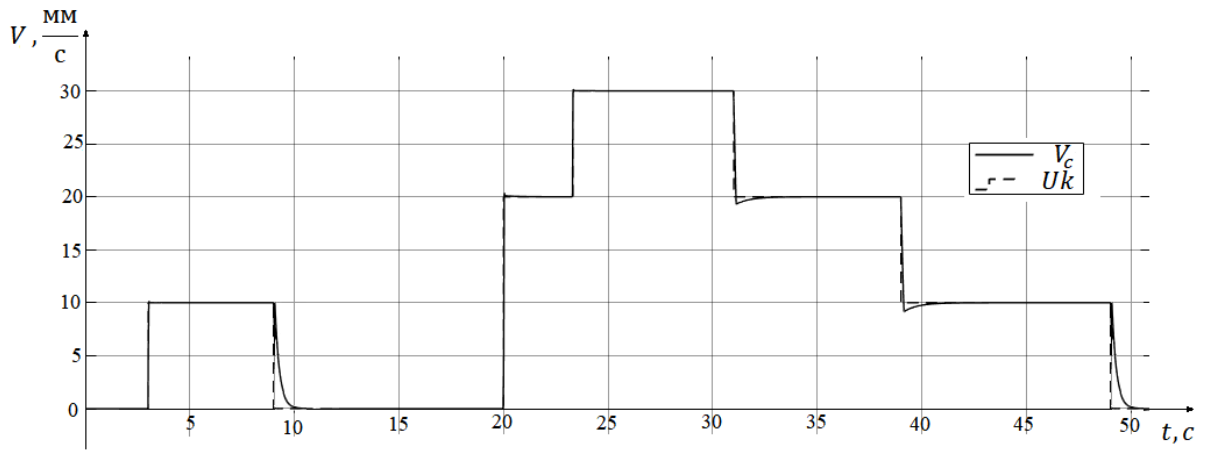


Рисунок 5.2 – Графік лінійної швидкості стрічки та керуючого впливу у ході дослідження

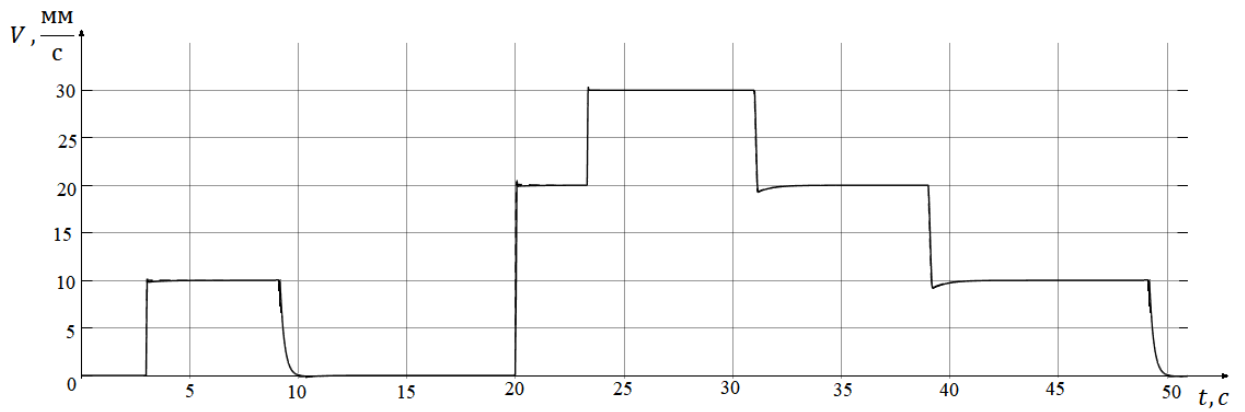


Рисунок 5.3 – Графік лінійної швидкості камери

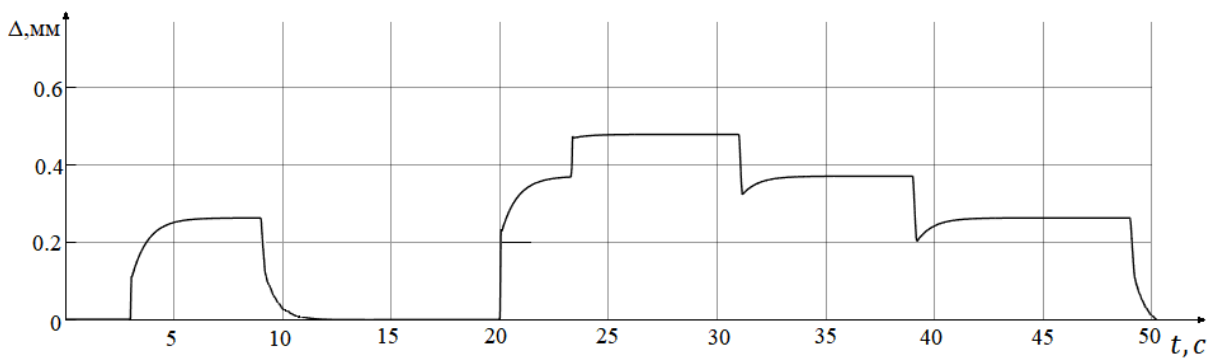


Рисунок 5.4 – Графік похибки між положенням камери та вантажу

В ході експериментального дослідження роботи системи, керуючий вплив відповідав знаку у кадрі камери, та були відсутні хибні розпізнавання. Перехідні процеси за лінійною швидкістю стрічки транспортеру, при збільшенні керуючого впливу відносно попереднього стану відповідають вимогам до системи та показникам отриманим у ході машинного моделювання, проте вони відрізняються при зменшенні керуючого впливу, це обумовлено наявністю елементів нелінійності, а саме обмеження приводу стрічки на роботу тільки з однією полярністю напруги живлення, це призвело до того що регулятор не може прискорити зменшення швидкості, проте у випадку реальної експлуатації транспортеру це не вплине на роботу системи, оскільки стрічка буде припиняти свій рух при відсутності вантажу або при поверненні камери на початкову позицію, обидва випадки надають достатньо часу для зупинки стрічки. Керуючим впливом для системи стабілізації швидкості камери є швидкість стрічки транспортеру, все перехідні процеси швидкості камери відповідають керуючому впливу. Похибка між положенням камери та вантажу на стрічці не перевищує 0.5мм що не вплине на положення знаку у кадрі камери та, відповідно, не перешкодить його класифікації.

5.5 Висновки

У ході виконання даного розділу проведено експериментальне дослідження розробленої системи автоматичного управління та програми класифікації знаків шляхом напівнатурного моделювання. Під час проведення дослідження програма класифікації передавала на модель САУ керуючий вплив відповідний до знаку у кадрі, були відсутні випадки хибної класифікації. Перехідні процеси обох контурів системи, за швидкістю стрічки та камери відповідали вимогам до якості функціонування системи.

