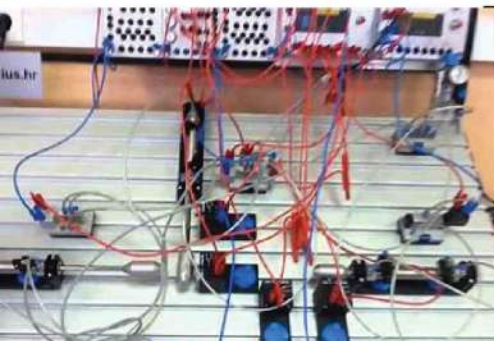


І. Ш. Невлюдов, О. В. Токарева,
О.М. Цимбал, А.І. Бронніков

НАВЧАЛЬНИЙ ПОСІБНИК

З ПІДГОТОВКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА

для здобувачів вищої освіти денної і заочної форми навчання
спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології»
та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка»
освітньої програми «Системна інженерія»



KITAP

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

**НАВЧАЛЬНИЙ ПОСІБНИК
З ПІДГОТОВКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА**

для здобувачів вищої освіти денної і заочної форми навчання
спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології»
та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка»
освітньої програми «Системна інженерія»



УДК 519.685

H15

Рецензенти:

Л. І. Нефьодов, завідувач кафедри автоматизації та комп'ютерно-інтегрованих технологій, ХНАДУ, доктор технічних наук, професор;

Ю. В. Ромашов, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки, ХНУРЕ, доктор технічних наук, доцент;

О. А. Янковський, доцент кафедри електронно-обчислювальних машин, ХНУРЕ, кандидат технічних наук, доцент.

*Рекомендовано Вченою радою
Харківського національного університету радіоелектроніки
(протокол № 8/7 від 6.07.2023 р.)*

Авторський колектив:

І. Ш. Невлюдов, О. В. Токарева, О.М. Цимбал, А.І. Бронніков.

Навчальний посібник з підготовки кваліфікаційної роботи бакалавра для H15 здобувачів вищої освіти денної і заочної форм навчання спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія»: Навчальний посібник / І. Ш. Невлюдов, О. В. Токарева, О.М. Цимбал та ін. – Харків : Видавництво Іванченка І.С., 2023. – 218 с.

ISBN 978-617-8332-16-7.

DOI: 10.30837/978-617-8332-16-7.

У навчальному посібнику подано загальні вимоги до організації підготовки і виконання кваліфікаційної роботи бакалавра для здобувачів вищої освіти денної і заочної форми навчання спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія», рекомендації щодо оформлення пояснювальної записки, а також стислий довідковий матеріал щодо змісту розрахункової частини роботи з прикладами.

Рекомендується здобувачам освіти першого (бакалаврського) рівня освіти денної і заочної форми навчання спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія».

УДК 519.685

ISBN 978-617-8332-16-7

DOI: 10.30837/978-617-8332-16-7

© І. Ш. Невлюдов, О. В. Токарева

О.М. Цимбал, А.І. Бронніков, 2023.

ЗМІСТ

Скорочення та умовні позначки.....	5
Загальні положення.....	7
1 Методичні рекомендації до кваліфікаційної роботи бакалавра.....	9
1.1 Мета та задачі кваліфікаційної роботи бакалавра.....	9
1.2 Організація виконання кваліфікаційної роботи бакалавра.....	9
1.3 Захист кваліфікаційної роботи.....	15
1.4 Критерії оцінювання кваліфікаційної роботи.....	18
2 Вимоги до оформлення кваліфікаційної роботи.....	23
2.1 Вимоги до оформлення пояснювальної записки.....	23
2.1.1 Загальні вимоги.....	23
2.1.2 Формули та рівняння.....	25
2.1.3 Таблиці.....	27
2.1.4 Рисунки та графіки.....	28
2.1.5 Переліки.....	29
2.1.6 Посилання.....	30
2.1.7 Додатки.....	30
2.2 Вимоги щодо оформлення графічної частини кваліфікаційної роботи бакалавра.....	31
2.3 Вимоги до обсягу кваліфікаційної роботи бакалавра.....	33
3 Загальна структура пояснювальної записки кваліфікаційної роботи.....	34
4 Рекомендації щодо вирішення основних завдань кваліфікаційної роботи бакалаврів спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія».....	38
4.1 Вибір теми кваліфікаційної роботи.....	38
4.2 Вибір об'єкту автоматизації.....	39
4.3 Аналіз та вибір мови програмування та програмна реалізація автоматизованої системи тестування роботи WEB-додатків приладобудівних підприємств.....	45
4.4 Аналіз існуючих приладів та систем контролю присутності.....	54
4.5 Аналіз конструкцій маніпуляторів.....	61
4.6 Аналіз та обґрунтування основних складових маніпулятора.....	71
4.7 Аналіз та принцип роботи системи дозування для відмивки плат.....	81
4.8 Аналіз методів інтелектуального керування складними системами.....	92
4.9 Аналіз використання комп'ютерного зору на сучасному виробництві	101

4.10	Аналіз методів отримання керуючих впливів для біонічних рук	105
4.11	Аналіз засобів комп'ютерного моделювання робочого простору транспортних роботів у сучасних системах.....	114
4.11.1	Microsoft Robotic Studio	114
4.11.2	MatLab	117
4.11.3	Robotino SIM Demo	120
4.12	Розробка програмного забезпечення для мобільного робота Festo Robotino	122
4.12.1	Особливості розробки програмного забезпечення Festo Robotino	122
4.12.2	Розробка загального алгоритму роботи програмного забезпечення	125
4.12.3	Розробка програми пересування робота за заданою траєкторією в Robotino View.....	128
4.12.4	Тестування розроблених алгоритмів та процедур у Robotino SIM.	130
4.13	Вибір апаратного та програмного забезпечення мобільної платформи .	132
4.13.1	Розробка архітектури мобільної платформи	132
4.13.2	Визначення складових апаратного забезпечення мобільної платформи	135
4.13.3	Вибір програмного забезпечення для керування мобільною платформою	137
4.14	Розробка апаратного забезпечення мобільної платформи.....	140
4.14.1	Привідна система коліс	140
4.14.2	Сервопривод SG90	141
4.14.3	Реверсивний драйвер двигуна L298N	143
4.14.4	Arduino Uno R3	144
4.14.5	Ультразвуковий датчик відстані HC-SR04.....	145
4.14.6	Зовнішній вид розробленої мобільної платформи.....	146
4.15	Розробка програмного забезпечення для мобільної платформи	147
4.15.1	Розробка алгоритмічного забезпечення.....	147
4.15.2	Порядок розробки програмного забезпечення.....	149
4.15.3	Тестування ПЗ	152
4.16	Типові закони управління та регулятори	154
4.17	Синтез коригуючих пристроїв за логарифмічними частотними характеристиками	165
4.18	Синтез лінійної неперервної системи автоматичного управління	173
4.19	Розрахунки надійності розробленого модуля.....	193
4.20	Зміст підрозділу з охорони праці	196
4.20.1	Виробнича санітарія в лабораторії	196

4.20.2 Пожежна безпека в приміщенні лабораторії	197
4.20.3 Охорона праці користувачів персонального комп'ютера	198
Перелік джерел посилання	201
Додаток А Зразок бланка завдання на кваліфікаційну роботу бакалавра	205
Додаток Б Форма заяви щодо самостійності виконання письмової роботи	207
Додаток В Форма експертного висновку результатів перевірки на унікальність тексту	208
Додаток Г Форма відгуку керівника роботи.....	209
Додаток Д Форма рецензії на роботу	210
Додаток Е Критерії оцінювання результатів виконання і захисту кваліфікаційної роботи	211
Додаток Ж Зразок бланка титульного аркуша пояснювальної записки кваліфікаційної роботи бакалавра	212
Додаток З Форма відомості кваліфікаційної роботи бакалавра	213
Додаток И Приклад оформлення реферату	214
Додаток К Приклади записів у переліку посилань згідно з ДСТУ ГОСТ 7.1:2006	215
Додаток Л Коди виробів і програмних документів	217

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

- АСК – автоматизована система керування;
АСУ ТП – автоматизована система управління технологічним процесом;
ВНЗ – вищий навчальний заклад;
ВТ - віддалений термінал ;
ІЕП - інтелектуальний електронний пристрій;
ГІС – гнучка інтегрована система;
ЕК – екзаменаційна комісія;
ЕОМ – електронно-обчислювальна машина;
КР – кваліфікаційна робота;
НДР – науково-дослідна робота;
ООП – об’єктно-орієнтоване програмування;
ОП – охорона праці;
ОС – операційна система;
ПЗ – пояснювальна записка;
ПК – персональний комп’ютер;
ПЛК - програмовані логічні контролери;
САУ – системи автоматичного управління;
ТП – технологічний процес;
ТУ – технічні умови;
IDE – Integrated Development Environment (інтегроване середовище розробки);
SCADA – Supervisory Control And Data Acquisition (диспетчерське управління та збирання даних);

ЗАГАЛЬНІ ПОЛОЖЕННЯ

Метою атестації здобувачів вищої освіти, які закінчують підготовку за першим (бакалаврським) освітнім рівнем, є визначення фактичної відповідності якості їхньої підготовки вимогам до фахівців з вищою освітою згідно з освітньо-професійною програмою підготовки випускників. Атестація здобувачів вищої освіти здійснюється відповідно до Закону України «Про вищу освіту», «Положення про організацію освітнього процесу в Харківському національному університеті радіоелектроніки», «Положення про протидію академічному плагіату в Харківському національному університеті радіоелектроніки» та інших нормативних документів [1-4]. Строк проведення і тривалість проведення атестації випускників визначається графіком освітнього процесу закладу вищої освіти.

До атестації допускаються здобувачі вищої освіти, які в повному обсязі успішно виконали навчальний план підготовки фахівців першого (бакалаврського) рівня вищої освіти.

Захист кваліфікаційної роботи (КР) має показати рівень теоретичної та практичної підготовленості випускників, їх якостей щодо систематизації наукової інформації та аналізу актуальних проблем із спеціальності або освітньої програми.

Написання кваліфікаційної роботи бакалавра – це перший самостійний крок майбутнього фахівця, коли право остаточного вибору інженерно-технічних рішень і відповідальність за їх прийняття цілком належить його автору – випускнику. Доповідь за темою кваліфікаційної роботи і її захист, як одна із основних форм державної атестації здобувачів вищої освіти – важливі етапи навчання в університеті.

Головним завданням кваліфікаційної роботи є підготовка випускника до самостійної, творчої, інженерної, наукової роботи тощо у відповідній галузі на основі знань, отриманих в процесі навчання в університеті.

Згідно з освітньо-професійною програмою підготовки фахівців у галузі автоматизації та приладобудування, випускник спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія» має бути готовий до розв'язання задач з розроблення нових і вдосконалення, модернізації та експлуатації існуючих систем автоматизації з застосуванням сучасних програмно-технічних засобів та

інформаційних технологій, на основі аналізу об'єктів автоматизації, обґрунтування вибору програмно-технічних засобів, проектування систем автоматизації на основі мехатронних та робототехнічних модулів та розроблення системного та прикладного програмного забезпечення різного призначення.

Варто ще раз підкреслити, що кваліфікаційна робота бакалавра є самостійною роботою здобувача, за всі прийняті в КР технічні і економічні рішення, а також правильність і обґрунтованість розрахунків і графічних робіт несе відповідальність сам здобувач – автор кваліфікаційної роботи.

За своїм змістом і характером кваліфікаційні роботи бакалаврів спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія» підрозділяються на апаратні та програмні і за тематикою є пов'язаними з автоматизацією процесів управління та робототехнікою.

Даний навчальний посібник містить необхідну інформацію для виконання та захисту кваліфікаційної роботи бакалавра, встановлює загальні вимоги до організації і виконання кваліфікаційної роботи, змісту й обсягу кваліфікаційних робіт, порядку виконання та захисту робіт згідно з вимогами та рекомендаціями освітньо-професійної програми за спеціальністю.

Зміст навчального посібника розподілений на чотири основні частини. Перша частина присвячена питанням організації процесу підготовки та захисту кваліфікаційної роботи. У другій – подано загальні вимоги щодо оформлення пояснювальної записки КР згідно з ДСТУ 3008:2015 [5] та ДСТУ ГОСТ 7.1:2006 [6]. У третій частині представлена загальна структура пояснювальної записки, її основні складові елементи. У четвертій частині подано можливі тематики кваліфікаційних робіт бакалаврів за спеціальностями 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія», подано стислий довідковий матеріал щодо змісту розрахункової частини роботи з прикладами.

В навчальному посібнику використано матеріали кваліфікаційних робіт студентів ОП «Системна інженерія» (Белов Петро, Бондарев Андрій, Бондаренко Антон, Кулик Аліна, Реука Євген, Фільчакова Дарина, Цаплій Ігор, Шевченко Анастасія).

1 МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ДО КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА

1.1 Мета та задачі кваліфікаційної роботи бакалавра

Виконання, оформлення та захист кваліфікаційної роботи є завершальним етапом підготовки бакалавра спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія».

Головною метою кваліфікаційної роботи є:

- систематизація, упорядкування, розширення та поглиблення теоретичних і практичних знань, набутих під час навчання;
- розвиток і закріплення розрахункових, експериментальних, дослідницьких навичок;
- можливість здобувача вищої освіти використовувати набуті під час навчання знання для вирішення конкретних науково-технічних задач у галузі автоматизації та приладобудування;
- розвиток уміння самостійної роботи з літературними джерелами;
- володіння методикою технічного пошуку, технічної творчості;
- володіння методикою досліджень та експериментів під час вирішення проблем і питань кваліфікаційної роботи.

Цього можна досягти шляхом розробки кваліфікаційної роботи під керівництвом викладача, базуючись на знаннях, вміннях і навичках, здобутих під час навчання за освітньою програмою «Системна інженерія», практичного досвіду та поглибленого вивчення конкретної області автоматизації та приладобудування.

Під час підготовки та захисту кваліфікаційної роботи бакалавра виявляється освітньо-професійний рівень здобувача та здатність виконувати завдання на рівні інженерних посад у промислових, проектних і наукових організаціях, а також для продовження навчання з метою отримання кваліфікації другого (магістерського) рівня.

1.2 Організація виконання кваліфікаційної роботи бакалавра

Процес виконання кваліфікаційної роботи складається з таких етапів:

- визначення теми кваліфікаційної роботи та формулювання завдання на її виконання;

– опрацювання задач кваліфікаційної роботи згідно з завданням та оформлення пояснювальної записки;

– подання роботи на нормоконтроль;

– проходження перевірки на плагіат;

– підготовка до захисту, відгук керівника та рецензування роботи;

– захист кваліфікаційної роботи на засіданні екзаменаційної комісії (ЕК).

Кваліфікаційна робота виконується випускниками за тематикою випускової кафедри відповідно до предметної галузі спеціальності та освітньої програми.

Фахівців зі спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані технології» та 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» освітньої програми «Системна інженерія» готує кафедра комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки (КІТАМ). Кафедра складає перелік тем кваліфікаційних робіт з урахуванням специфіки спеціальності та освітньої програми, вимог галузевих стандартів вищої освіти для першого (бакалаврського) рівня. Окремі теми кваліфікаційних робіт можуть бути запропоновані здобувачами вищої освіти з відповідним обґрунтуванням доцільності їх розробки.

Кваліфікаційні роботи можуть також виконуватися за реальними завданнями, отриманими від підприємств, що є потенційними роботодавцями. До складу кафедральних кваліфікаційних комісій для атестації здобувачів освітнього рівня бакалавр включаються представники компаній та підприємств.

Кафедра визначає порядок допуску до захисту кваліфікаційної роботи та приймає рішення про недопущення до атестації здобувачів вищої освіти, які не виконали програму виробничої (або передатестаційної) практики або здобувачів, які не виконали календарний план атестації і не надали у встановлений термін підготовлену до захисту кваліфікаційну роботу, і подає це рішення до деканату.

Практично робота над кваліфікаційною роботою починається в період виробничої та передатестаційної практики. Здобувач повинен вивчити питання, які безпосередньо пов'язані з темою кваліфікаційної роботи.

Відповідно до теми КР здобувачам вищої освіти рекомендується: ознайомитися з основними публікаціями, нормативними, довідковими матеріалами з питань теми кваліфікаційної роботи; вивчити існуючі подібні об'єкти, їхні позитивні і негативні сторони; зібрати необхідні вихідні дані для роботи; ознайомитись та вибрати методику вирішення завдань; провести аналіз завдання на кваліфікаційну роботу та сформулювати мету і задачі КР.

Всі зібрані матеріали мають бути відповідним чином відображені у звітах з виробничої та передатестаційної практик. Заповнений щоденник з виробничої практики та оформлений звіт подається керівнику виробничої практики, а щоденник і звіт з передатестаційної практики – відповідно керівнику передатестаційної практики у терміни, регламентовані графіком навчального процесу відповідної освітньої програми.

Завдання на кваліфікаційну роботу видається після затвердження наказу. Зразок бланку завдання подано у додатку А. Перед початком виконання КР здобувач вищої освіти за допомогою керівника розробляє календарний план роботи на весь період з зазначенням черговості та термінів виконання окремих етапів. Завдання, яке підписане керівником кваліфікаційної роботи, затверджується завідувачем кафедри. Здобувач отримує від керівника КР завдання та ставить свій підпис. Керівник КР при цьому заповнює графу дати видачі завдання.

Керівник КР протягом всього періоду виконання кваліфікаційної роботи надає здобувачу вищої освіти методичну допомогу, рекомендує необхідну технічну літературу, систематично консулює і контролює хід роботи. Але це не звільняє здобувача від повної відповідальності за обґрунтованість прийнятих ним рішень, дотримання вимог нормативних документів і виконання календарного плану роботи. Здобувач вищої освіти зобов'язаний у встановлені кафедрою терміни звітувати про виконане перед керівником і завідувачем кафедри, які визначають ступінь готовності. Терміни завершальних етапів подання кваліфікаційної роботи подані у табл. 1.1.

Таблиця 1.1 – Терміни подання кваліфікаційних робіт

Етап	Строк, днів не пізніше до дня захисту
Подання КР на підпис керівнику	7
Подання КР на нормоконтроль	7
Подання КР на перевірку на плагіат	7
Подання КР на підпис завідувачеві кафедри	4
Подання КР на рецензію	3
Попередній захист КР	2
Подання КР з рецензією до ЕК	1

Здобувач вищої освіти, який не виконує план кваліфікаційної роботи або значно відстає в його виконанні, запрошується для пояснення цього на засідання кафедри, яка приймає відповідне рішення.

Закінчена кваліфікаційна робота бакалавра підписується здобувачем вищої освіти, далі надається керівнику для перевірки на відповідність завданню і вимогам. У разі необхідності керівник організовує заслуховування здобувача вищої освіти про виконану роботу.

Для запобігання потрапляння в публічні бази даних кваліфікаційних робіт, що мають високий рівень плагіату, впроваджено систему виявлення академічного плагіату.

Академічна доброчесність – це сукупність етичних принципів та визначених законом правил, якими мають керуватися учасники освітнього процесу під час навчання, викладання та провадження наукової діяльності.

Плагіат – оприлюднення (опублікування), повністю або частково, чужого твору під іменем особи, яка не є автором цього твору. Плагіат академічний – плагіат в академічному середовищі.

Видами плагіату є:

- копіювання інформації іншого автора без оформлення цитування;
- повторне використання свого тексту як абсолютно нової роботи;
- парафразування або адаптація – переказ своїми словами тексту автора;
- компіляція – написання роботи на підставі чужих матеріалів без самостійного дослідження та опрацювання джерел.

Перевірку кваліфікаційних робіт здобувачів першого рівня освіти на плагіат здійснює експерт із числа досвідчених штатних працівників профілюючої кафедри.

Здобувачі вищої освіти заповнюють і підписують заяву за встановленою формою (додаток Б), якою підтверджується факт відсутності в письмовій роботі запозичень з друкованих та електронних джерел третіх осіб без відповідних посилань, та інформованість про можливі санкції у випадку виявлення плагіату. Відмова у написанні заяви означає недопуск письмової роботи до захисту. Файли кваліфікаційної роботи разом із заявою подаються здобувачем вищої освіти керівникові кваліфікаційної роботи не пізніше ніж за тиждень до дати попереднього захисту.

Керівник роботи надає файли з текстом пояснювальної записки кваліфікаційної роботи для перевірки відповідальній особі (експерту). Експерт протягом не більш ніж 3 робочих днів здійснює перевірку, формує експертний висновок (додаток В) та передає їх керівнику атестаційної роботи. Керівник

кваліфікаційної роботи у відгуку разом з іншими пунктами (актуальність, якість, ступінь самостійності, наукове та практичне значення тощо) формулює остаточний висновок щодо оригінальності роботи, враховуючи наступні критерії оцінювання рівня оригінальності тексту [3]:

- високий (наукові та науково-методичні роботи) – більше 75% оригінального тексту;
- прийнятний (кваліфікаційні роботи) – більше 50% оригінального тексту;
- неприйнятний (потребує доопрацювання) – менше 50% оригінального тексту.

Якщо рівень оригінальності кваліфікаційної роботи на думку керівника є недостатнім, робота повертається автору на доопрацювання. Негативний експертний висновок є підставою до не допуску роботи до захисту завідувачем кафедри.

Здобувачі освітнього ступеня бакалавр несуть відповідальність за порушення вимог подання своєї кваліфікаційної роботи для перевірки антиплагіатною системою, у порядку визначеному Законодавством України, зокрема ст. 177 Кримінального кодексу, ст. 431, 432 Цивільного кодексу України, а також «Положенням про протидію академічному плагіату в Харківському національному університеті радіоелектроніки».

Керівник несе відповідальність за перевірку роботи у встановлені строки, прийняття рішення про доопрацювання та повторну перевірку на плагіат, а також про допуск роботи до попереднього захисту. Екзаменаційна комісія, оцінюючи кваліфікаційну роботу випускника, обов'язково має брати до уваги рівень плагіату в роботі. Наявність плагіату є підставою для зниження оцінки або визнання захисту незадовільним.

Керівник складає відгук на КР здобувача, в якому має бути подана характеристика роботи випускника, зокрема:

- новизна розробки, актуальність і ступінь її складності;
- вміння здобувача вищої освіти працювати з науково-технічною і патентною літературою;
- самостійність роботи здобувача вищої освіти, виявлена ним ініціатива, вміння користуватися сучасними методами і засобами досліджень, використання здобувачем вищої освіти стандартів та інших нормативних документів;
- ставлення здобувача вищої освіти до роботи над проектом, вміння працювати систематично, його акуратність, грамотність;
- рівень оригінальності тексту кваліфікаційної роботи.

У кінці відгуку робиться висновок про підготовленість здобувача вищої освіти до самостійної інженерної діяльності і можливості подання кваліфікаційної роботи до захисту в ЕК. Керівник дає якісну, але не кількісну оцінку роботі здобувача вищої освіти. Зразок форми відгуку на кваліфікаційну роботу бакалавра поданий у додатку Г.

Повністю оформлена і підписана керівником кваліфікаційна робота бакалавра, яка пройшла перевірку на нормоконтроль і плагіат, разом з відгуком подається завідувачу кафедри. Завідувач кафедри вирішує питання про допущення здобувача вищої освіти до захисту, здійснює про це відповідний запис на титульному аркуші пояснювальної записки.

Кваліфікаційна робота бакалавра, допущена кафедрою до захисту, направляється на рецензію. Рецензування кваліфікаційних робіт здійснюється науково-педагогічним працівником університету, який має науковий ступінь кандидата наук (доктора філософії) та/або доктора наук, або зовнішнім фахівцем, який має вищу освіту за відповідною спеціальністю. Склад рецензентів кваліфікаційних робіт визначається на засіданні випускової кафедри не пізніше ніж за два тижні до початку атестації випускників.

У рецензії мають бути відображені такі питання:

- відповідність кваліфікаційної роботи бакалавра, що рецензується, завданню і встановленим вимогам;
- актуальність теми КР бакалавра;
- оцінка відповідності завданню розділів і обсягу роботи;
- оцінка змісту матеріалу кваліфікаційної роботи бакалавра;
- обґрунтованість прийнятих у роботі наукових та інженерних рішень;
- доцільність науково-технічної літератури та вміння її цитувати;
- науково-технічний рівень та якість розрахунків, вирішення оптимізаційних задач;
- оцінка обґрунтованості наведеного комплексу конструкторської та технологічної документації і якість виконання креслень;
- оцінка правильності використання стандартів;
- обсяг і доцільність експериментальних досліджень, оцінка отриманих результатів;
- наявність зразків, макетів та інших результатів діяльності випускника та їхній рівень;
- можливість впровадження у виробництво або освітній процес результатів КР бакалавра в цілому або частково;

- оцінка стилю викладення матеріалу пояснювальної записки і відповідність оформлення вимогам стандартів;
- помилки та недоліки виконаної роботи.

У висновку рецензент вказує, чи відповідає кваліфікаційна робота бакалавра вимогам за даною спеціальністю та освітньою програмою та оцінює якість роботи за чотирибальною системою («відмінно», «добре», «задовільно», «незадовільно»). Зразок форми рецензії на КР бакалавра поданий у додатку Д.

Випускник має бути ознайомлений зі змістом рецензії не пізніше ніж за один день до її захисту для того, щоб він міг заздалегідь підготувати відповіді на зроблені рецензентом зауваження. Внесення змін в кваліфікаційну роботу після ознайомлення з рецензією забороняється. Рецензія, що містить аргументований критичний аналіз переваг і недоліків роботи, оголошується на засіданні ЕК під час обговорення результатів захисту.

Після отримання рецензії на кваліфікаційну роботу здобувач вищої освіти за рішенням кафедри проходить попередній захист КР перед комісією, склад якої визначається на засіданні кафедри і затверджується завідувачем кафедри.

Завершена кваліфікаційна робота з відгуком керівника і рецензією подається в екзаменаційну комісію.

1.3 Захист кваліфікаційної роботи

Захист кваліфікаційної роботи здобувачем відбувається на відкритому засіданні екзаменаційної комісії. До діяльності ЕК належить перевірка науково-теоретичної, навчальної та практичної підготовки майбутніх бакалаврів, вирішення питання про присвоєння їм відповідної кваліфікації та про видачу диплома бакалавра (з відзнакою/без відзнаки), розробка пропозицій про подальше вдосконалення бакалаврської підготовки на кафедрі. До складу ЕК входять викладачі факультету та працівники промисловості, голова призначається ректором ХНУРЕ.

Засідання ЕК проводяться у терміни, які визначаються навчальними планами, згідно з розкладом, що затверджується проректором або деканом факультету. Розклад роботи ЕК доводиться до загалу за місяць до початку засідань. Запис випускників на захист здійснює секретар ЕК після оголошення розкладу.

До початку захисту за кілька днів здобувач вищої освіти має залишити секретарю ЕК довідку з деканату факультету про повне виконання ним навчального плану та успішне складання всіх іспитів та заліків, випуску з усіма

оцінками за період навчання з теоретичних дисциплін, курсових проектів, робіт та практики, відгук керівника кваліфікаційної роботи та залікову книжку. В день захисту здобувач вищої освіти подає до ЕК пояснювальну записку, кресленики та презентацію. За бажанням здобувача вищої освіти можливо подати інші матеріали, які характеризують якість кваліфікаційної роботи: макет, зразки конструкцій, фотографії, авторські свідоцтва, статті, тези доповідей тощо.

Захист комплексної кваліфікаційної роботи, як правило, проводиться на одному засіданні екзаменаційної комісії, причому здобувачу, який захищається першим, доручається доповісти як про загальну частину роботи, так і про індивідуальну частину завдання. Усі випускники, які виконували комплексну КР, повинні бути повною мірою обізнані із загальною частиною роботи і готові до запитань членів екзаменаційної комісії не тільки з індивідуальної, а й із загальної частини роботи.

Захист КР відбувається на відкритому засіданні ЕК за участі не менш трьох її членів, а також за обов'язкової присутності голови комісії. Тривалість засідання ЕК не має перевищувати шести академічних годин на день. Тривалість захисту однієї КР не має перевищувати 20 хв.

На засіданні ЕК оголошується назва КР, прізвище, ім'я і по батькові її автора, надається інформація про наявність необхідних у справі документів і стисла характеристика студента (його успішність, наявність публікацій і виступів за темою роботи на засіданнях наукових конференцій, семінарів, гуртків тощо). Далі випускнику надається до 10 хвилин для доповіді.

Доповідь здобувач викладає у довільній формі про сутність роботи, основні технічні, наукові рішення, отримані результати (при цьому можуть використовуватися різні форми візуалізації доповіді: обов'язковий графічний матеріал КР, визначений завданням, слайди, мультимедійні проектори тощо). У своєму виступі він має зосередити основну увагу на головних підсумках роботи, що ним особисто розроблені. За наявності здобувач може провести демонстрацію експерименту, макетів, зразків тощо (демонстрація проводиться безпосередньо на засіданні екзаменаційної комісії). Зміст доповіді, характер її виконання і впевненість під час відповідей на запитання значною мірою визначають оцінку роботи.

Після закінчення доповіді починається дискусія. Члени екзаменаційної комісії й особи, запрошені на захист, в усній формі можуть ставити будь-які запитання за матеріалами, поданими до захисту. Відповідаючи на запитання, здобувач вищої освіти має освітлювати тільки суть справи, виявляти скромність в оцінках своїх результатів і тактовність до осіб, що ставлять питання.