6 ЕКОНОМІЧНА ЧАСТИНА

6.1 Основні поняття і положення

В економічній частині даного проекту розглядається розділ собівартості розробки САУ стрічкового транспортеру.

Далі розглядається собівартість розроблюваного макету даного виробу, тобто виражена в грошових одиницях сума усіх витрат підприємства на виробництво та реалізацію продукції. Собівартість продукції складається з ряду найменувань витрат. Сюди входять: витрати на основні матеріали, на комплектуючі вироби, пряма та додаткова заробітна плата, витрати на утримання та експлуатацію обладнання, утримання транспорту, а також цілий ряд загальнодержавних аналогів та відрахувань.

6.2 Опис виробу

Виріб являє собою систему автоматичного управління стрічковим транспортером за контурами швидкості стрічки та швидкості камери. Система реалізована на основі мікроконтролеру AT89S8252.

6.3 Сегментування ринку

Сегментація ринку - це процес поділу споживачів на групи з урахуванням різних принципів і факторів сегментації. У процесі сегментації виділяють сегменти ринку.

Розроблювана система поставляється заводам, підприємствам, які використовують стрічкові транспортери з необхідністю класифікації вантажу. Оскільки дані системи виробляються на замовлення, то сегментування ринку доцільно проводити за кінцевим замовником.

Основною характеристикою сегмента є ємність - кількість виробів, які можуть бути продані в рік.

Повна потреба сегменту розраховується з урахуванням конкретних особливостей товару і сегментів. Для багатьох видів товарів індивідуального і промислового призначення $S_{1пол.i}$ можна розрахувати по наступній формулі:

$$S_{1пол.} = N_1 \cdot Q_1 \cdot m_1, \quad (6.1)$$

де: N_1 – кількість підприємств споживачів виробу в i -му сегменті;

Q_1 – середня річна програма виробів в i -му сегменті, для яких буде постачатись розглянутий товар;

m_1 – кількість комплектуючих у одному виробі (1 шт.)

Результати аналізу ринку на потребу сегменту наведені у табл. 6.1.

Таблиця 6.1 – Сегментування і розрахунок повної ємності ринку

№ з/п	Виробник	Завод-виробник			
		N_1	Q_1 , шт	m_1 ,шт	S_1 ,шт/рік
1	ООО«Альбатрос»	1	70	1	70
2	АВАКС	1	40	1	40
3	SpaiTech	1	40	1	40
4	Разом	3	100	3	150

Виходячи з показників наведених у табл. 6.1. повна ємність ринку складає 150 шт./рік.

6.4 Аналіз конкурентоспроможності

Конкурентоспроможність товару – це ступінь його відповідності обраному ринку з комерційних, технічних і економічних показників, забезпечуючи можливість збуту товару на цьому ринку. Це ті характеристики, які вигідно вирізняють даний товар від товарів-конкурентів.

Проведемо аналіз конкурентоспроможності нового виробу, за допомогою узагальненого показника якості, який враховує всю систему основних одиничних показників, і характерний для даного пристрою.

У процесі аналізу ринку збуту були виявлені основні конкуренти на українському ринку, такі як PackTech, ENGITRUST. Головною відмінністю їх виробів від розроблюваної системи є відсутність контролю засобами технічного зору, які представляються окремо, після чого вся система потребує додаткового налагоджування.

При порівнянні варіантів використовується наступний порядок розрахунку.

1. Виділяються еталонні значення окремих показників якості і формується гіпотетичний варіант.

Гіпотетичний варіант - штучно створений варіант, в якому кожен показник має найкраще можливе значення.

2. Кожному показнику якості присвоюємо коефіцієнт вагомості, в сумі рівні 1.3.

Розраховується відносне значення одиничних показників шляхом порівняння P_{ij} із $P_{i_{гип}}$ з урахуванням того, що для гіпотетичного варіанти відносні показники дорівнюють одиниці ($K_{i_{гип}} = 1$), для інших $K_{ij} \leq 1$.

$$K_{ij} = \frac{P_{ij}}{P_{гип}}, \quad (6.2)$$

де: P_{ij} – абсолютні значення і-го показника для j-го варіанту;

$P_{гип}$ – абсолютні значення і-го показника для гіпотетичного варіанти

Отримані результати наведені в табл. 6.2.

Таблиця 6.2 - Аналіз конкурентоспроможності

Показник и якості	Коеф іцієн т ваго мости	Абсолютні значення показників якості				Відносні значення показників якості							
		1	2	3	4	1		2		3		4	
		Розроблю ваний	PackTech	ENGTRU	Гіпотетич ний	Ki1	Ki1*b _i	Ki2	Ki2*b _i	Ki3	Ki3*b _i	Ki4	Ki4*b _i
Точність стабілізац ії	0.4	10	8	7	10	1.00	0.4	0.80	0.32	0.70	0.28	1.00	0.4
Здатність підлаштов уватися під зміну параметрі в	0.15	10	8	7	10	1.00	0.15	0.80	0.12	0.70	0,1	1.00	0,15
Швидкод ія	0.25	9	9	6	10	0.90	0.22	0.90	0.22	0.60	0.15	1.00	0.25
Складніст ь реалізац ії	0.1	8	9	5	10	0.80	0.08	0.90	0.09	0.50	0.05	1.00	0.1
Напрацюв ання на відмову	0.1	9	7	10	10	0.90	0.09	0.70	0.07	1.00	0.1	1.00	0.1
Разом	1						0.94		0.82		0.68		1.00

Виходячи з результатів порівняльного аналізу, найбільше значення показника якості має розробляється виріб. Найближчим конкурентом є продукція PackTech. Рівень якості пропонованого виробу порівняно з цим конкурентом становить:

$$Y_{н-б} = Y_{1-2} = \frac{K_1^0}{K_2^0} = \frac{0.94}{0.82} = 1.14. \quad (6.3)$$

Порівняння виробів, присутніх на українському ринку, показало, що розробляється продукт забезпечує найбільшу точність стабілізації.

За отриманими рівнем якості можна зробити висновок, що становище на ринку вироби, буде стійким.

6.5 Розрахунок собівартості і ціни виробу

Собівартість продукції складається з ряду найменувань витрат. Сюди входять: розробники і їх денні оклади; трудомісткість робіт; основна заробітна плата; матеріали і покупні вироби.

Для розробки виробу необхідна участь наступних робочих, таких як: керівник, розробник, складальник, монтажник. Тривалість робочого місяця в середньому вважається 22 дня. Склад виконавців наведено в таблиці 6.3.