Після того, як вичерпані всі питання, голова надає слово керівникові студента. У своєму виступі керівник розкриває ставлення студента до роботи, характеризує його особисті якості як майбутнього фахівця. За відсутності на засіданні екзаменаційної комісії керівника голова зачитує його письмовий відгук про роботу.

Потім заслуховується висновок рецензента щодо поданої до захисту роботи. У ньому має бути розкрита актуальність теми, виконаний аналіз її структури, підтверджена обґрунтованість висновків і рекомендацій, зазначені недоліки роботи. За відсутності рецензента на захисті рецензія зачитується головою ЕК.

Далі голова ЕК надає слово всім охочим виступити за темою і змістом розглянутої роботи. На закінчення слово надається студентові для відповіді на висловлені зауваження і побажання.

Після того, як дискусію закінчено, за бажанням студента йому може бути надане заключне слово. На цьому основна частина процедури захисту кваліфікаційної роботи вважається завершеною.

Підсумки захисту підводяться на закритому засіданні членів екзаменаційної комісії. У процесі обговорення результатів роботи докладному аналізу мають піддаватися вірогідність й обґрунтованість усіх положень, висновків і рекомендацій, що містяться в роботі. Для вирішення суперечливих питань на обговорення запрошується керівник кваліфікаційної роботи, завідувач кафедри, за профілем якої захищається кваліфікаційна робота.

Рішення щодо оцінки за захист кваліфікаційної роботи стосовно кожного випускника приймається відкритим голосуванням лише членами цієї ЕК звичайною більшістю голосів. За однакової кількості голосів, що визначають різні оцінки, голос голови ЕК є ухвальним.

Підсумки захисту й оцінка кваліфікаційної роботи оголошуються головою екзаменаційної комісії після обговорення оцінок на закритому засіданні комісії. Повторний захист кваліфікаційної роботи з метою підвищення оцінки не дозволяється.

Випускник, який отримав оцінку «незадовільно» під час захисту кваліфікаційної роботи, після завершення атестації відраховується з Університету як такий, що виконав навчальний план, але не пройшов атестації. При цьому йому видається академічна довідка. ЕК встановлює, чи може здобувач подати на повторний захист ту саму роботу з доопрацюванням, чи він зобов'язаний опрацювати нову тему.

У разі, якщо випускник не з'явився на засідання ЕК для захисту кваліфікаційної роботи, то у протоколі комісії відзначається, що він є не

атестованим у зв'язку з неявкою на засідання ЕК. Випускники, які не атестовані у затвердженій для них строк або не захистили кваліфікаційну роботу, мають право на повторну атестацію в університеті упродовж трьох років після відрахування з університету.

Повторна атестація проводиться лише з тих форм атестації, з яких раніше були отримані незадовільні оцінки, і в обсязі тих навчальних програм, за якими навчався випускник. У випадках повторного незадовільного захисту кваліфікаційної роботи зазначені особи втретє до атестації не допускаються.

У разі неприбуття випускника на захист кваліфікаційної роботи з поважної причини (хвороба, сімейні обставини тощо), підтвердженої документально, голова ЕК за клопотанням ректора може визначити окремий порядок та час захисту кваліфікаційної роботи для цього випускника в межах строку повноважень ЕК.

Випускнику, який склав атестацію, рішенням ЕК присвоюється відповідний освітній рівень і на підставі рішення ЕК видається диплом. Результати захисту кваліфікаційних робіт оформлюються протоколом, який складається на кожного випускника окремо.

Матеріали КР, за якими прийнято позитивне рішення екзаменаційною комісією, здаються в архів університету, де зберігаються у встановленому порядку. За необхідності передачі матеріалів роботи промисловому підприємству з метою впровадження отриманих результатів у виробництво з них знімається копія. Електронні копії робіт передаються до бібліотеки ХНУРЕ, а також можуть зберігатися в базі даних випускаючої кафедри і використовуватися в навчальному процесі. Кращі кваліфікаційні роботи кафедра рекомендує на університетські (міські, обласні) конкурси.

1.4 Критерії оцінювання кваліфікаційної роботи

Під час атестації і визначенні оцінки випускника враховуються:

– рівень його фахової підготовки, уміння творчо застосовувати отримані знання для вирішення практичних завдань;

– якість доповіді (обґрунтованість, чіткість, стислість), здатність випускника упевнено та правильно відповідати на теоретичні питання і обґрунтовувати практичні дії, логічно будувати свою доповідь (виступ), аргументовано відстоювати власну точку зору.

Результати захисту кваліфікаційної роботи визначаються оцінками за стобальною шкалою. Під час оцінювання якості виконання здобувачем кваліфікаційної роботи застосовуються критерії, подані у додатку Е.

Для отримання відповідної оцінки за кваліфікаційну роботу бакалавра необхідно досягти виконання наступних умов.

Відмінно (90-100 балів): Кваліфікаційна робота бакалавра виконана вчасно, самостійно, забезпечує повне розкриття теми. Правильно визначено предмет, об'єкт дослідження. Мета та завдання чітко окреслені та реалізовані у дослідженні. Автор використовує сучасні аналітичні та методологічні інструментарії. Робота містить елементи новизни та характеризується високою якістю і глибиною теоретико-методологічного аналізу, критичного огляду літературних джерел, наявністю наукової проблематики. Узагальнення і висновки базуються на якісно опрацьованій статистичній інформаційній базі, що дозволяє чітко визначити авторську позицію.

Подані рекомендації автора мають практичну цінність, містять розрахунки показників економічної ефективності. У роботі розроблені суттєві аспекти впровадження рекомендацій.

Для проектного варіанту КР:

- реалізована власна концепція пристрою за розробленою власноруч структурною схемою (не типова, розроблена для рішення конкретної задачі);
- програмний код створений здобувачем власноруч з використанням існуючих бібліотек;
- власноруч реалізовано інтерфейс керування та конфігурування параметрів роботи пристрою;
- реалізовано продуманий інтерфейс користувача з урахуванням сучасних вимог щодо подібних рішень.

Для програмного варіанту КР:

- наявність розвинутого інтерфейсу користувача;
- наявність інструментів редагування параметрів роботи програмного засобу (меню конфігурації);
- реалізована можливість формування звітів у вигляді форм на екрані;
- розроблена структура бази даних;
- реалізована архітектура MVC;
- реалізована можливість формування звітів у вигляді друкованих форм, що реалізовано власноруч, а не вбудованою можливістю, наприклад, в програмі AdobeReader.

Для КР з автоматизації процесів управління:

- побудовано функціональну схему лінійної неперервної системи автоматичного управління (САУ);
- визначено повний перелік конструктивних і електричних властивостей об'єкту або системи, що підлягають автоматичному контролю;
- обчислені передавальні функції всіх елементів системи і побудовано структурну схему нескорегованої системи;
- визначені передавальні функції розімкненої й замкненої системи регулювання, а також їхні характеристичні поліноми;
- проведена оцінка стійкості розімкненої системи та замкненої системи за допомогою декількох критеріїв, визначені запаси стійкості за амплітудою і фазою;
- побудовано ЛАЧХ і ЛФЧХ вихідної розімкненої системи, ЛАЧХ коригуючої ланки, визначено її передавальну функцію;
- побудовано структурну схему системи з коригуючою ланкою і розраховано перехідну характеристику, проведено моделювання САУ;
- проведено критичний аналіз отриманих результатів і оцінку якості управління в скорегованій системі.

Відгук і рецензія позитивні. Доповідь аргументована, проілюстрована бездоганно оформленими наочними матеріалами, свідчить про формулювання власної думки студента щодо предмета дослідження та є логічною, повною. Відповіді на питання правильні та стислі.

Добре (75-89 балів): Кваліфікаційна робота виконана вчасно, теоретичні узагальнення та висновки аналітичної частини в основному правильні. Проте існують несуттєві недоліки у виявленні логічності зв'язку заходів, що пропонуються для вирішення проблем за допомогою проведеного аналізу статистичних та фактичних матеріалів, обґрунтування та розрахунків ефективності запропонованих рішень, що впливає на глибину особистого аналізу студентом фактичної інформації. Застосування сучасного аналітичного інструментарію обмежено. Подані у роботі авторські пропозиції повною мірою не містять аналітичного обґрунтування економічної доцільності їх реалізації.

Для проектного варіанту КР:

- реалізована типова концепція засобу автоматизації для рішення поставленої задачі;
- програмний код створений здобувачем власноруч з використанням існуючих бібліотек;
- реалізовано простий інтерфейс користувача для управління автоматизованим пристроєм.

Для програмного варіанту КР:

- наявність розвинутого меню користувача (категорії, підкатегорії);
- реалізація перевірки правильності вводу даних в екранні форми;
- реалізована клієнт серверна архітектура;
- розроблена структура бази даних.

Для КР з автоматизації процесів управління:

- побудовано функціональну схему лінійної неперервної системи автоматичного управління (САУ);
 - визначено достатній перелік конструктивних і електричних властивостей об'єкту або системи, що підлягають автоматичному контролю;
 - передавальні функції елементів системи не достатньо обґрунтовані, побудовано структурну схему нескорегованої системи;
 - визначені передавальні функції розімкненої й замкненої системи регулювання, а також їхні характеристичні поліноми;
 - проведена оцінка стійкості розімкненої системи та замкненої системи за одним критерієм, визначені запаси стійкості за амплітудою і фазою;
 - побудовано ЛАЧХ і ЛФЧХ вихідної розімкненої системи, ЛАЧХ коригуючої ланки, визначено її передавальну функцію;
 - побудовано структурну схему системи з коригуючою ланкою і розраховано перехідну характеристику, проведено моделювання САУ;
 - наявний аналіз отриманих результатів.

Відгук і рецензія позитивні, але мають окремі зауваження до роботи. Доповідь насичена фактичною інформацією, що відображає відповідні результати проведеного дослідження. Відповіді на питання правильні, але не завжди повні чи конкретні.

Задовільно (60-74 бали): Тема роботи в основному розкрита, але мають місце недоліки змістового характеру. Теоретико-аналітична частина та пропозиції обґрунтовано непереконливо, відсутні розрахунки, що дозволяють аргументувати зроблені авторські узагальнення та висновки. Є зауваження щодо логічності та послідовності викладеного матеріалу, який має переважно описовий характер. Робота недбало оформлена.

Для проектного варіанту КР:

- програмний код створений з прикладів в мережі Internet;
- здобувач зібрав пристрій з готових модулів та поєднав їх провідниками.

Для програмного варіанту КР:

- реалізовані базові функції обчислення завдання;
- дані для розрахунку вводяться в відповідні поля на екрані, але не мають функцій перевірки правильності вводу даних;

– коефіцієнти розрахунку задані в тілі програми та не змінюються в процесі користування програмним засобом (тільки за допомогою перекомпіляції вихідного коду).

Для КР з автоматизації процесів управління:

– функціональна схема лінійної неперервної системи автоматичного управління (САУ) побудована з помилками;

– перелік конструктивних і електричних властивостей об'єкту або системи, що підлягають автоматичному контролю, представлений не в повному обсязі або зовсім відсутній;

– передавальні функції елементів системи підібрані з помилками, структурна схема має недоліки;

– проведена оцінка стійкості розімкненої системи та замкненої системи за одним критерієм, запаси стійкості за амплітудою і фазою визначені з помилками;

– побудовані ЛАЧХ і ЛФЧХ вихідної розімкненої системи мають похибки;

– побудовано структурну схему системи з коригуючою ланкою; розрахована перехідна характеристика, проведено моделювання САУ;

– аналіз отриманих результатів відсутній.

Доповідь прочитана за текстом, студент не володіє окремими питаннями теми, не всі відповіді на запитання правильні або повні. Наочні матеріали не повною мірою відображають зміст виконаної роботи.

Незадовільно (0-59 балів): Відсутня логіка у побудові структури дослідження. У роботі відсутнє розуміння мети, завдань, предмета дослідження. Назви окремих розділів не відповідають їх змісту. Теоретичний аналіз та визначення стану процесів, що є предметом розгляду, мають компіляційний характер, відсутні посилання на використані літературні джерела. Відсутні самостійні судження у запропонованих рекомендаціях і пропозиції. Поданий статистичний матеріал є застарілим. Оформлення роботи має суттєві недоліки.

Доповідь не відображає зміст виконаної роботи, більшість відповідей на питання неправильні, здобувач не володіє предметом дослідження. Наочні матеріали до захисту роботи відсутні.

Робота до захисту не допускається: роботу подано з порушенням строків, установлених регламентом. Відгук наукового керівника негативний. Тему роботи вчасно не було затверджено наказом по університету. Студент не володіє поданим матеріалом, не орієнтується у предметі дослідження. Порушена логіка поданого матеріалу. Назви розділів не відповідають змісту. Змістове наповнення розділів не пов'язано між собою. Оформлення роботи не відповідає вимогам.

2 ВИМОГИ ДО ОФОРМЛЕННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

2.1 Вимоги до оформлення пояснювальної записки

Робота має бути оформлена згідно з вимогами нормативних документів Міністерства освіти і науки України, національних стандартів України і міждержавних комплексних систем стандартів.

2.1.1 Загальні вимоги

Текст пояснювальної записки (ПЗ) набирається в редакторі Microsoft Word шрифтом Times New Roman чорного кольору прямого накреслення через півтора міжрядкові інтервали кеглем 14 та роздруковується в одному примірнику одностороннім друком на аркушах формату А4 (210 x 297мм). Рекомендовано на сторінках КР використовувати поля такої ширини: верхнє і нижнє – не менше ніж 20 мм, лівє – не менше ніж 25 мм, правє – не менше ніж 10 мм. Абзацний відступ для всього тексту ПЗ має бути однаковим і дорівнювати п'яти символам (15-17 мм).

Мова пояснювальної записки державна або іноземна (за бажанням здобувача і за узгодженням з кафедрою).

Заголовки структурних елементів КР та заголовки розділів слід друкувати з абзацного відступу великими літерами напівжирним шрифтом без крапки в кінці. Дозволено їх розміщувати посередині рядка.

Окремі частини пояснювальної записки нумерують арабськими цифрами. Розділи мають наскрізну нумерацію. Номер підрозділу складається з номера розділу та порядкового номера підрозділу, відокремлених крапкою (наприклад, 1.2 – другий підрозділ першого розділу). Номер пункту – з номера розділу, підрозділу та порядкового номера пункту, відокремлених крапками (наприклад, 1.2.5 – п'ятий пункт другого підрозділу першого розділу).

Структурні елементи «Титульний аркуш», «Завдання», «Реферат», «Зміст», «Вступ», «Висновки», «Перелік джерел посилань» не нумерують.

Переноси слів у заголовках розділів не дозволяються. Якщо слово у заголовку не може розміститися на одному рядку, то його слід перенести на наступний рядок. Переноси слів у заголовках підрозділів, пунктів або підпунктів дозволяються. Не дозволяється розміщувати назву розділу, підрозділу, а також пункту й підпункту на останньому рядку сторінки.

У текстовій частині пояснювальної записки не слід вживати розмовні звороти, техніцизми, професіоналізми, жаргонні слова та звороти; використовувати для позначення одного поняття різні терміни, близькі за своїм змістом, також іншомовні слова та терміни за наявності в українській мові рівнозначних слів і термінів; вживати скорочення слів і словосполучень, крім встановлених правилами орфографії та нормативними документами.

Скорочувати можна пояснювальні слова (та інше – та ін., дивися – див.). Скорочують назви великих чисел. При цьому скорочення, утворенні методом відсікання, потребують крапки в кінці, а утворені методом стягування – ні (тис., але млн, грн).

Прізвища, назви фірм, установ та організацій, інші власні назви в тексті наводять мовою оригіналу. Допускається такі власні назви наводити у перекладі, додаючи за першою згадкою оригінальну назву у дужках (наприклад, роботи Люїса (Lewis)).

Кількісні числівники рекомендується писати в цифровій формі без відмінкових закінчень. Цифрова форма не рекомендується для однозначних чисел (шість мікросхем, а не 6 мікросхем). Числа з позначенням одиниць фізичних величин, грошових та інших величин завжди пишуться цифрами.

Дозволяється скорочувати слова або словосполучення, що часто вживаються в тексті, за такою схемою. Слово або словосполучення при його першому вживанні в тексті слід писати повністю, а після нього поруч у дужках навести скорочений варіант, який і застосовують у подальшому тексті (наприклад, технологічний процес (ТП)).

У тексті, за винятком формул, таблиць і рисунків, не слід використовувати математичний знак «←» перед значеннями величин. Замість математичного знака «←» слід вживати слово «мінус». Не слід вживати будь-які математичні знаки №; % тощо без цифр, а індекси нормативних документів (ДСТУ, ГОСТ, ТУ, ISO) без реєстраційного номера.

У тексті слід використовувати стандартизовані одиниці фізичних величин, їхні позначення. Застосовувати в одному тексті різні системи фізичних величин та їх позначень не дозволяється. У позначеннях одиниць крапку як знак скорочення не ставлять (крім випадків скорочення слів, які входять до найменування одиниці, але самі не є найменуванням одиниць, наприклад, 760 мм рт. ст.). Позначення одиниць, назви яких походять від імен учених, пишуть з великої літери, а їх найменування – з малої (наприклад, кДж, але кілоджоуль). Не слід скорочувати позначення фізичних величин, якщо вони використовуються без цифр.

2.1.2 Формули та рівняння

Формули та рівняння розташовують безпосередньо після посилання на них у тексті. Формули виділяють в окремі рядки і розташовують посередині сторінки. Вище і нижче кожної формули або рівняння слід залишати не менше одного вільного рядка.

Нескладні нумеровані формули розташовують безпосередньо в тексті. Нумерують тільки ті формули та рівняння, на які є посилання в тексті. Нумерувати слід арабськими цифрами в круглих дужках у крайньому правому положенні на рядку. Їх номер складається з номера розділу та порядкового номера формули або рівняння, відокремлених крапкою наприклад, формула (1.3) – третя формула, яка належить до першого розділу.

Номер, який не вміщується у рядку з формулою, переносять у рядок нижче формули. Номер формули-дробу подають на рівні основної горизонтальної риски. Номер групи формул, розміщених на окремих рядках і об'єднаних фігурною дужкою (парантезом), ставиться справа від вістря парантеза, яке знаходиться в середині групи формул і звернене в сторону номера.

Пояснення значень символів і числових коефіцієнтів, що входять до формули чи рівняння, якщо вони не пояснені раніше у тексті, слід наводити безпосередньо під формулою у тій послідовності, в якій вони наведені у формулі чи рівнянні.

Після написання формули ставиться кома і з нового рядка даються пояснення до кожного символу та числового коефіцієнта. Пояснення познач слід подавати без абзацного відступу з нового рядка, починаючи зі слова «де» без двокрапки. Позначки, яким встановлюють визначення чи пояснення, рекомендовано вирівнювати у вертикальному напрямку.

Обов'язковим є пояснення позначки одиниці виміру відповідної фізичної величини. Між останньою цифрою та одиницею виміру залишають проміжок (крім позначення одиниць плоского кута – кутових градусів, кутових мінут і секунд, які пишуть безпосередньо біля числа вгорі). У формулах і/або рівняннях верхні та нижні індекси, а також показники степені, в усьому тексті мають бути однакового розміру, але меншими за літеру чи символ, до якого вони відносяться.

Приклад

Напруження в стержні розраховується за формулою:

$$\sigma = \frac{S_i}{\varphi \cdot F}, \quad (1.1)$$

де S_i – розрахункове зусилля в i -му стержні, $S_i = 54 \cdot 10^3$ Н;

φ – коефіцієнт зменшення допустимих напружень під час роботи стержня в умовах поздовжнього згину, $\varphi = 0,89$;

F – площа поперечного перетину стержня, $F = 3,87 \cdot 10^{-4}$ м².

$$\sigma = \frac{54000}{0,89 \cdot 3,87 \cdot 10^{-4}} \approx 156,78 \cdot 10^6 \text{ (Па)}.$$

Якщо під час вирішення формули слід розв'язати одну або декілька проміжних формул, то послідовність дій така:

- записують основну формулу та перелічують символи, які входять до неї;
- з нового рядка записують зміст допоміжних формул, перелічують символи, які входять до них, та проводять розрахунки за допоміжними формулами;

- проводять розрахунки за основною формулою.

Переносити формули чи рівняння на наступний рядок, коли вони не вміщуються в одному, дозволяється тільки на знаках виконуваних операцій: рівності «=», плюс «+», мінус «-», множення «×». Перенесення на знаку ділення «:» слід уникати. Знак операції повторюється на початку наступного рядка. Номер формули при її перенесенні вміщують на рівні останнього рядка.

Загальне правило пунктуації в тексті з формулами таке: формула входить до речення як його рівноправний елемент. Тому в кінці формул і в тексті перед ними розділові знаки ставлять відповідно до правил пунктуації.

Двокрапку перед формулою ставлять лише у випадках, передбачених правилами пунктуації:

- у тексті перед формулою є узагальнююче слово;
- цього вимагає побудова тексту, що передує формулі.

Формули, що йдуть одна за одною і не відокремлені між собою текстом, слід відокремлювати комами або крапками з комою.

Знаки, що відокремлюють формули, об'єднані фігурною дужкою, ставлять всередині. Після громіздких математичних виразів (визначники, матриці) розділові знаки можна не ставити.

Приклади числових значень величин:

- числові значення величин з допусками наводять так:

$$(65 \pm 3) \% ; 80 \text{ мм} \pm 2 \text{ мм} \text{ або } (80 \pm 2) \text{ мм}$$

– діапазон чисел фізичних величин наводять, використовуючи прикметники «від» і «до»:

від 1 мм до 5 мм (а не від 1 до 5 мм)

– якщо треба зазначити два чи три виміри, їх подають так:

80 мм × 25 мм × 50 мм (а не 80 × 25 × 50 мм)

2.1.3 Таблиці

Таблицю розташовують безпосередньо після посилання на неї або на наступній сторінці. На всі таблиці мають бути посилання в тексті. Таблиці слід нумерувати арабськими цифрами наскрізною нумерацією, за винятком таблиць, що наводяться у додатку. Можлива нумерація таблиць у межах розділу. Тоді номер таблиці складається з номера розділу та порядкового номера таблиці, відокремлених крапкою. Назву таблиці друкують з великої літери і розміщують над таблицею з абзацного відступу. Наприклад,

Таблиця 2.1 – Базові показники технологічності

Коефіцієнт	Позначення	φ_i
1. Використання мікросхем і мікроскладань у блоці	$K_{в.мс}$	1,000
2. Автоматизації та механізації монтажу	$K_{а.м}$	1,000
3. Механізації підготовки ЕРЕ	$K_{м.п.ере}$	0,750

Назва таблиці має відображати її зміст, бути конкретною та стислою. Якщо з тексту КР можна зрозуміти зміст таблиці, її назву можна не наводити. Якщо таблиця виходить за межі формату сторінки, її поділяють на частини і розміщують одну частину під іншою, або поруч, або переносять частину таблиці на наступну сторінку, повторюючи в кожній частині таблиці її головку та боковик. У разі переносу назву таблиці вказують тільки над першою частиною таблиці, а над іншими вказують «Продовження таблиці ...».

В процесі поділу таблиці на частини допускається її головку або боковик замінити відповідно номерами граф чи рядків. У цьому випадку номери граф чи рядків вказують арабськими цифрами у першій частині таблиці. Наступні частини таблиці починаються з нумерованого рядка або боковика. При цьому в першій частині таблиці гранична лінія знизу не проводиться.

У кінці заголовків і підзаголовків таблиць крапки не ставлять. Діагональний розподіл головки таблиць не дозволяється. Висота рядків таблиці має бути не менше 8 мм.

Окремі графи «№ з/п» та «Позначення одиниць фізичної величини» в таблицю не включають. За необхідності нумерації рядків їх порядкові номери вказують у графі перед найменуванням.

Заголовки граф таблиці починають з великої літери. Підзаголовки починають з малої літери, якщо вони складають одне речення з заголовком, або з великої літери, якщо вони мають самостійне значення. Заголовки і підзаголовки граф вказують в однині. У разі відсутності окремих даних у відповідній частині таблиці слід ставити риску.

Позначення одиниці фізичної величини, що є загальною для всіх даних у таблиці, слід вказувати після її назви. Позначення одиниці фізичної величини, що є загальною для всіх даних у графі або в рядку, вказувати після назви графі або рядка.

Цифри в графах таблиць слід розташовувати за класами чисел по всій графі одна понад одною. Числові величини в одній графі повинні мати однакову кількість десяткових знаків.

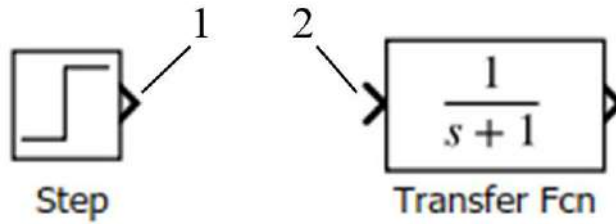
2.1.4 Рисунки та графіки

Ілюстрації слід нумерувати арабськими цифрами порядковою нумерацією в межах розділу. Номер ілюстрації складається з номера розділу і порядкового номера ілюстрації, відокремлених між собою крапкою.

Ілюстрації (рисунки, графіки, схеми, діаграми, фотознімки) слід розміщувати у тексті симетрично до ширини рядка безпосередньо після того, як вони згадуються вперше, або на наступній сторінці. На всі ілюстрації у тексті мають бути посилання. Кожна ілюстрація позначається словом «Рисунок», яке розташовують на наступному рядку по центру, поряд з ним ставиться номер і далі через тире з великої літери пишеться його назва без крапки у кінці. Якщо з тексту роботи зрозуміло зміст рисунка, його назву можна не наводити. Якщо рисунок, крім назви, має пояснення, то їх розміщують на наступному рядку після рисунку перед його назвою. Приклад оформлення рисунку та посилання на нього наведено нижче.

Приклад.

На рис. 1.2 показано з'єднання блоків відповідно до структури моделі.



1 – вихідний порт блоку; 2 – вхідний порт блоку

Рисунок 1.2 – З'єднання блоків

Якщо ілюстрація не вміщується на одному аркуші, її можна переносити на інші аркуші, розташовуючи назву ілюстрації на першому аркуші, а на інших аркушах на місці назви вказують: «Рисунок ____, аркуш ____».

Графіки та діаграми для зображення функціональних залежностей дозволяється виконувати з нанесенням шкал значень величин або без них. Шкали на координатній осі можуть бути нанесені координатною сіткою, розподільчими штрихами або їх комбінацією. Значення змінних величин можна відкладати на координатних осях у лінійному, нелінійному або в логарифмічному масштабах. Одиниці вимірювання на графіках і діаграмах слід наносити поруч з назвою змінної величини через кому. Числа біля шкал слід розміщувати поза полем графіка або діаграми.

Якщо на графіку зображено залежність декількох функцій від одного аргументу, то шкали функцій слід розташовувати паралельно координатній осі.

У разі розташування на одному графіку кількох функціональних залежностей дозволяється зображувати ці залежності лініями різних типів.

2.1.5 Переліки

Перед переліком слід ставити двокрапку, а позиції переліку починати з малої літери з абзацного відступу відносно попереднього рівня підпорядкованості. Якщо подають переліки одного рівня підпорядкованості, на які у звіті немає посилань, то перед кожним із переліків ставлять знак «тире», наприклад:

Оптимальні температурні діапазони:

- від плюс 1 до мінус 5° С;
- від мінус 10 до мінус 40° С.

Якщо у звіті є посилання на переліки, підпорядкованість позначають малими літерами української абетки, далі – арабськими цифрами, далі – через знаки «тире».

Після цифри або літери певної позиції переліку ставлять круглу дужку. Наприклад:

Оптимальні температурні діапазони:

а) від плюс 1 до мінус 5° С:

1)

2)

–

–

б) від мінус 10 до мінус 40° С.

Якщо наводять ряд або діапазон числових значень, які виражені однаковою одиницею фізичної величини, позначення цієї одиниці вказують за останнім числовим значенням ряду або діапазону, наприклад: 1,0; 1,5; 2,0; 2,5; 3,0 мм.

2.1.6 Посилання

У разі посилання у тексті пояснювальної записки на розділи, підрозділи, пункти, підпункти, ілюстрації, таблиці, формули, рівняння слід писати так.

Для розділів, підрозділів, підпунктів, таблиць, ілюстрацій:

– «...розрахунок проведений у розділі 4,»;

– «...результат дивись у 2.1.»;

– «...за даними 1.3.4 ...»;

– «...який показано на рисунку 1.3»;

– «...згідно з таблицею 3.2,....».

Для математичних виразів, формул та рівнянь:

– «...за результатом (3.2) можна.....»;

– «...відповідно до формули (3.1) ...»;

– «...у рівняннях (1.23)-(1.25) ...»;

– «...за рівнянням (3.25) отримуємо.....».

2.1.7 Додатки

До додатків слід відносити матеріал, який з будь-яких причин не внесено до тексту (ілюстрації, таблиці, програми, зразки розрахунків). Кожен додаток має починатися з нової сторінки і мати заголовок, надрукований малими літерами з першої великої симетрично відносно тексту сторінки. Посередині

рядка над заголовком друкується слово «ДОДАТОК» і відповідна велика літера, що позначає додаток.

Додатки слід позначати послідовно великими літерами української абетки, за винятком Г, Є, І, Ї, Й, О, Ч, Ь, наприклад: ДОДАТОК А, ДОДАТОК Б тощо. Якщо додаток один, то він позначається як ДОДАТОК А.

Текст кожного додатка за необхідності може бути поділений на розділи і підрозділи, які нумерують у межах кожного додатка. У цьому разі перед кожним номером ставлять позначення додатка (літеру) і крапку, наприклад, А.2 – другий розділ додатка А; В.3.1 – перший підрозділ третього розділу додатка В.