Таблиця 6.3 - Склад виконавців роботи

№ з/п	Посада	Посадові оклади, грн	
		Денні	Місячні
1	Керівник	390	8580
2	Розробник	350	7700
3	Монтажник	310	6820
4	Складальник	310	6820
5	Разом	1360	29920

Денної посади оклад складу виконавців робіт становить: керівник - 390 грн., розробник - 350 грн., монтажник - 310 грн., складальник - 310 грн.

Розрахунок трудомісткості робіт для розробки САУ представлений в таблиці 6.4.

Таблиця 6.4 – Розрахунок трудомісткості робіт

Вид робіт	Тривалість, дні	Трудомісткість, чол / дні	Керівник	Розробник	Складальник	Монтажник
Постановка задачі	1	2	+	+		
Розробка ТЗ	2	4	+	+		
Погодження та затвердження ТЗ	2	4	+	+		
Розробка ескізного проекту	4	8	+	+		
Комплектування (покупка і доставка необхідних деталей)	4	4			+	
Збірка і монтаж комплектації	3	6			+	+
Збори продукту	4	12		+	+	+
Налагодження продукту	5	5				+
Перевірка відповідності продукту	2	4	+	+		
Випробування і здача продукту в експлуатацію	5	15		+	+	+
Разом	32	64	11	20	16	17

Після чого необхідно обчислити основну заробітну плату безпосередньо виконавців розробки системи автоматичного управління, з урахуванням трудовитрат, кількості виконавців і середньоденної заробітної плати. Для цього заробітну плату працюючих виконавців і заробітну плату за день, треба помножити на кількість днів, тоді буде отримано:

$$ОЗП = \sum N_i * ЗП_{cp}, \quad (6.4)$$

де: N_i – кількість днів, відпрацьовані i -ми виконавцями за стадіями,
 $ЗП_{cp}$ – денні оклади i -х виконавців.

Таким чином, основна заробітна плата за місяць за формулою (6.5) становить:

$$ОЗП = 390 * 11 + 350 * 20 + 310 * 16 + 310 * 17 = 21520 \text{ грн.} \quad (6.5)$$

Таким чином, основна заробітна плата за місяць становить 21520 грн.

Далі потрібно зробити розрахунок додаткової заробітної плати, яка дорівнює 15% від основної заробітної плати:

$$\begin{aligned} ДЗП &= ОЗП * 15\%; \\ ДЗП &= 21520 * 0.15 = 3228 \text{ грн.} \end{aligned} \quad (6.6)$$

Після цього розрахуємо вартість покупки виробів і матеріалів, необхідних для виготовлення системи управління. Перелік покупних виробів складається з урахуванням переліку всіх комплектуючих елементів системи. Перелік покупних виробів і матеріалів заведено в таблицю 6.5. Ціни наведені в гривнях.

Таблиця 6.5 – Перелік покупних виробів і матеріалів

№ з/п	Назва матеріалів	Кількість виробів, шт.	Ціна, грн./шт.
1	Печатна плата	1	95
2	Мікроконтролер AT89S8252	1	180
3	Роз'єми	3	10
4	ЦАП	1	50
Ціна покупних виробів			355

Таким чином, вартість покупних виробів для системи позиціонування становить 355 грн.

Для виготовлення САП використовувалося певне обладнання, вартість якого наведено в таблиці 6.6.

Таблиця 6.6 - Вартість основних засобів

№ з/п	Основні засоби	Кількість,шт	Ціна, грн.
1	Робочий стіл з лещатами	1	3150
2	Лампа	1	300
3	Стілець	4	400
4	ПК	1	15000
5	Паяльна установка	1	1200
6	Випалювач	1	735
7	Всього, Сп	9	21985

Також необхідно зробити розрахунок амортизації (Ам), яка дорівнює 25% від Об.

$$A = \frac{Об * 25\%}{Дм * М} * Пр = \frac{21985 * 0.25}{22 * 12} * 32 = 666.21 \text{ грн}, \quad (6.7)$$

де: Об – вартість усього обладнання;

Дм – кількість робочих днів у місяці;

М – кількість місяців в року;

Пр – тривалість виконання роботи в днях.

Розрахунок витрат виробництва по калькуляційних статтях витрат, наведені у таблиці 6.7.

Таблиця 6.7 - Собівартість і ціна виробу за статтями

№	Статті	Сума, грн.	Примітки
1	Основна заробітна плата (ОЗП)	29920	
2	Додаткова заробітна плата (ДЗП)	4488	15% від ОЗП
3	Єдиний соціальний податок	7569.76	22%*(ОЗП+ДЗП)
4	Матеріали й куплені вироби	355	Із табл. 6.4
5	Амортизація	666.21	25%от Сп/ Нднів* Пр
6	Позавиробничі витрати	11968	40% від ОЗП
7	Собівартість (С)	54966.97	п.1+п.2+п.3...п.6
8	Прибуток (П)	10993.40	20% від С
9	Ціна без НДС (ПДВ)	65960.37	П + С
10	НДС (ПДВ)	13192.08	20% від ціни без НДС
11	Ціна з НДС (ПДВ)	79152.45	п.9 + п.10

Таким чином, ціна лічильника без урахування ПДВ становить – 65960.37 грн., а з урахуванням ПДВ – 79152.45грн.

Рентабельність продукції (норма прибутку) - це відношення загальної суми прибутку до витрат виробництва і реалізації продукції (відносна величина прибутку, що припадає на 1 грн. Поточних витрат), дорівнює:

$$P = \frac{Ц - С}{С} * 100\% = \frac{79152.45 - 54966.97}{54966.97} * 100\% = 44\%, \quad (6.7)$$

де: Ц – ціна одиниці продукції;

С – собівартість одиниці продукції.

Виробнича собівартість визначається за формулою:

$$BC = \frac{С}{КП} = \frac{54966.97}{150} = 366.45, \quad (6.8)$$

де: КП – кількість замовлених екземплярів.

Розрахуємо величину оптової ціни одного виробу ЦПП (без врахування ПДВ):

$$\text{ЦПП} = \text{ВС} \cdot \left(1 + \frac{P}{100}\right) = 366.45 * \left(1 + \frac{44}{100}\right) = 527.69 \text{грн} \quad (6.9)$$

6.6 Розрахунок точки беззбитковості

При впровадженні у виробництво приладу важливо знати чи стане цей виробничий процес рентабельним і чи буде він приносити бажаний прибуток. Для цього необхідно визначити точку беззбитковості (ТБ).

Для підтвердження стійкості проекту необхідно, щоб значення ТБ було менше значень нормальних обсягів виробництва. Чим далі від них значення ТБ (у відсотковому співвідношенні), тим стійкіше проект.