Ілюстрації, таблиці і формули, які розміщені в додатках, нумерують у межах кожного додатка, наприклад: рисунок Д.1.2 – другий рисунок першого розділу додатка Д; формула (А.1) – перша формула додатка А; таблиця Е.2 – друга таблиця додатка Е.

Додатки повинні:

- мати спільну з рештою тексту документа наскрізну нумерацію сторінок;
- розташовуватися в порядку появи посилань на них у тексті документа.

2.2 Вимоги щодо оформлення графічної частини кваліфікаційної роботи бакалавра

Графічна частина кваліфікаційної роботи бакалавра виконується за правилами ЄСКД. Склад графічної частини визначається керівником КР.

Частина графічного матеріалу оформлюється у вигляді презентації. Електронна презентація – це електронний документ, який є набором слайдів, призначеним для демонстрації аудиторії. Метою будь-якої презентації є візуальне подання роботи, максимально зручне для сприйняття.

Презентація для захисту кваліфікаційної роботи бакалавра виконується в електронному вигляді на комп'ютері у програмі Microsoft PowerPoint та може містити від 12 до 20 слайдів.

До презентації можна винести:

- основні формули;
- рисунки, ескізи;
- графіки, діаграми;
- схеми алгоритмів;
- структурні, електричні принципові та інші типи схем;
- таблиці;
- результати розрахунків.

Задачі презентації:

- включати всю необхідну інформацію, достатню для сприйняття аудиторією без пояснень;
- звернути увагу на найбільш суттєві результати виконаної роботи;
- надавати інформацію аудиторії максимально комфортно.

Кожен слайд має містити інформацію, яка б у ході перегляду на екрані проектора легко читалася, тобто розмір шрифту має бути оптимальний. Обсяг інформації має бути також оптимальний. Неправильно буде представити одну фотографію або один графік на одному слайді, оптимально – чотири фотографії (графіка) на плакат. Але якщо графік або рисунок (схема) дуже складний, то допускається один рисунок на один слайд. Слайди мають бути пронумеровані, номери повинні легко читатися.

На кожному слайді має бути заголовок, розташований зверху. Заголовок зазвичай виділяється більшим шрифтом, ніж основний текст презентації та має бути коротким і лаконічним, оптимально 6-7 слів.

Презентація починається зі слайда, що містить тему КР, ім'я здобувача, шифр групи, в якій він навчається та дані про керівника (посада, кафедра, ПІБ). Другий слайд містить актуальність кваліфікаційної роботи. На наступному слайді зазначають мету і завдання роботи. Потім подають основні розділи КР у тій послідовності, в якій вони викладені у пояснювальній записці. Наприкінці презентації наводять висновки. Останній слайд може містити за темою кваліфікаційної роботи перелік власних публікацій у наукових журналах, тез доповідей, авторських свідоцтв, патентів тощо.

До презентації виносять оригінальні (власні) результати розробок та досліджень – фотографії, рисунки, схеми, кресленики, формули, графіки, таблиці тощо. На слайдах бажано уникати подання інформації у вигляді суцільного тексту або, навпаки, занадто обмежено та стисло. Також не можна використовувати матеріал, запозичений (сканований) з літературних джерел.

Для комфортного сприйняття презентації необхідно дотримуватися ряду основних вимог:

- дизайн має бути простим і лаконічним (краще сприймається інформація на світлому однотонному фоні; фон, перевантажений рисунками або емблемами, відволікає увагу від основної інформації);
- кожен слайд повинен мати заголовок;
- слайди мають бути пронумеровані із зазначенням загальної кількості слайдів в презентації через похилу риску (1/23, 2/23 і тощо). Так ви дасте можливість аудиторії зрозуміти, скільки залишилось до кінця;

– завершувати презентацію слід коротким резюме (висновками), яке містить основні положення зробленої роботи, важливі дані, які озвучувалися в доповіді.

2.3 Вимоги до обсягу кваліфікаційної роботи бакалавра

Залежно від тематики та варіанта реалізації АР, вимоги до її обсягу та оформлення ілюстративного матеріалу можуть бути різними.

Рекомендований загальний обсяг пояснювальної записки 50-60 сторінок формату А4 в редакторі Microsoft Word (шрифт Times New Roman, розмір 14, інтервал 1,5).

Графічна частина має ілюструвати та доповнювати основні розділи КР. Конкретний перелік обов'язкових графічних матеріалів визначається керівником і вказується в завданні на кваліфікаційну роботу бакалавра.

Кількість креслеників, що розробляються одним студентом, не повинна бути меншою встановленої кількості графічного матеріалу. Загальний обсяг графічного матеріалу 3-6 аркушів. Форматів А3-А1 (за узгодженням з керівником роботи).

Зазначені обсяги в певних межах корегуються керівником бакалаврської кваліфікаційної роботи.

3 ЗАГАЛЬНА СТРУКТУРА ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Кваліфікаційна робота бакалавра має складатися з пояснювальної записки (текстової частини) та графічної частини. Матеріал пояснювальної записки та графічної частини – це результат самостійної творчої роботи студента з питань, сформульованих у завданні на КР бакалавра.

Пояснювальна записка складається з таких частин:

- титульний аркуш (додаток Ж);
- завдання на КР бакалавра (додаток А);
- реферат;
- перелік умовних позначень, скорочень, термінів;
- зміст;
- основна частина;
- висновки;
- перелік джерел посилань;
- додатки;
- відомість КР бакалавра (додаток З).

Порядок викладення матеріалів має відображати логічну послідовність досліджень та розрахунків за темою кваліфікаційної роботи. Зміст пояснювальної записки визначається завданням на КР.

Зміст розділів пояснювальних записок кваліфікаційних робіт, виконаних за комплексною тематикою, визначається їх призначенням в процесі вирішення як загальних, так і конкретних задач розробки. Причому в пояснювальних записках таких робіт, виконаних різними студентами, не допускається текстуального збігу в матеріалах загального призначення.

Основні вимоги щодо структурних елементів пояснювальної записки, а також деяких розділів, що входять до її змістовної частини наведено нижче.

Титульний аркуш та *завдання* оформляються на бланках, які видає кафедра (див. додатки Ж та А). Завдання є основним документом кваліфікаційної роботи. Воно оформляється керівником КР на спеціальному бланку (додаток А), в якому вказані: прізвище, ім'я та по батькові керівника кваліфікаційної роботи, тема КР, вихідні дані, перелік основних питань, які підлягають розробці.

У завданні наводиться також перелік графічного матеріалу, а також календарний план виконання КР.

Тема КР у завданні повинна коротко визначати об'єкт(и) або задачу розробки. Якщо кваліфікаційна робота комплексна, то тема записується таким чином: *КТ 1. Назва загальної теми кваліфікаційної роботи. ПТ 1. Назва теми, що виконується індивідуально кожним здобувачем.* (Комплексна тема (номер за порядком). Назва загальної теми КР. Підтема (номер за порядком). Назва індивідуальної теми).

Реферат – стислий виклад змісту пояснювальної записки, який містить основні фактичні відомості та висновки, необхідні для початкового ознайомлення з кваліфікаційною роботою бакалавра. Він має містити відомості щодо обсягу пояснювальної записки, кількості ілюстрацій, таблиць, використаних джерел. Має бути конкретно вказано: об'єкт дослідження, методи дослідження, мету та задачі роботи, використані технічні та програмні засоби, практичні рекомендації щодо використання отриманих результатів і перелік ключових слів.

Перелік ключових слів, які є визначальними для розкриття суті роботи, має містити 5-15 слів (словосполучень). Рекомендовано подавати їх перед текстом реферату великими літерами в рядок із прямим порядком слів у називному відмінку однини, розташованих за абеткою та розділених комами.

Реферат наводиться державною та однією іноземною мовами. Обсяг реферату – не більше однієї сторінки. Приклад оформлення реферату подано у додатку И.

До змісту бакалаврської роботи включають: перелік умовних позначень, символів, одиниць, скорочень і термінів; вступ; послідовно перелічені назви всіх розділів, підрозділів, а також пунктів і підпунктів, якщо вони мають заголовки; висновки; перелік посилань; назви додатків та номери сторінок, на яких розміщується початок матеріалу. Розривати слова знаком переносу у змісті не рекомендовано.

Заголовки змісту мають точно повторювати заголовки в тексті. Скорочувати заголовки у змісті або наводити їх в іншій редакції порівняно із заголовком у тексті не можна. Усі заголовки у змісті починають з великої літери без крапки в кінці. Відповідний номер сторінки роботи наводять у крайньому правому положенні в рядку.

У вступі стисло викладають: оцінку сучасного стану проблеми; світові тенденції вирішення поставлених задач; актуальність теми бакалаврської роботи; мету бакалаврської роботи; задачі, які необхідно вирішити під час виконання роботи. Рекомендований обсяг вступу – 1...2 сторінки.

У висновках необхідно зазначити: результат і повноту виконання завдання на бакалаврську роботу, аналіз досягнутих кількісних та якісних показників; отримані наукові результати, можливість використання матеріалів кваліфікаційної роботи у галузі, а також рекомендації щодо використання результатів кваліфікаційної роботи у навчальному процесі в університеті та подальшої роботи в даному напрямку.

Перелік джерел посилання, на які посилаються в пояснювальній записці, має бути наведений у кінці тексту після висновків перед додатками, починаючи з нової сторінки. У переліку джерел посилання бібліографічні описи подають у порядку, за яким джерела вперше згадуються у тексті ПЗ. Порядкові номери бібліографічних описів у переліку джерел мають відповідати посиланням на них у тексті (номерні посилання).

Бібліографічні описи джерел у переліку наводять згідно з ДСТУ ГОСТ 7.1:2006.

Джерела, на які є посилання лише в додатку, наводять в окремому переліку, який розміщують у кінці цього додатка. Приклади оформлення бібліографічного опису посилань наведено у додатку К.

У *додатках* розміщують матеріали, які є необхідними для повноти розуміння бакалаврської роботи, але включення їх в основну частину роботи може змінити упорядковане та логічне подання роботи. Зокрема, в додатки виносять блок-схеми алгоритмів, структурні схеми та схеми електричні принципові розроблених пристроїв, код розроблених програм, презентаційні матеріали. Останнім додатком має бути відомість кваліфікаційної роботи, в яку вноситься перелік документів та виробів (макетів, дослідних зразків), що входять до складу виконаної роботи. Відомість кваліфікаційної роботи виконується на аркуші формату А4 за формою, наведеною у додатку З.

У відомості вказується умовне позначення документа, згідно з вимогами стандарту ЄСКД про позначення виробів та конструкторських документів.

Номенклатура конструкторських документів, яка використовується в роботі, містить коди документів згідно з діючими стандартами.

Позначення конструкторських документів має складатися з позначення виробу та коду документа (див. рис. 3.1).

Наприклад, загальне позначення пояснювальної записки може бути таким: ГЮИК.ХХХХХХ.013ПЗ, де символами Х позначено номер за класифікаційною характеристикою згідно з темою кваліфікаційної роботи.

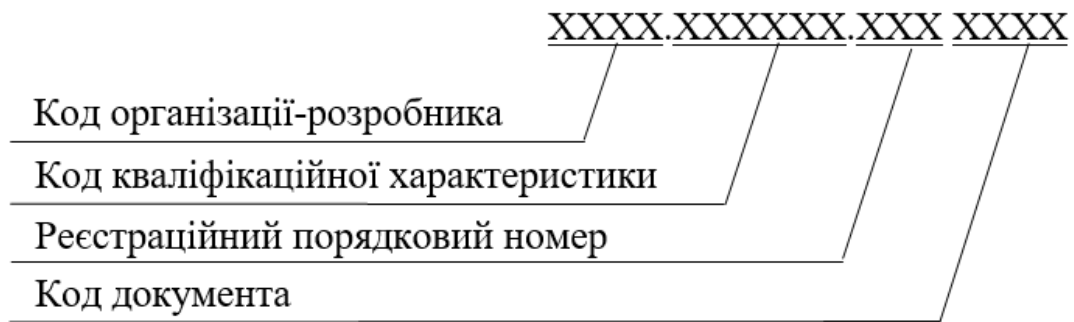


Рисунок 3.1 – Позначення конструкторських документів

Для ХНУРЕ встановлено код організації-розробника – ГЮОИК. Код кваліфікаційної характеристики обирають згідно з класифікатором ЄСКД. Структура коду має вигляд, поданий на рис. 3.2.

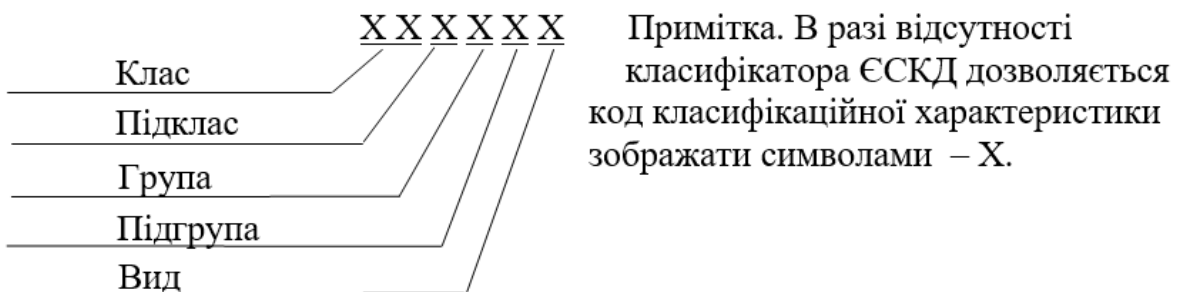


Рисунок 3.2 – Структура коду кваліфікаційної характеристики

Деякі коди виробів і програмних документів наведено у додатку Л.

Коди конструкторських документів згідно зі стандартами, які згадані вище, є такими:

- | | |
|------------------------------------|-------|
| – схема електрична структурна | – E1; |
| – схема електрична функціональна | – E2; |
| – схема електрична принципова | – E3; |
| – схема електрична розміщення | – E7; |
| – відомість кваліфікаційної роботи | – ВД; |
| – кресленик загального вигляду | – ОВ; |
| – складальний кресленик | – СБ. |

4 РЕКОМЕНДАЦІЇ ЩОДО ВИРІШЕННЯ ОСНОВНИХ ЗАВДАНЬ КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРІВ СПЕЦІАЛЬНОСТЕЙ 151 «АВТОМАТИЗАЦІЯ ТА КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ» ТА 174 «АВТОМАТИЗАЦІЯ, КОМП'ЮТЕРНО- ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА РОБОТОТЕХНІКА» ОСВІТНЬОЇ ПРОГРАМИ «СИСТЕМНА ІНЖЕНЕРІЯ»

4.1 Вибір теми кваліфікаційної роботи

Теми кваліфікаційних робіт бакалавра розробляються випускаючою кафедрою, щорічно оновлюються і затверджуються на засіданні кафедри. При цьому враховуються вимоги стандарту вищої освіти і освітньо-професійної програми, тенденцій розвитку комп'ютерних технологій, напрям науково-дослідної роботи кафедри, реальних держбюджетних і госпдоговірних замовлень. Студент має право вибрати тему кваліфікаційної роботи із встановленого переліку або запропонувати свою тему за умови доцільності її розробки.