Дохід від реалізації виробів знаходимо множенням ціни одного ЦПП на кількість замовлених екземплярів КП:

$$\text{ДР} = \text{ЦПП} \cdot \text{КП} = 527.69 \cdot 150 = 79153.20 \text{ грн.} \quad (6.10)$$

Аналітичний розмір критичної програми (РКП) розраховують діленням постійних витрат $\text{РП}_{\text{ост}}\text{В}$ на різницю між ціною одного виробу ЦПП і змінними витратами, які приходяться на один виріб ($\text{З}_\text{м}\text{В}_0$), тобто:

$$\text{РКП} = \frac{\text{РП}_{\text{ост}}\text{В}}{\text{ЦПП} - \text{З}_\text{м}\text{В}_0} = \frac{12634.21}{527.69 - 282.22} = 51.46 \approx 52 \text{ од.} \quad (6.11)$$

Річні постійні витрати $\text{РП}_{\text{ост}}\text{В}$ складаються із суми наступних витрат:

$$\text{РП}_{\text{ост}}\text{В} = \text{А}_\text{м} + \text{ПВ} = 666.21 + 11968 = 12634.21 \text{ грн,} \quad (6.12)$$

де: ВУО – витрати на утримання обладнання;

$\text{А}_\text{м}$ – амортизація;

ПВ – позавиробничі витрати;

Річні змінні витрати $\text{РЗ}_\text{м}\text{В}$ складаються із суми наступних витрат:

$$\text{РЗ}_\text{м}\text{В} = \text{ВМ} + \text{ФОП} + \text{ЄСВФОП} = 355 + 34408 + 7569.76 = 42332.76 \text{ грн,} \quad (6.13)$$

де: ВМ – матеріали та куплені вироби;

ФОП – фонд оплати праці;

ЄСВФОП – єдиний соціальний внесок.

Змінні витрати, які приходяться на один виріб, визначаємо діленням річних змінних витрат на річну програму випуску виробу:

$$З_мV_0 = \frac{PЗ_мV}{КПП} = \frac{42332.76}{150} = 282.22 \text{ грн.} \quad (6.14)$$

Графічне представлення точки безбитковості представлено на рис. 6.1

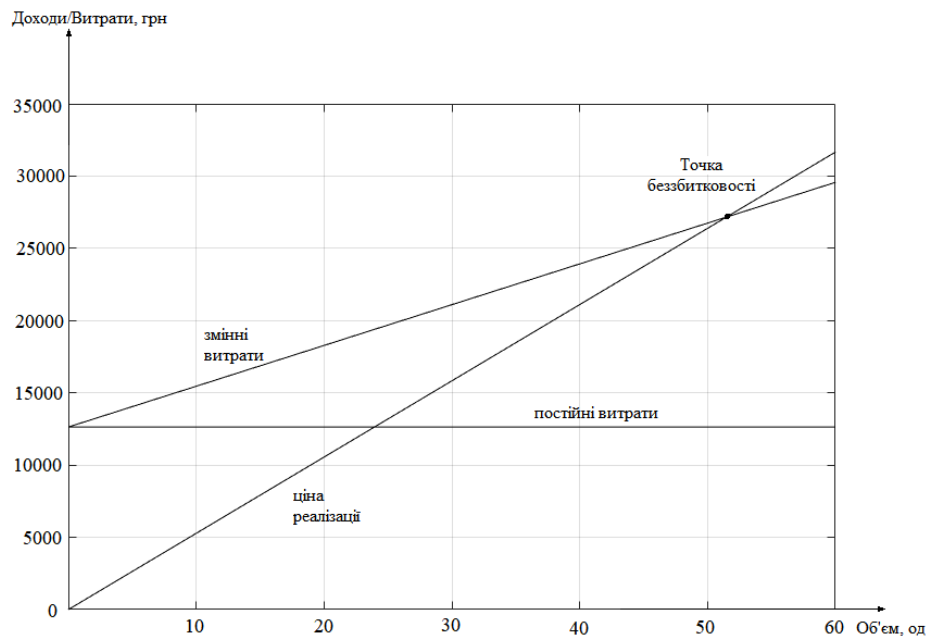


Рисунок 6.1 – Графік безбитковості

Графік безбитковості показує, що при об'ємі продаж 52 шт., дохід від реалізації продукції перевищує загальні витрати, таким чином, в зоні між ними ми отримуємо прибуток.

8.7 Висновки

В даному розділі проведено аналіз вартості та ціна системи управління. Проведено розрахунок вартості розробки регулятора з ПДВ, що становить 79152.45 грн. Визначена собівартість, що становить 54966.97 грн. Заробітна плата при розробці склала 29920 грн та відрахування у соціальний фонд становлять 7569.76грн. Проведено аналіз ринку, виходячи з якого виведена конкурентоспроможність пристрою.

Визначена точка безбитковості при реалізації проекту. Обсяг безбитковості становить 52 одиниці.

ЗАКЛЮЧЕННЯ

Метою виконання даної роботи виступала розробка САУ стрічкового транспортера за контурами швидкості стрічки транспортеру та швидкості руху камери та розробка засобу класифікації вантажу з використанням засобів ТЗ. В результаті виконання роботи синтезована САУ показники якості якої відповідають висунутим умовам до якості функціонування системи та програмний додаток на мові програмування Python що дозволяє класифікувати вантаж транспортеру за умовними позначеннями.

На першому етапі виконання проекту проведено дослідження ОАУ, що представляє собою лабораторний стенд стрічкового транспортеру, визначена математична модель що описує рух стрічки транспортеру та камери й проведено машинне моделювання ОАУ у середовищі Matlab Simulink. Сформовано структурні схеми контурів управління швидкістю стрічки та камери й проведено синтез регуляторів за ПІ законом з подальшою модифікацією коефіцієнтів. Показники якості скоригованої системи відповідають вимогам, а саме $t_{пп}=0.027$ с, $\sigma=1.49$ %, $L_3=\infty$ дБ, $\varphi_3=69.7^\circ$ для контуру швидкості стрічки та $t_{пп}=0.022$ с, $\sigma=4.6$ %, $L_3=\infty$ дБ, $\varphi_3=65.5^\circ$ для контуру швидкості камери. Проведено визначення оптимального періоду дискретності регулятора для реалізації закону управління на мікроконтролері.

Розроблено програмний додаток на мові програмування Python спираючись на пакети OpenCV та Tensorflow що дозволяє знаходити положення знаку на зображенні з камери та класифікувати його тип, відповідно до якого до САУ подається керуючий вплив.

Проведено напівнатурне моделювання роботи системи, метою якого була перевірка якості функціонування додатку класифікації знаків, робота інших складових системи моделювалася у середовищі Matlab Simulink, керуючий вплив передавався до моделі через віртуальний СОМ-порт.

У результаті аналізу отриманих в ході дослідження роботи системи параметрів виявлено, що сформована система відповідає висунутим вимогам до показників якості та додаток коректно класифікує знаки.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. What is a Belt Conveyor? – [Електронний ресурс]. URL: <https://www.spaceguard.co.uk/belt-conveyors/what-is-a-belt-conveyor/>
2. Belt Conveyors – [Електронний ресурс]. URL: https://en.wikipedia.org/wiki/Conveyor_belt
3. HSL та HSV – [Електронний ресурс]. URL: https://hmn.wiki/uk/HSV_and_HSL
4. Методи синтезу систем автоматичної стабілізації та позиціонування / А.С. Кулік, А.Є. Басова, С.М. Пасічник, Н.М. Харіна. – Харків: Нац. аерокосм. ун-т ім. М.Є. Жуковського «Харків. авіац. інт-т», 2019. – 192 с.
5. Методи моделювання об'єктів автоматичного управління / А.С. Кулік, С.М. Пасічник. . – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2018. – 166 с
6. Об'єктно-орієнтоване проектування систем керування : навч. посіб. / Л. О. Краснов, О. В. Гавриленко. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2020. – 184 с.
7. Конвеєри: застосування, вибір та інтеграція / Патрік М. Макгуайр П.Е. – Boca Raton “ CRC Press ”. – 210с.
8. Система управління конвеєрної стрічки: пат. 11132651 Японія: Ган Хоу; заявл. 18.12.2017 ;опубл. 28.09.2021.
9. Система керування накопичувальним конвеєром: пат. US6315104В1 США: Джеффри Т. Еберт; заявл. 22.04.2000 ; опубл. 13.11.2001.
10. Системи та методи ідентифікації властивостей упаковки в автоматизованій системі промислової робототехніки: пат. 10949635 США: Шон Едвардс, Ерік Нівес, Пол Хвасс, Джошуа Кертіс; заявл. 11.04.2020 ;опубл. 16.03.2021.
11. Satoshi Suzuki and others. Topological structural analysis of digitized binary images by border following. Computer Vision, Graphics, and Image Processing, 30(1):32–46, 1985.
12. Thresholding Operations using inRange – [Електронний ресурс]. URL: https://docs.opencv.org/4.x/da/d97/tutorial_threshold_inRange.html
13. Проектирование и программирование микропроцессорных систем управления [Текст]: учеб. пособие по курс. проектированию / В. Г. Джулгаков.

– Х.: Нац. аэрокосм. ун-т им. Н. Е. Жуковского «Харьк. авиац. ин-т», 2016. – 92 с.

14. ATMEGA – [Електронний ресурс]. URL: <https://avr.ru/docs/d-sheet/atmega>

15. Modelling and Simulation of a Conveyor Belt System for Optimal Productivity / Ganiyat Salawu, Glen Bright and Chiemela Onunka – South Africa: International Journal of Mechanical Engineering and Technology, 2020 – 7 p.

16. Методи синтезу систем автоматичної стабілізації та позиціонування: навч. посібник / А. Е. Басова, А. С. Кулік, С. Н. Пасичник, Н. Н. Харіна. – Харків : Нац. аерокосм. ун-т ім. М.Є. Жуковського «Харків. авиац. ін-т», 2019. – 192 с.

16. Кулік, А. С. Методи моделювання об'єктів автоматичного управління [Текст]: навч. посібник / А. С. Кулік, С. Н. Пасичник. – Харків: Нац. аерокосм. ун-т ім. М.Є. Жуковського «Харків. авиац. інт-т», 2018. – 168 с.

,

ДОДАТОК А

Таблиця А.1 – Огляд науково-технічної літератури

Країна	Назва літератури	Автор	Реферат
Україна	Методи синтезу систем автоматичної стабілізації та позиціонування	А.С. Кулік А.Є. Басова С.М. Пасічник Н.М. Харіна	В посібнику розглянуто три типових методи синтезу систем автоматичної стабілізації та позиціонування.
Україна	Методи моделювання об'єктів автоматичного управління	А.С. Кулік С.М. Пасічник	У даному посібнику були проаналізовані основні методи моделювання об'єктів управління та отримані навички по формуванню нелінійних рівнянь за допомогою рівнянь Лагранжа другого роду, а також по проведенню аналітичної лінеаризації при розкладанні у ряд Тейлора
Україна	Об'єктно-орієнтоване проектування систем керування (з використанням Python і бібліотеки OpenCV)	Л. О. Краснов О. В. Гавриленко	Запропоновано матеріали для теоретичного й практичного вивчення об'єктноорієнтованого підходу до створення програмного забезпечення систем керування, зокрема систем із технічним зором. Докладно розглянуто можливості мови Python з використанням ресурсів бібліотеки OpenCV для оброблення зображень і відеоінформації. Наведено багато прикладів.
США	Конвеєри: застосування, вибір та інтеграція	Патрік М. Макгуайр, П.Е.	Ця книга призначена для ознайомлення з широкою різноманітністю типів конвеєрів і методів застосування.

ДОДАТОК Б

Таблиця Б.1 – Патентний пошук

Код патенту	Країна, рік	Назва	Автор	Реферат
11132651	Японія, 2021	Система управління конвеєрної стрічки	Ган Хоу	Система керування конвеєрною стрічкою здатна детально визначати стан конвеєрної стрічки. Стан конвеєрної стрічки відстежується обчислювальним блоком на підставі даних індикатора, що вказує на стан пристрою стрічкового конвеєра, і даних індикатора, що вказує на середовище використання стрічки конвеєрної, встановленої на пристрої стрічкового конвеєра; щонайменше, одні дані про величину зносу гуми верхнього покриття, ударної сили, що діє на конвеєрну стрічку, що розтягує сили, індикатор, що показує стан серцевини, і індикатор, що показує стан нескінченної ділянки, що входить в сервер блоком введення; та базу даних специфікацій ременів та базу даних діапазонів допусків, в яку діапазон допусків для кожного з вхідних елементів запроваджується заздалегідь відповідно до специфікацій ременя, що зберігаються в блоці зберігання.

Продовження табл. Б.1

Код патенту	Країна, рік	Назва	Автор	Реферат
US6315104B1	США, 2001	Система керування накопичувальним конвеєром	Джеффри Т. Еберт	Система управління накопичувальним конвеєром включає кілька конвеєрів зони накопичення, кожен з яких оснащений модулем управління, з'єднаним з муфтою з електронним управлінням. Система включає один двигун, пов'язаний з кожним накопичувальним конвеєром для приводу конвеєрних роликів. Муфта зв'язує приводний ролик на кожному конвеєрі із двигуном через приводний елемент, такий як ланцюг. Модуль управління управляє роботою конвеєрів, визначаючи наявність чи відсутність предмета по дорозі одного чи кількох датчиків предмета. Коли шлях заблокований, що означає наявність предмета, модуль посилає сигнал вгору потоком і отримує сигнали з зон нижче потоку для управління роботою ролика для кожної зони.