Тематика може передбачати автоматизацію виробничих процесів, розроблення нових і вдосконалення існуючих систем автоматизації із застосуванням сучасних програмно-технічних комплексів, технічних засобів автоматизації і інформаційних технологій; проектування систем управління сучасними виробництвами; розробку програмного забезпечення, що орієнтоване на використання технології Інтернету речей та хмарних обчислень.

У кваліфікаційній роботі можуть бути розглянуті такі питання:

- системи автоматизації або їх компоненти;
- інтелектуалізовані засоби технологічного оснащення автоматизації, контролю, діагностики сучасного виробництва;
- математичне, програмне, інформаційне, алгоритмічне технічне забезпечення процесів автоматизації та засобів робототехніки;
- методи, засоби проектування, виготовлення, налагодження, виробничих випробувань і наукових досліджень тощо.

Тематика КР має на меті розв'язання типових задач діяльності, призначених для максимально ефективного забезпечення створення систем та пристроїв автоматизації, використання комп'ютерно-інтегрованих технологій та розробки програмного забезпечення для розв'язання завдань ГІС:

- розробка програмно-технічного та інформаційного забезпечення автоматизованої системи управління (АСУ);

- проектування конструкції вузла промислового (мобільного) робота або робототехнічного пристрою в складі ГІС;
- розробка автоматичної (автоматизованої) системи контролю (АСК) гнучкої інтегрованої системи;
- розробка підсистеми підтримки прийняття рішень інтелектуального промислового/мобільного робота;
- розробка підсистеми керування промисловим/мобільним роботом або структурним елементом ГІС;
- проектування пристрою керування промисловим/мобільним роботом або структурним компонентом ГІС;
- розробка сенсорної системи (або окремих елементів) роботизованих пристроїв ГІС;
- розробка програмного забезпечення сенсорних систем маніпуляційних / мобільних роботів у складі ГІС;
- розробка елементів (підсистем) системи автоматизованого проектування керуючих програм для обладнання з числовим програмним керуванням
- розробка підсистеми діагностики промислового робота або структурного елемента ГІС;
- розробка систем автоматизованого проектування технологічних процесів ГІС.

4.2 Вибір об'єкту автоматизації

Об'єкти автоматизації (об'єкти управління) – це окремі пристрої, верстати, агрегати, апарати, комплекси машин та апаратів, якими потрібно керувати. Вони дуже різноманітні за своїм призначенням, будовою і принципом дії.

Об'єкт автоматизації є основною складовою автоматичної системи, який визначає характер системи, тому його вивченню приділяється особлива увага. Складність об'єкта визначається головним чином ступенем його вивченості та різноманіттям виконуваних ним функцій. Результати досліджень об'єкта мають бути представлені у вигляді чітких рекомендацій щодо можливості повної або часткової автоматизації об'єкта або відсутності необхідних умов для автоматизації.

Проектуванню автоматичної системи управління має передувати вивчення об'єкта з метою встановлення зв'язків об'єкта. У загальному випадку ці зв'язки можна подати у вигляді чотирьох груп змінних (рис. 4.1).

Перша група – контрольовані впливи, що збурюють, сукупність яких утворює L-мірний вектор $H = h_1, h_2, h_3, \dots, h_L$. До них належать вимірювані змінні, що залежать від зовнішнього середовища, і фактори, які впливають на процес. Для контрольованих впливів, що збурюють, накладаються обмеження за умовами технології.

Показник технологічного процесу, яким необхідно керувати, називають керованою величиною (координатою), а ту фізичну величину, за допомогою якою керується показник технологічного процесу, називають керуючим впливом (вхідною величиною, координатою).

Друга група змінних – керуючі дії, сукупність яких утворює N-мірний вектор $X = x_1, x_2, x_3, \dots, x_N$, незалежні від зовнішнього середовища і мають найбільш істотний вплив на технологічний процес. За їх допомогою цілеспрямовано змінюють перебіг процесу. Прикладом керуючих впливів є увімкнення і вимкнення електродвигунів, електричних нагрівачів, виконавчих механізмів, положення задатчиків регуляторів тощо.

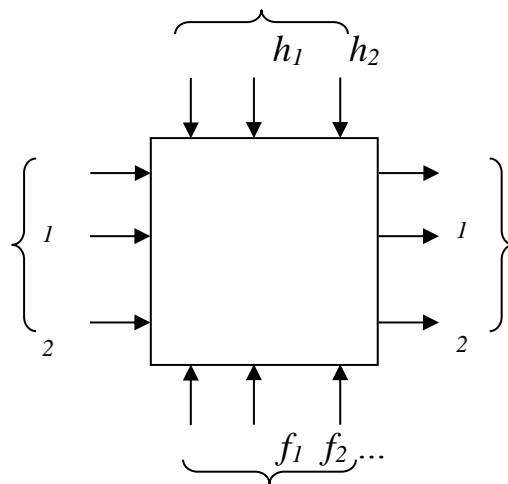


Рисунок 4.1 – Входи та виходи об’єкта автоматизації

Наступна група – вихідні змінні, сукупність яких утворює M-мірний вектор стану $Y = y_1, y_2, y_3, \dots, y_M$, є виходом об’єкта, що характеризує його стан і визначає якісні показники готового продукту.

І остання, четверта група – неконтрольовані впливи, що збурюють, сукупність яких утворює G-мірний вектор $F = f_1, f_2, f_3, \dots, f_G$. До них належать такі збурення, які не можуть бути виміряні з тих чи інших причин, наприклад, через відсутність датчиків.

Вивчення розглянутих зв’язків об’єкта, що підлягає автоматизації, може призвести до отримання двох діаметрально протилежних висновків: між

вихідними та вхідними змінними об'єкта є строга математична залежність або між цими змінними відсутня залежність, яка може бути виражена достовірною математичною формулою.

У теорії і практиці автоматичного управління технологічними процесами накопичено достатній досвід опису стану об'єкта у таких ситуаціях. При цьому об'єкт розглядається як одна з ланок системи автоматичного управління. У випадках, коли відома математична залежність між вихідною змінною y і керуючим вхідним впливом x об'єкта, виділяють дві основні форми запису математичних описів – це статичні та динамічні характеристики об'єкта.

Статична характеристика математичної або графічної форми виражає залежність вихідних параметрів від вхідних. Бінарні зв'язки зазвичай мають чіткий математичний опис, наприклад, статична характеристика ступіні деформації пружних елементів має вигляд $h = k \cdot m$ (де m – маса матеріалу; k – коефіцієнт пропорційності, що залежить від властивостей матеріалу пружного елемента).

За наявності кількох змінних параметрів як статичні характеристики можна використовувати номограми.

Статична характеристика об'єкта визначає подальше формування цілей автоматизації. З боку практичної реалізації у ливарному виробництві ці цілі можуть бути зведені до трьох видів:

- стабілізації вихідних параметрів об'єкта;
- зміни вихідних параметрів за заданою програмою;
- зміни якості деяких вихідних параметрів при зміні умов протікання процесу.

Низка технологічних об'єктів однак не може бути описана математично через безліч взаємопов'язаних факторів, що впливають на перебіг процесу, наявності неконтрольованих факторів та недостатньої вивченості процесу. Такі об'єкти з точки зору автоматизації складні. Ступінь складності визначається кількістю входів та виходів об'єкта. Такі об'єктивні проблеми виникають під час вивчення процесів пов'язаних з масо- і теплопередачею. Тому їх автоматизація вимагає припущень чи умов, які мають сприяти головній меті автоматизації – підвищення ефективності управління шляхом максимального наближення технологічних режимів до оптимальних.

Для вивчення складних об'єктів використовують метод, який полягає в умовному поданні об'єкта у вигляді чорного ящика. Досліджують лише зовнішні зв'язки, внутрішню структуру системи не враховують, тобто вивчають, що робить об'єкт, а не його функціонування.

Поведінку об'єкта визначають за реакцією вихідних величин на зміну вхідних. Основним інструментом вивчення такого об'єкта є статистико-математичні методи. Методичне вивчення об'єкта здійснюється наступним чином: визначають головні параметри, встановлюють дискретний ряд змін основних параметрів, штучно змінюють вхідні параметри об'єкта в межах встановленого дискретного ряду, фіксують всі зміни виходів і статистично обробляють отримані результати.

Динамічні характеристики об'єкта автоматизації визначаються низкою його властивостей, деякі з них сприяють якісному процесу управління, інші перешкоджають йому.

З усіх властивостей об'єктів автоматизації, можна виділити основні: ємність, здатність до самовирівнювання і запізнювання.

Ємністю називають здатність об'єкта акумулювати робоче середовище та запасати його всередині об'єкта. Накопичення речовини або енергії можливе завдяки тому, що в кожному об'єкті є опір виходу.

Мірою ємності об'єкта служить коефіцієнт ємності C , що характеризує кількість речовини або енергії, яку потрібно підвести до об'єкта, щоб змінити керовану величину на одиницю прийнятої розмірності вимірювання:

$$C = \frac{\Delta Q}{dp_y/dt},$$

де ΔQ – різниця між припливом та витратою речовини або енергії;

p_y – керований параметр;

t – час.

Розмірність коефіцієнта ємності може бути різною залежно від розмірності керованих параметрів.

Швидкість зміни керованого параметра зменшується зі збільшенням коефіцієнту ємності об'єкта. З цього випливає, що легше керувати об'єктами, які мають більші коефіцієнти ємності.

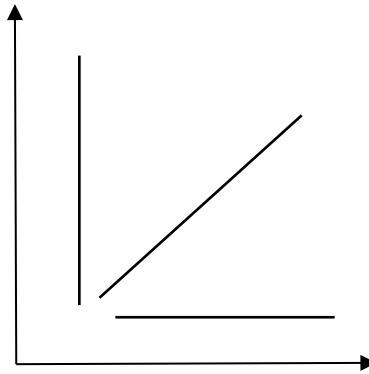
Самовирівнювання – це здатність об'єкта приходити після збурення в новий стан без втручання керуючого пристрою (регулятора). Об'єкти з самовирівнюванням називають статичними, а ті, що не мають цю властивість – нейтральними або астатичними. Самовирівнювання сприяє стабілізації керуючого параметра об'єкта та полегшує роботу керуючого пристрою.

Самовирівнювання об'єктів характеризується коефіцієнтом (ступенем) самовирівнювання, який має вигляд:

$$\rho = \frac{d(\Delta\lambda)}{dy}$$

Залежно від коефіцієнта самовирівнювання статичні характеристики об'єкта набувають різного вигляду (рис. 4.2).

Залежність 1 характеризує об'єкт, у якого керована величина не змінюється при будь-яких збуреннях. Такий об'єкт не потребує керуючих пристроїв. Залежність 2 показує нормальне самовирівнювання об'єкта. Залежність 3 характеризує об'єкт з самовирівнюванням. Коефіцієнт ρ – змінна величина, яка підвищується зі збільшенням навантаження і в більшості випадків має додатне значення.



1 – ідеальне самовирівнювання; 2 – нормальне самовирівнювання;
3 – відсутність самовирівнювання

Рисунок 4.2 – Залежність керованого параметра від навантаження (відносного збурення) за різних коефіцієнтів самовирівнювання

Запізнення – це час, що проходить між моментом порушення рівноваги та початком зміни керованої величини об'єкта. Воно обумовлюється наявністю опорів та інерційністю системи.

Розрізняють два види запізнення: чисте (або транспортне) та перехідне (або ємнісне), які у сумі становлять загальне запізнення в об'єкті.

Чисте запізнення отримало свою назву тому, що в об'єктах, де воно є, спостерігається зсув у часі реакції на виході об'єкта порівняно з моментом появи вхідного впливу без зміни величини та форми впливу. Мінімальне чисте запізнення має об'єкт, який працює при максимальному навантаженні або в якому сигнал поширюється з великою швидкістю.

Перехідне запізнення виникає у разі подолання потоком речовини чи енергії опорів між ємностями об'єкта. Воно визначається кількістю ємностей та величиною перехідних опорів.

Чисте та перехідне запізнення погіршують якість управління, тому необхідно прагнути зменшення їх величин. До заходів, що сприяють цьому, можна віднести розміщення вимірювальних та керуючих пристроїв у безпосередній близькості від об'єкта, застосування малоінерційних чутливих елементів, конструктивна раціоналізація самого об'єкта тощо.

Результати аналізу найбільш важливих характеристик і властивостей об'єктів автоматизації, а також методів їх вивчення дозволяють сформулювати низку вимог та умов, виконання яких забезпечує можливість успішної автоматизації.

Головні з них такі:

- математичний опис зв'язків об'єкта, представлений у вигляді статичних показників. Для складних об'єктів, які не можуть бути описані математично, це використання математико-статистичних, табличних, просторових та інших методів вивчення зв'язків об'єкта, що ґрунтуються на введенні деяких припущень;

- побудова динамічних характеристик об'єкта, таких як диференціальні рівняння або графіки з метою вивчення перехідних процесів в об'єкті. При цьому враховують всі основні властивості об'єкта (ємність, запізнювання, самовирівнювання);

- використання в об'єкті таких технічних засобів, які б забезпечували видачу інформації про зміну всіх параметрів об'єкта, що цікавлять, у вигляді уніфікованих сигналів, що вимірюються датчиками;

- використання виконавчих механізмів, які мають керовані приводи, для управління об'єктом;

- встановлення достовірно відомих меж зміни зовнішніх збурень об'єкта.

До підлеглих вимог можна віднести:

- визначення граничних умов автоматизації відповідно до завдань управління;

- встановлення обмежень вхідних величин та керуючих впливів;

- розрахунок критеріїв оптимальності (ефективності).

4.3 Аналіз та вибір мови програмування та програмна реалізація автоматизованої системи тестування роботи WEB-додатків приладобудівних підприємств

Java є мовою програмування та платформою обчислень, яка була вперше випущена Sun Microsystems в 1995 р. Існує безліч додатків і веб-сайтів, які не працюють при відсутності встановленої Java, і з кожним днем число таких веб-сайтів і додатків збільшується. Java відрізняється швидкістю, високим рівнем захисту та надійністю.

Java – це мова програмування загального призначення, який слід парадигмі об'єктно-орієнтованого програмування і підходу «Написати один раз і використовувати скрізь». Java використовується для настільних, мережевих, мобільних і корпоративних додатків. Java – це не тільки мова програмування, а й екосистема інструментів, що охоплює майже всі, що може знадобитися при програмуванні на Java. До неї входять:

- Java Development Kit (JDK) – комплект розробника Java. За допомогою JDK і стандартного блокнота можна писати і запускати / компілювати код на Java.