Продовження табл. Б.1

Код патенту	Країна, рік	Назва	Автор	Реферат
10949635	США, 2021	Системи та методи ідентифікації властивостей упаковки в автоматизованій системі промислової робототехніки	Шон Едвардс, Ерік Нівес, Пол Хвасс, Джошуа Кертіс	Транспортна система сканування пакетів, яка об'єднує перше двовимірне зображення принаймні з другим двовимірним зображенням або тривимірними хмарами точок для розрахунку чи оцінки фізичних властивостей пакета та/або визначення наявності двох чи більше пакетів у купа. Зібрані дані зображення перетворюються системою за винаходом для забезпечення зворотного зв'язку в режимі реального часу з людиною-оператором, щоб він міг підтримувати розуміння продуктивності системи. Дані зображень також можна використовувати для перевірки якості інших частин систем доставки та транспортування пакетів і документування стану пакетів, які пройшли через систему.

ДОДАТОК В

Лістинг В.1 – Код програми навчання нейронної мережі

```

import tensorflow as tf
from tensorflow import keras
from keras import layers
from keras.preprocessing.image import ImageDataGenerator
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.image as img
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
import itertools
import cv2 as cv
import os
import pandas as pd

#Setting the Theme of the data visualizer Seaborn
#sns.set(style="dark",context="notebook",palette="muted")

#Tensorflow Version
#print("TensorFlow Version: "+tf.version.VERSION)
#print("Keras Version: "+tf.keras.__version__)

def display_errors(errors_index, img_errors, pred_errors, obs_errors):
    """ This function shows 6 images with their predicted and real labels"""
    n = 0
    nrows = 2
    ncols = 3
    fig, ax = plt.subplots(nrows, ncols, sharex=True, sharey=True)
    for row in range(nrows):
        for col in range(ncols):
            error = errors_index[n]
            ax[row, col].imshow((img_errors[error]).reshape((28, 28)))
            ax[row, col].set_title(" Predicted :{} True
:{}".format(pred_errors[error], obs_errors[error]))
            n += 1
            if n>=len(errors_index):
                break
        if n >= len(errors_index):
            break

def plot_confusion_matrix(cm, classes,
                          normalize=False,
                          title='Confusion matrix',
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    plt.imshow(cm, interpolation='nearest', cmap=cmap)
    plt.title(title)
    plt.colorbar()
    tick_marks = np.arange(len(classes))
    plt.xticks(tick_marks, classes, rotation=45)
    plt.yticks(tick_marks, classes)

    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]

    thresh = cm.max() / 2.

```

```

for i, j in itertools.product(range(cm.shape[0]), range(cm.shape[1])):
    plt.text(j, i, cm[i, j],
             horizontalalignment="center",
             color="white" if cm[i, j] > thresh else "black")

plt.tight_layout()
plt.ylabel('True label')
plt.xlabel('Predicted label')

def get_govna(folder1):
    X_out = np.empty((1, 28, 28, 1))
    Y_out = []
    for i in range(0, 13):
        folder2 = str(i)
        file_list = os.listdir(folder1 + "/" + folder2)
        for j in file_list:
            img = cv.imread(folder1 + "/" + folder2 + "/" + j)
            img = cv.cvtColor(img, cv.COLOR_BGR2GRAY)
            img_norm = img / 255
            img_norm = np.expand_dims(img_norm, axis=2)
            img_norm = np.expand_dims(img_norm, axis=0)
            X_out = np.append(X_out, img_norm, axis=0)
            Y_out.append(i)
    X_out = X_out[1:X_out.shape[0]]
    return X_out, Y_out

if __name__ == '__main__':
    np.random.seed(2)
    X_train, Y_train = get_govna("cropped")
    test, test_y = get_govna("test_cropped")
    #X_train = X_train.values.reshape(-1, 28, 28, 1)
    #test = test.values.reshape(-1, 28, 28, 1)
    Y_train = tf.keras.utils.to_categorical(Y_train, num_classes=13)
    random_seed = 2
    X_train, X_val, Y_train, Y_val = train_test_split(X_train, Y_train,
    test_size=0.1, random_state=random_seed)
    g = plt.imshow(X_train[0][:, :, 0])
    model = keras.Sequential()
    model.add(layers.Conv2D(filters=32, kernel_size=(5, 5), padding='Same',
    activation=tf.nn.relu, input_shape=(28, 28, 1)))
    model.add(layers.Conv2D(filters=32, kernel_size=(5, 5), padding='Same',
    activation=tf.nn.relu))
    model.add(layers.MaxPool2D(pool_size=(2, 2)))
    model.add(layers.Dropout(0.25))
    model.add(layers.Conv2D(filters=64, kernel_size=(3, 3), padding='Same',
    activation=tf.nn.relu, input_shape=(28, 28, 1)))
    model.add(layers.Conv2D(filters=64, kernel_size=(3, 3), padding='Same',
    activation=tf.nn.relu))
    model.add(layers.MaxPool2D(pool_size=(2, 2), strides=(2, 2)))
    model.add(layers.Dropout(0.25))

    model.add(layers.Flatten())
    model.add(layers.Dense(256, activation=tf.nn.relu))
    model.add(layers.Dropout(0.25))
    model.add(layers.Dense(13, activation=tf.nn.softmax))

    optimizer = tf.keras.optimizers.legacy.RMSprop(lr=0.001, rho=0.9,
    epsilon=1e-08, decay=0.0)

    model.compile(optimizer=optimizer, loss='categorical_crossentropy',
    metrics=["accuracy"])

    learning_rate_reduction =
    tf.keras.callbacks.ReduceLROnPlateau(monitor='val_accuracy',
    patience=3,