- Java Runtime Environment (JRE) – виконуюча система Java. Механізм розповсюдження програмного забезпечення, складається з автономної віртуальної машини Java, стандартної бібліотеки Java (Java Class Library) і інструментів настройки.

- Integrated Development Environment (IDE) – інтегроване середовище розробки. Інструменти, які допомагають запускати, редагувати і компілювати код.

Java включає в себе об'єктно-орієнтоване програмування (ООП) – концепцію, в якій ви не тільки визначаєте тип даних і його структуру, а й набір функцій, що застосовуються до нього. Таким чином, структура даних стає об'єктом, яким можна управляти для створення відносин між різними об'єктами.

При іншому підході – процедурному програмуванні – потрібно слідувати чітким інструкціям, використовувати змінні та функції. При ООП можна групувати ці змінні і функції за допомогою контексту, маркувати їх і посилатися на функції в контексті кожного конкретного об'єкта.

Java – це мова високого рівня, тобто він схожий на людську мову. На відміну від мов низького рівня, які нагадують машинний код. Мови високого рівня перетворюється за допомогою компіляторів або інтерпретаторів. Це

спрощує розробку, роблячи мову більш легким для написання, читання і обслуговування.

Основні переваги Java:

– Гнучкість. Java довела, що C – процедурний, керований вручну і залежить від платформи код – це не межа досконалості. Завдяки Java, все більше людей почали застосовувати об'єктно-орієнтоване програмування, яке зараз використовується повсюдно.

– Аплети. Ще до появи JavaScript, в Java додали аплети – невеликі веб-програми, які надають інтерактивні елементи для візуалізації та навчання. Вони не використовуються ні для чого, крім простої анімації, однак аплети привернули увагу багатьох програмістів і підштовхнули їх до розробки HTML5, Flash і JavaScript.

– Розробка через тестування. Java TDD – вже давно не експериментальна практика, а стандартний спосіб розробки програмного забезпечення. Введення JUnit 2000 року вважається одним з найбільших досягнень Java.

– Безпека. Існує думка, що Java – безпечна мова, проте це не зовсім так. Сама мова не захищає вас від вразливостей, але деякі його функції усувають поширені уразливості. По-перше, на відміну від C, в Java немає покажчиків. Покажчик – це об'єкт, який зберігає адресу осередки пам'яті іншого значення, що може викликати несанкціонований доступ до пам'яті. По-друге, в Java є Security Manager, створена для кожної програми політика безпеки, в якій можна вказати правила доступу.

– Незалежність від платформи. «Написати один раз і використовувати скрізь» (WORA) – популярна в IT-сфері фраза, за допомогою якої Sun Microsystems описує крос-платформні можливості Java.

Maven – інструмент для автоматизації збирання проєктів. З ним працюють в основному Java-розробники, хоча є плагіни для інтеграції з C / C ++, Ruby, Scala, PHP і іншими мовами.

Головна перевага зовнішніх складальників для проєктів – гнучке управління залежностями. Maven вміє довантажувати в свій локальний репозиторій сторонні бібліотеки, вибирати необхідну версію пакету, обробляти транзитивні залежності. Приклад підключення зовнішніх бібліотек в файлі .pom зображений на рис. 4.1.

```

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.ihor_kovalov.java</groupId>
  <artifactId>FinalTask</artifactId>
  <version>1.0-SNAPSHOT</version>
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <serenity.version>2.0.39</serenity.version>
    <serenity.cucumber.version>1.9.31</serenity.cucumber.version>
    <webdriver.driver>chrome</webdriver.driver>
    <maven.surefire.plugin.version>2.19.1</maven.surefire.plugin.version>
    <testFailureIgnore.value>true</testFailureIgnore.value>
  </properties>
  <dependencies>
    <dependency>
      <groupId>net.serenity-bdd</groupId>
      <artifactId>serenity-core</artifactId>
      <version>${serenity.version}</version>
    </dependency>
    <dependency>
      <groupId>net.serenity-bdd</groupId>
      <artifactId>serenity-cucumber</artifactId>
      <version>${serenity.cucumber.version}</version>
    </dependency>
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-simple</artifactId>
      <version>1.7.7</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>

```

Рисунок 4.1 – Підключення залежностей в .pom файлі

Selenium WebDriver – це програмна бібліотека для управління браузерами. Часто вживається також більш коротку назву WebDriver.

Іноді кажуть, що це «драйвер браузера», але насправді це ціле сімейство драйверів для різних браузерів, а також набір клієнтських бібліотек на різних мовах, що дозволяють працювати з цими драйверами.

Як вже було сказано, WebDriver є сімейство драйверів для різних браузерів плюс набір клієнтських бібліотек для цих драйверів на різних мовах програмування.

В рамках проекту Selenium розробляються драйвери для браузерів Firefox, Internet Explorer і Safari, а також драйвери для мобільних браузерів Android та iOS.

Основна задумка бібліотеки Selenium WebDriver – це запуск автоматизованих тестів за допомогою спеціалізованого «драйвера», який є інструментом для управління відповідним браузером.

Для спрощення роботи з програмою та ухилення від необхідності вручну завантажувати веб-драйвер, була застосована бібліотека Boni Garcia.

Boni Garcia – WebDriverManager, який зберігає в собі всі існуючі веб-драйвери для відповідних браузерів.

Як згадувалося вище, BDD – розширення підходу TDD до розробки та тестування, при якому особлива увага приділяється поведінці системи / модуля в термінах бізнесу (замовника).

Однак BDD не може існувати без певного «перетворювача» та мови до нього, якими є Cucumber та Gherkin відповідно.

Cucumber – це інфраструктура, що дозволяє подолати прірву між розробниками додатків і бізнес-менеджерами. Тести пишуться простою мовою, де BDD зі стандартним набором Коли / Тоді зрозумілий будь-якому користувачеві.

Cucumber інтерпретує тести у певну програмну мову, в той час як Selenium використовується для контролю тестів в браузері. Cucumber працює як програмна платформа, яка повинна бути використана в зв'язці з Selenium.

Gherkin – це назва мови, який зрозумілий для Cucumber, та людини. Основний синтаксис мови Gherkin, полягає в простих «словах-спілках». Given / When / Then відповідають параметрам Умова / Дія / Результат.

Також може бути проміжне слово And, яке інтерпрітується як слово «І». Приклад тестового сценарію зображений на рис. 4.2.

```
Scenario: Find exists product
  Given Customer opens Eliz page
  When Customer writes a product 43101
  And User click button search
  Then Product has founded
```

Рисунок 4.2 – Приклад тестового сценарію, який використовує мову Gherkin в .feature файлі

Створимо алгоритм роботи програмного забезпечення (рис. 6.3).

Для створення програмного забезпечення було обрано мову Java, інтеграційну бібліотеку Serenity-BDD, пагін Cucumber for Java та Gherkin for Java.

Спершу підключити необхідні додатки за допомогою файла pom.xml, та підключити їх з сайту mvnrepository.com. Цей сайт являється базою даних всіх існуючих бібліотек, які підтримуються Maven збирачем.

Після підключення необхідних залежностей, створюється файл CucumberRubber.class – це точка входу до нашої програми (рис 4.3).

Написання програми починається зі створення файлу .feature.

Файл такого типу, є збірником тестових сценаріїв. За допомогою мови Gherkin створюємо тестовий сценарій:

Scenario:	Open	History	page	of	the	factory
Given	Customer	opens	Eliz	page		
When	Customer	chooses	History	catalog		
Then	History catalog is opened					

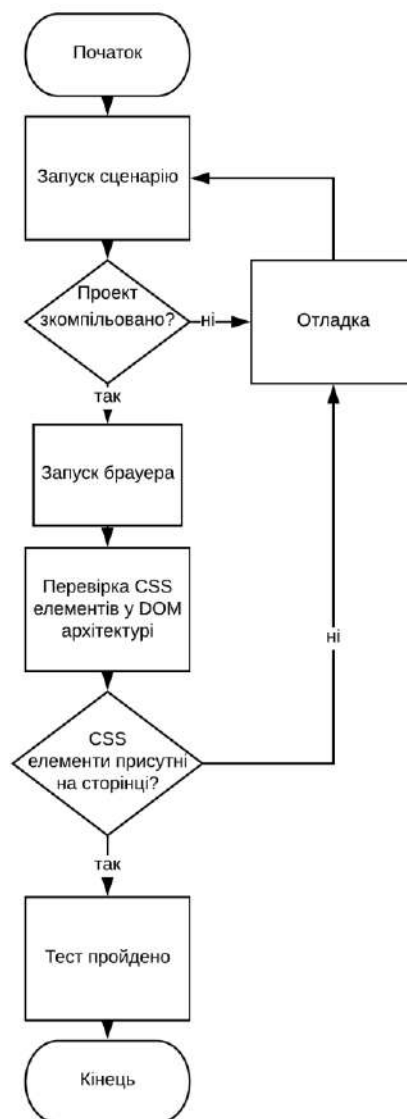


Рисунок 4.3 – Алгоритм роботи програмного забезпечення

```

import cucumber.api.CucumberOptions;
import net.serenitybdd.cucumber.CucumberWithSerenity;
import org.junit.runner.RunWith;

@RunWith(CucumberWithSerenity.class)
@CucumberOptions(
    plugin = {"html:target/cucumber-report/smoketest", "json:target/cucumber.json"},
    features = {"src/test/resources/features/"})

public class CucumberRunner {
}

```

Рисунок 4.4 – Точка входу до програми

Мета цього сценарію – перевірка здатності веб-додатку перейти до гіперпосилання History.

Спершу пишеться крок відкриття головної сторінки веб-додатку Eliz:

Given Customer opens Eliz page

Після опису початкового кроку, описується внутрішня логіка роботи цього самого кроку.

```

@Steps
private LoginSteps loginSteps;

@Given("^Customer opens Eliz page$")
public void customerOpensElizPage() {
    loginSteps.openPage();}

```

Анотація @Given вказує Cucumber менеджеру на те, що цей крок є початком (точкою входу до тестового сценарію) теста. Далі у дужках йде короткий опис кроку, це робиться для того, щоб JDK «розуміло» де буде реалізація цього методу.

Анотація @Steps – створює об'єкт класу LoginSteps, де буде описуватися подальша логіка сценарію.

У реалізації кроку відкриття Eliz сторінки визиваємо об'єкт класу LoginSteps та його метод openPage().

```

private final static String URL = "http://www.eliz.com.ua/";

```

```

@Step
void openPage() {
    open(URL);}

```

У реалізації метода openPage(), викликаємо метод open(параметр), який в свою чергу є методом бібліотеки codeborne.selenium з подальшою реалізацією:

```

public static void open(String relativeOrAbsoluteUrl) {
    WebDriverRunner.getSelenideDriver().open(relativeOrAbsoluteUrl);}

```

WebDriverManager створює об'єкт класу Driver, таким чином передає внутрішні налаштування, для подальшого управління вебдрайвером та браузером. Для методу open(), передаємо статичну змінну, де зберігається URL адреса. Перший крок відкриття сторінки – виконано.

When Customer chooses History catalog

Створення кроку «дії». Мета цього кроку – перейти за допомогою UI-інтерфейсу до сторінки History.

За допомогою DOM-архітектури веб-сторінки, та CSS-локаторів, намагаємося описати алгоритм дії нашого скрипта.

На рис. 4.5 зображена необхідна кнопка, та DOM-архітектура.



Рисунок 4.5 – Зображення кнопки «Історія» та DOM-архітектура веб-додатку

Пишемо реалізацію метода переходу таким самим чином, як описували метод роботи переходу на головну сторінку.

```

    @When("^Customer chooses History catalog$")
public void customerChoosesHistoryCatalog() {
    loginSteps.redirectToHistory();
}

```

Анотація `@When` вказує на те, що цей крок є «дією», що необхідно зробити `WebDriverManager`'у. Викликаємо метод `redirectToHistory()`.

```

private static final Logger logger = LogManager.getLogger(LoginSteps.class);

public void redirectToHistory() {
    logger.info("Redirecting to history content");
    loginPage.redirectToHistory();
}

```

У реалізації методу `redirectToHistory()`, є пряме посилання на логіку праці цього кроку, та рівень логування, який ми додали до проекту, за допомогою сторонньої бібліотеки. Створюємо об'єкт класу `Logger`, та визиваємо метод `LogManager.getLogger(.class)`. У параметрі цього методу, передаємо назву класу, в якому описуємо логіку. Після створення `LogManager`'а, додаємо рівень логування та необхідне повідомлення, яке у майбутньому, у випадку помилки, буде легше зрозуміти, де виникла помилка. Об'єкт `loginPage`, є посиланням на клас `LoginPage`:

```
private LoginPage loginPage = new LoginPage();
```

У нижчому рівні реалізації тестових кроків, знаходяться CSS-локатори та їх обробка:

```

private final SelenideElement companyButton = $(By.cssSelector(".sf-with-ul"));
private final SelenideElement historyButton =
$(By.xpath("//a[contains(@title,'История')]"));

public void redirectToHistory() {
    companyButton.hover();
    historyButton.isDisplayed();
    historyButton.click();}

```

CSS-локатори зберігаються у змінних формату `SelenideElement`. Знак «\$» вказує на те, що це локатор відповідає за простий об'єкт на веб-сторінці (кнопка, посилання, рисунок і т.д.). Існує ще символ «\$\$» – символ який вказує на те, що локатор зберігає в собі «лист» даних. Прикладом існування таких локаторів, є результат пошуку на інтернет-сайтах за продажом різноманітних товарів.

`companyButton` – локатор який вказує на елемент «Предприятие», який ми можемо бачити на барі головної сторінки веб-додатку.

Метод `hover()` – наводить «курсор» на цю кнопку, та починає одержувати її там. Це зроблено для того, щоб уникнути проблему, коли випадючий список може зникнути за особливостей роботи.

`historyButton` – локатор, який вказує на посилання «История».

Метод `isDisplayed()` – булевий метод, який перевіряє: чи відображається цей локатор зараз на веб-сторінці? Якщо так, він повертає відповідь «True», якщо ні, повертає відповідь «False», та завершує виконання програми з помилкою. Якщо було повернуто відповідь «True», виконується наступний метод.

Метод `click()` – передає команду `WebDriver`'у, клікнути на кнопку, що призведе до переходу на сторінку «История».

Другий крок переходу на іншу сторінку – виконано.

Останній крок нашого сценарію – перевірка, що дійсно відкрилася сторінка «История».

Then History catalog is opened

За тим самим алгоритмом створюємо реалізацію методу.

```
@Then("^History catalog is opened$")
public void historyCatalogIsOpened() {
    loginSteps.isHistoryPageOpen();}
```

Анотація `@Then` вказує на те, що цей крок є перевіркою «чогось». Так само визиваємо метод `isHistoryPageOpen()`. Бачимо знайому реалізацію методу, але с деякими нововведеннями:

```
public void isHistoryPageOpen() {
    logger.info("History page is opened");
    Assert.assertTrue(loginPage.isHistoryPageOpened());}
```

Assert – об'єкт класу Assert, який «вшитий» у бібліотеку Junit.