```

```
verbose=1,
factor=0.5,
```

```
min_lr=0.00001)

epochs = 30
batch_size = 112

datagen = ImageDataGenerator(
    featurewise_center=False, # set input mean to 0 over the dataset
    samplewise_center=False, # set each sample mean to 0
    featurewise_std_normalization=False, # divide inputs by std of the
dataset
    samplewise_std_normalization=False, # divide each input by its std
    zca_whitening=False, # apply ZCA whitening
    rotation_range=10, # randomly rotate images in the range (degrees, 0 to
180)
    zoom_range=0.1, # Randomly zoom image
    width_shift_range=0.1, # randomly shift images horizontally (fraction of
total width)
    height_shift_range=0.1, # randomly shift images vertically (fraction of
total height)
    horizontal_flip=False, # randomly flip images
    vertical_flip=False) # randomly flip images

datagen.fit(X_train)

if (tf.test.is_built_with_cuda() == True):
    print("CUDA Available.. Just wait a few moments...")
else:
    print("CUDA not Available.. May the force be with you.")

history = model.fit_generator(datagen.flow(X_train, Y_train,
batch_size=batch_size),
                             epochs=epochs, validation_data=(X_val,
Y_val),
                             verbose=2, steps_per_epoch=X_train.shape[0]
// batch_size
                             , callbacks=[learning_rate_reduction])

fig, ax = plt.subplots(2, 1)
ax[0].plot(history.history['loss'], color='b', label="Training loss")
ax[0].plot(history.history['val_loss'], color='r', label="validation loss",
axes=ax[0])
legend = ax[0].legend(loc='best', shadow=True)

ax[1].plot(history.history['accuracy'], color='b', label="Training
accuracy")
ax[1].plot(history.history['val_accuracy'], color='r', label="Validation
accuracy")
legend = ax[1].legend(loc='best', shadow=True)

# Predict the values from the validation dataset
Y_pred = model.predict(X_val)
# Convert predictions classes to one hot vectors
Y_pred_classes = np.argmax(Y_pred, axis=1)
# Convert validation observations to one hot vectors
Y_true = np.argmax(Y_val, axis=1)
# compute the confusion matrix
confusion_mtx = confusion_matrix(Y_true, Y_pred_classes)
# plot the confusion matrix
plot_confusion_matrix(confusion_mtx, classes=range(10))

errors = (Y_pred_classes - Y_true != 0)

Y_pred_classes_errors = Y_pred_classes[errors]
```

```

Y_pred_errors = Y_pred[errors]
Y_true_errors = Y_true[errors]
X_val_errors = X_val[errors]
# Probabilities of the wrong predicted numbers
Y_pred_errors_prob = np.max(Y_pred_errors, axis=1)

# Predicted probabilities of the true values in the error set
true_prob_errors = np.diagonal(np.take(Y_pred_errors, Y_true_errors,
axis=1))

# Difference between the probability of the predicted label and the true
label
delta_pred_true_errors = Y_pred_errors_prob - true_prob_errors

# Sorted list of the delta prob errors
sorted_dela_errors = np.argsort(delta_pred_true_errors)

# Top 6 errors
most_important_errors = sorted_dela_errors[-6:]

# Show the top 6 errors
display_errors(most_important_errors, X_val_errors, Y_pred_classes_errors,
Y_true_errors)

# predict results
results = model.predict(test)

# select the indix with the maximum probability
results = np.argmax(results, axis=1)

results = pd.Series(results, name="Label")
submission = pd.concat([pd.Series(range(1, 28001), name="ImageId"),
results], axis=1)

submission.to_csv("cnn_mnist_datagen.csv", index=False)
#model.save("neirocal")

```

Лістинг В.2 – Код програми класифікації знаків

```

import numpy as np
import cv2 as cv
import math
import os
import tensorflow as tf
import serial
from tensorflow import keras

def check_ellipse (cnt):
    ell=cv.fitEllipseDirect(cnt) # обрисовываает контур эллипсом, возвращает
размеры осей эллипса и его поворот по часовой
    #ell[0]=[int(ell[0][0]),int(ell[0][1])]
    #ell[1] = [int(ell[1][0]), int(ell[1][1])]
    #ell[2] = int(ell[2])
    cnt_rotated= rotate_contour (cnt,int(ell[2])-
90,int(ell[0][0]),int(ell[0][1])) #поворачивает контур против часовой и
смещает его так чтобы центр обводящего эллипса был в 0 0
    #dltx=int(ell[1][0]/50)+1 # погрешность в которой может быть точка
контурa относительно эллипса
    #dltx = int(ell[1][1] / 50)+1
    if (ell[1][1]/ell[1][0]>=3) or (ell[1][0]/ell[1][1]>=3):
        return -1
    dltx=int(ell[1][1]/20)

```

```

dltx=int(ell[1][0]/20)
h_size_x=int(ell[1][0]/2) #длина осей в длину полуосей
h_size_y= int(ell[1][1] / 2)
cnt_points_in_ell=int(0) # счетчик точек прошедших проверку
cnt_points_checked = int(0)
cnt_size=len(cnt)
x_prev=0
y_prev=0
#test_img=np.zeros((int(ell[0][1]*3),int(ell[0][0]*3),3),np.uint8)
#test_img=cv.ellipse(test_img, (int(ell[0][0]),int(ell[0][1])),
(int(ell[1][0]/2),int(ell[1][1]/2)),int(ell[2]), 0, 360, (255,255,255),1)
#cv.drawContours(test_img, cnt, -1, (255, 0, 0), 2)
#cv.drawContours(test_img,cnt_rotated, -1, (0, 255, 0), 2)

for deg in range(0,361): # проверяем на все 360 градусов, типа при
    большом круге не будет слишком много точек (и так корды считать проще чем при
    переборе всех x y)
    rad=math.radians(deg)
    x=int(h_size_x*math.sin(rad))
    y= int(h_size_y*math.cos(rad))#-h_size_y
    x=x+int(ell[0][1])
    y=y+int(ell[0][0])
    #test_img[x,y]=[0,0,255]
    if (x_prev==x) and (y_prev==y): #если круг маленький и разные
        градусы попадают в одну точку то не проверяем
            continue
    else:
        cnt_points_checked+=1
        for i in range (0,cnt_size):
            if (((cnt_rotated[i,0,1]>=x-dltx) and
(cnt_rotated[i,0,1]<=x+dltx)) and (cnt_rotated[i,0,0]==y)) or
            (((cnt_rotated[i,0,0]>=y-dlty) and (cnt_rotated[i,0,0]<=y+dlty)) and
            (cnt_rotated[i,0,1]==x)): # перебираем точки контура пока одна из них не
            попадет в квадрат с погрешностями вокруг точки эллипса
                cnt_points_in_ell+=1
                break
            x_prev=x
            y_prev=y

#cv.imshow("current contour", test_img)

if cnt_points_in_ell/cnt_points_checked >=0.75: # проверка валидности
контура
    return ell[2]
else:
    return -1

def rotate_contour (cnt,angle,cntx,cnty):
    cnt_norm = cnt - [cntx, cnty]

    coordinates = cnt_norm[:, 0, :]
    xs, ys = coordinates[:, 0], coordinates[:, 1]
    thetas, rhos = cart2pol(xs, ys)

    thetas = np.rad2deg(thetas)
    thetas = (thetas - angle) % 360
    thetas = np.deg2rad(thetas)

    xs, ys = pol2cart(thetas, rhos)

    cnt_norm[:, 0, 0] = xs
    cnt_norm[:, 0, 1] = ys

    cnt_rotated = cnt_norm + [cntx, cnty]

```



```

cnt_rotated = cnt_rotated.astype(np.int32)
return cnt_rotated

#cnt_rotated = cnt_norm.astype(np.int32)
#return cnt_rotated

def cart2pol(x, y):
    theta = np.arctan2(y, x)
    rho = np.hypot(x, y)
    return theta, rho
def pol2cart(theta, rho):
    x = rho * np.cos(theta)
    y = rho * np.sin(theta)
    return x, y

if __name__ == '__main__':

    using_simulink =1

    serialport = serial.Serial("COM3",115200)
    recognize_counter = 0

    model = tf.keras.models.load_model('neirocal')
    cap=cv.VideoCapture(0)

    while True:

        ret,img=cap.read()
        cv.imshow("cam", img)
        cv.waitKey(1)

        #clahe = cv.createCLAHE(clipLimit=3., tileGridSize=(8, 8))
        #lab = cv.cvtColor(img, cv.COLOR_BGR2LAB) # convert from BGR to LAB
color space
        #l1, a, b = cv.split(lab) # split on 3 different channels
        #l2 = clahe.apply(l1) # apply CLAHE to the L-channel
        #lab = cv.merge((l2, a, b)) # merge channels
        #img = cv.cvtColor(lab, cv.COLOR_LAB2BGR) # convert from LAB to BGR

        #img_gray=cv.cvtColor(img,cv.COLOR_RGB2GRAY)
        hsv = cv.cvtColor(img,cv.COLOR_BGR2HSV)

        #hsv = cv.blur(hsv, (5, 5))
        mask_blue=cv.inRange(hsv, (102,80,10), (128,255,255))

        #cv.imshow("maskb",mask_blue)

        mask_blue=cv.erode(mask_blue,None,iterations=1)
        mask_blue = cv.dilate(mask_blue, None, iterations=1)

        cv.imshow("maskb", mask_blue)
        cv.waitKey(1)

        mask_red2 = cv.inRange(hsv, (0, 60, 0), (7, 255, 255))
        mask_red1 = cv.inRange(hsv, (160, 60, 0), (180, 255, 255))
        mask_red=np.bitwise_or( mask_red1,mask_red2)

        #cv.imshow("maskr", mask_red)

        mask_red = cv.erode(mask_red, None, iterations=1)
        mask_red = cv.dilate(mask_red, None, iterations=3)

```

```

cv.imshow("maskr", mask_red)
cv.waitKey(1)

contours_blue, hierarchy=cv.findContours
(mask_blue,cv.RETR_TREE,cv.CHAIN_APPROX_NONE)
contours_red, hierarchy = cv.findContours(mask_red, cv.RETR_TREE,
cv.CHAIN_APPROX_NONE)

contours=contours_blue + contours_red

if contours:
    #print(contours)
    contours = sorted(contours, key=lambda x: cv.contourArea(x),
reverse=True)
    # print(len(contours))
    for i in range(0,len(contours)):
        if len(contours[i])>=100:
            #print(contours[i])
            angl = check_ellipse(contours[i])
            if angl != -1:
                #print("1")

left_top=[min(contours[i][:,0,0]),min(contours[i][:,0,1])]
          right_bottom = [max(contours[i][:, 0, 0]),
max(contours[i][:, 0, 1])]
          sign =
img[left_top[1]:right_bottom[1],left_top[0]:right_bottom[0]]

        cv.imshow("sign", sign)
        cv.waitKey(1)
        sign2 = cv.resize(sign, (28,28),
interpolation=cv.INTER_AREA)
        sign2 = cv.cvtColor(sign2, cv.COLOR_BGR2GRAY)
        sign_norm = sign2 / 255
        sign_norm = np.expand_dims(sign_norm, axis=2)
        sign_norm = np.expand_dims(sign_norm, axis=0)
        predictions=model.predict(sign_norm)
        predictions=predictions[0]
        print(predictions)
        if max(predictions)>=0.85:
            index =
np.where(predictions==max(predictions))[0]
            print ("sign number "+str(index))
            if using_simulink ==1:
                match index:
                    case 0:
                        serialport.write(b"0")
                    case 1:
                        serialport.write(b"0")
                    case 2:
                        serialport.write(b"0")
                    case 3:
                        serialport.write(b"0")
                    case 4:
                        serialport.write(b"0")
                    case 5:
                        serialport.write(b"0")
                    case 6:
                        serialport.write(b"0")
                    case 7:
                        serialport.write(b"0")
                    case 8:
                        serialport.write(b"2")
                    case 9:
                        serialport.write(b"4")

```

```

        case 10:
            serialport.write(b"6")
        case 11:
            serialport.write(b"0")
        case 12:
            serialport.write(b"0")

        break
    print ("cant recognize")
    recognize_counter +=1
    if (recognize_counter>=300) and (using_simulink == 1):
        recognize_counter = 0
        serialport.write(b"0")

    #cv.drawContours (img, contours, -1, (255,0,255), 2)
    #cv.imshow("contours", img)

cap.release
cv.destroyAllWindows ()

```

Лістинг В.3 – Код для дослідження бінарзації зображень

```

import numpy as np
import cv2 as cv
if __name__ == '__main__':
    img = cv.imread('ref/5.jpg')
    img_hsv = cv.cvtColor(img, cv.COLOR_BGR2HSV)
    img_gray = cv.cvtColor(img, cv.COLOR_BGR2GRAY)
    edges_sobel = cv.Sobel(img_gray, cv.CV_16S, 1, 1, ksize=3, scale=1, delta=0,
borderType=cv.BORDER_DEFAULT)
    edges_sobel = cv.convertScaleAbs(edges_sobel)
    edges_prewitt = cv.filter2D(img_gray, ddepth=0, kernel= np.array([[1,1,1],[0,0,0],[-1,-1,-1]]))
    edges_canny = cv.Canny (img_gray, 50,200)
    mask_blue = cv.inRange(img_hsv, (105, 100, 30), (125, 255, 255))
    cv.imshow("sobel", edges_sobel)
    cv.waitKey()
    cv.imshow("prewitt", edges_prewitt)
    cv.waitKey()
    cv.imshow("canny", edges_canny)
    cv.waitKey()
    cv.imshow("hsv", mask_blue)
    cv.waitKey()
    contours, hierarchy = cv.findContours (cv.inRange (edges_sobel, 90,255), cv.RETR_TREE,
cv.CHAIN_APPROX_NONE)
    print (len(contours))
    contours, hierarchy = cv.findContours (cv.inRange (edges_prewitt, 90,255),
cv.RETR_TREE, cv.CHAIN_APPROX_NONE)
    print (len(contours))
    contours, hierarchy = cv.findContours (edges_canny, cv.RETR_TREE,
cv.CHAIN_APPROX_NONE)
    print (len(contours))
    contours, hierarchy = cv.findContours (mask_blue, cv.RETR_TREE,
cv.CHAIN_APPROX_NONE)
    print (len(contours))

```