Junit – інструмент розроблений для запуску та обробки коду, призначений для написання тестових сценаріїв. Без цього інструменту, запуск будь-якого тесту – неможливий.

Викликаємо метод `assertTrue(параметр)`. Далі у параметрі, виклакається метод `isHistoryPageOpened()`. Його реалізація виглядає таким чином:

```
public          boolean          isHistoryPageOpened()          {  
return historyPage.isDisplayed();}
```

`historyPage` – знайомий за механікою локатор. Визиваємо метод `isDisplayed`, який повертає нам відповідь «True». Метод має повертаєму змінну `boolean`, та за допомогою методу `return`, «пробрасує» значення «True» до вищого рівня реалізації. Тепер параметр методу `assertTrue` має такий вигляд:

```
Assert.assertTrue(True);
```

Метод `assertTrue` перевіряє – чи внутрішній параметр у стані «True»? Якщо так – повертає «зелене світло» и перевірка стає коректною, якщо ні – завершую роботу програми з помилкою. На цьому тестовий сценарій закінчено.

4.4 Аналіз існуючих приладів та систем контролю присутності

При вирішенні завдань охорони об'єктів різного призначення в якості датчиків виявлення і контролю присутності людини (порушника) використовуються прилади, засновані на різних фізичних принципах побудови. При цьому людина визначається як біологічний об'єкт з певними параметрами, заданими середньостатистичними значеннями в залежності від використовуваного методу контролю.

Ряд методів контролю присутності людини дозволяє визначити напрямок руху, швидкість руху і деякі інші параметри, необхідні для здійснення ефективно охорони об'єктів. Крім того, цифрові методи обробки отриманої інформації дозволяють здійснювати в результатах контролю присутності фільтрацію наявності інших біологічних об'єктів, наприклад, птахів і дрібних ссавців. Якщо використовувати сучасні бездротові технології і девайси, якими користується повсякденно, то можна уникнути випадкових спрацьовувань системи.

Датчик присутності (або датчики руху) – це електронний прилад(сенсор), реєструючий безконтактними методами об'єкти певного

класу на території свого контролю. Залежно від результатів реєстрації він може комутувати електричні імпульси, за сигналами яких інші пристрої виконують різного роду дії. Прилади, які допомагають виявити наявність переміщення теплового об'єкта. Вони використовуються для того, щоб вчасно встановити присутність людини в приміщенні або ж забезпечують включення освітлювальних приладів.

Датчик руху або присутності застосовується в сигналізаціях (в тому числі й автомобільних), системах контролю доступу, розумного будинку і просто для управління вуличним освітленням.

Досить підключити спеціальний пристрій в ланцюг електропостачання, яка живить освітлювальне обладнання, і якщо в поле дії датчика з'явиться якийсь пересувається об'єкт, він відразу відправить команду на включення світла в потрібній області. Вимикається світло через певний проміжок часу, коли об'єкт залишає оброблювану зону дії або зупиняється. Головною функцією, яку має будь-який сучасний датчик присутності, є забезпечення комфорту [13].

Основними типами приладів уловлювання присутності є:

- ультразвукові;
- фотоелектричні;
- ємнісні;
- акустичні;
- інфрачервоні;
- датчики навантаження;
- комбіновані датчики.

Ультразвуковий датчик досліджує простір використовуючи звукові хвилі, які миттєво реагують на появу стороннього предмета. Такий пристрій відображає будь-які рухи і не піддається впливу зовнішніх факторів, однак його випромінювання можуть бути шкідливі для тварин. Також, прилад такого типу можна обдурити, оскільки він вловлює тільки різкі рухи. Інтелектуальні ультразвукові датчики використовуються в пристроях для визначення рівня наповнення ємності, зберігають в пам'яті сигнал відображення при порожній ємності. При цьому зберігаються всі відбиті сигнали від спеціальної оснастки, яка вмонтована в ємність, наприклад, спіралі нагрівання, мішалки, аварійні трапи. Якщо рівень наповнення ємності змінюється, то відбиті сигнали порівнюються з початковим значенням (рис. 4.6).



Рисунок 4.6 – Ультразвуковий датчик

Отже, ультразвукові датчики гарно підходять для моніторингу певної області, де знаходяться рухомі об'єкти, але рух самих об'єктів повинен бути відчутним, щоб сенсор його зафіксував. Виходячи з цього, подібні прилади погано підійдуть для чіткого контролю виробничої території.

Фотоелектричні датчики використовуються в багатьох галузях промисловості для забезпечення точного виявлення об'єктів без фізичного контакту.

У більшості основних форм фотоелектричний датчик може розглядатися як пристрій типу кінцевого перемикача, в якому функцію механічного приводу або плеча важеля виконує промінь або світло. Фотоелектричні датчики працюють, виявляючи зміни в інтенсивності світла, який або відбивається, або затримується, якщо виявляється об'єктом (мішенню). Зміни в інтенсивності світла можуть бути результатом присутності або відсутності мішені або результатом зміни розміру, форми, коефіцієнта відбиття або кольору мішені (рис. 4.7).



Рисунок 4.7 – Фотоелектричний датчик

Отже, фотоелектричні датчики не підходять для точного контролю присутності виробничої території тому, що необхідно ідентифікувати людей, а до інформаційного поля будуть потрапляти різні об'єкти, які не будуть нести особистих проблем.

Такі пристрої широко використовуються в багатьох галузях. Ємнісні датчики застосовуються в системах контролю різних виробничих процесів. Вони дозволяють визначити рівень заповнення резервуарів, виконують функцію вимикачів на автоматизованих лініях, верстатах і конвеєрах.

Сьогодні активно використовуються датчики наближення. Такі пристрої мають великий функціонал. Вони сигналізують про рівень заповнення скляних і пластикових ємностей і прозорих упаковок, регулюють натяг стрічки та інше. Вданий час найбільш активно використовуються датчики кутових і лінійних переміщень. Їх застосовують в машинобудівній, енергетичній та інших промислових галузях (рис. 4.8).



Рисунок 4.8 – Ємнісні датчики (датчики об'єму)

Отже, ємнісні датчики не вийде використовувати для зовнішніх замірів і отримання інформації про навколишнє середовище, так як вони розраховані більше для контролю технологічних процесів на виробництві.

У акустичних датчиках за допомогою п'єзоелектричних матеріалів відбувається перетворення звукової хвилі в електричний сигнал. Використовуються як звукові датчики світла, що працюють спільно з реле часу і економлять електроенергію. При перевищенні порога шуму в кімнаті відбувається автоматичне включення світла. Якщо настає тиша, від 20 с до 25 с лампи вимикаються.

У акустичних сенсорів проста конструкція (рис. 4.9), яку можна втілити до різних розумних, інтелектуальних приладів, що також можна віднести до переваг приладу. Також до переваг можна занести надійність.

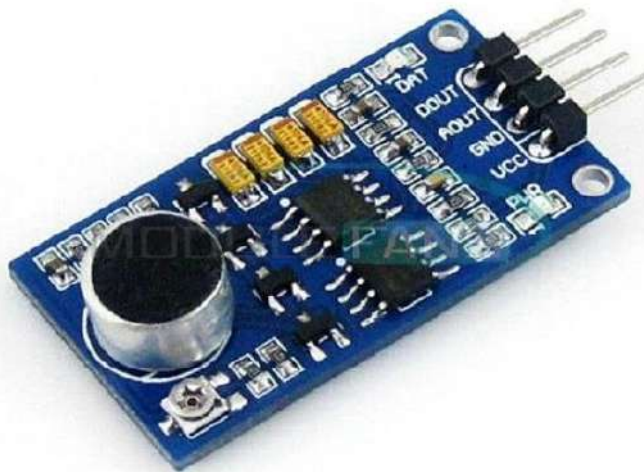


Рисунок 4.9 – Акустичний датчик

Але у даного датчика є свої недоліки, із-за яких не вийде використати його у системі контролю руху: необхідність використання підсилювачів, ймовірність помилкових спрацьовувань в результаті зовнішніх і внутрішніх шумів (різкі звуки з вулиці, включення радіо, телефонні дзвінки), тому для проєкту модулю обробки інформації він не підходить.

Принцип дії приладів заснований на фіксації змін потоку інфрачервоних (ІЧ) променів в результаті переміщень людини. Його перебування розпізнається по більшій інтенсивності (в порівнянні з предметами інтер'єру) випромінювання, яке безпосередньо залежить від температури тіла. Основні деталі датчика – фотоелементи і мультилінза, що складається з великого числа сегментів - маленьких лінз (рис. 4.10). Кожна з них направляє потрапляють в неї промені на фотоелемент. Рухаючись, людина виявляється в зонах контролю різних сегментів. Світло на фотоелементі то зникає, то з'являється, генеруючи електричний сигнал.

У строгому розумінні за принципом дії такий пристрій – датчик руху, а не присутності. До останньої категорії відносять особливо точні прилади з великим числом ділянок контролю. Вони здатні вловлювати наявність людини, що знаходиться в стані майже повного спокою. Фіксуються дрібні жести: похитування головою, натискання пальцями клавіатури тощо.



Рисунок 4.10 – Інфрачервоний датчик

Радіус виявлення – основна характеристика пристрою. Його монтаж повинен проводитися так, щоб відстань до найдальших кутків кімнати не перевищувало радіус. У приміщеннях великої площі потрібна установка декількох датчиків.

Необхідно, щоб на шляху ІЧ променя не було перегородок, навіть скляних, які для нього непрозорі. Неприпустимо попадання на прилад прямого світла ламп, він повинен знаходитися на максимальній відстані від вентиляторів, кондиціонерів та обігрівачів.

Інфрачервоні датчики використовують як засоби охоронної сигналізації, в якості додаткового обладнання для відеоспостереження і для автоматизації електропостачання, що приводить до економії коштів.

До переваг можна віднести точність регулювання, повну безпеку для здоров'я через відсутність будь-яких видів випромінювання, а також реакція тільки наоб'єкти, температура яких перевищує порогову межу.

До недоліків можна віднести неточність функціонування на відкритих просторах (вплив опадів, сонячного світла), ймовірність помилкових включень під впливом теплих потоків повітря, перешкоди від об'єктів, що не пропускають ІЧ випромінювання, а також низький діапазон робочих температур.

Датчики навантаження – це конвертери, які перетворюють механічне зусилля в електричний струм. Конструктивно датчик являє собою тензорезистор у вигляді тонкого дроту, зигзагоподібно, як обігрівач автомобільного скла, закріплений на еластичною підкладці (рис. 4.11). Як пружний елемент використовується тканина, гума, полімерна плівка. Під дією сили провідник деформується, опір його змінюється, що генерує електричний сигнал, що подається після посилення на виконавчі пристрої.



Рисунок 4.11 – Датчик навантаження

Отже, датчики навантаження не підходять для чіткого контролю присутності. Його конструкція підвержена модифікуванню, необхідність використання підсилювача сигналу, зниження чутливості при перепадах температур, а також схильність багаторазово повторюється механічних навантажень, що призводить до виходу з ладу.

Комбінований тип включає в себе одночасно кілька властивостей, наприклад, інфрачервоне та мікрохвильове виявлення (рис. 4.12). Це допомагає досягти найбільш точного результату. Іноді для досягнення поставлених цілей пристрої одного типу недостатньо. У таких випадках їх можна задіяти кілька з різними принципами роботи.

На прикладі розглянемо експлуатацію інфрачервоного датчика присутності в комбінації з датчиком освітленості. Перший подає сигнал на включення ламп у разі виявлення ним людини в кімнаті. Другий – в разі показників освітленості нижче встановленого порогового значення. Працюючи спільно, вони в автоматичному режимі запалюють світильники тільки в разі присутності людей в приміщенні в темний час доби.

Такий підхід створює комфортні умови життєдіяльності та ще призводить від 30 % до 40 % економії електроенергії.

Отже, подібні датчики та прилади гарно підходять для інтеграції з системою контролю пропуску. На основі комбінованих датчиків будується основоположна взаємодія головного МК, тобто хаба, з периферійними модулями та сенсорами для отримання точної та якісної інформації про навколишнє середовище.



Рисунок 4.12 – Комбінований датчик

4.5 Аналіз конструкцій маніпуляторів

Одним із дієвих засобів автоматизації виробничих процесів вважається створення систем, що моделюють властивість рук людини і відтворюють їх робочі функції. Прикладом такої системи вважається промисловий маніпулятор. Він складається з приладу управління та виконавчого механізму маніпулятора, утворюючи єдиний комплекс з технологічним обладнанням.

При виконанні виробничих операцій роботам треба виконувати переміщення всіляких предметів: переносити дрібниці з верстата на верстат, направляти їх в потрібні міста, переміщати інструмент важливою траєкторією і зі строго певною швидкістю, з високою точністю виконувати установку предмета. Потрібну рухливість роботу має гарантувати виконавчий пристрій.

Для прикладу можемо розглядати маніпулятор, який має деяку кількість ступенів рухливості. Вертикальний рух маніпулятора, обертання колони відносно вертикальної осі, поступальний рух висування руки, обертання схвату.

Робоча зона маніпуляторів орієнтується довжинами ланок і системами кінематичних пар.

Провідними якостями оцінки промислових маніпуляторів вважаються: коефіцієнт обслуговування, точність позиціонування, швидкодія.

За допомогою кута і коефіцієнта обслуговування розглядають ймовірність кутових рухів в робочій зоні.

Точність роботи маніпулятора розглядають за допомогою позиціонування помилки-промаху.

Необхідною рисою роботи робота, вважається швидкість виконання даних операцій.

Класифікація руки робота по конструкції заснована на системі координат трьох головних осей, які забезпечують вертикальний хід, хід всередину і назовні, обертальний або поперечне рух щодо вертикальної осі робота.

Розрізняють такі систем координат:

– циліндрична. Горизонтальна рука робота може висуватися і вдвигатися паралельно підставі, рухатися вгору і вниз по вертикальній стійці (залишаючись паралельною підставі), а підстава може повертатися разом з рукою і стійкою навколо вертикальної осі, утворюючи робочу зону, що має циліндричну форму (рис. 4.13);

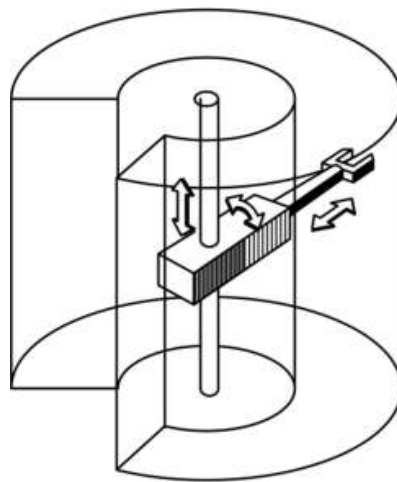


Рисунок 4.13 – Циліндрична система координат руху робота.

– сферична (полярна) система координат. Рука робота може висуватися, вдвигатися і обертатися на підставі (як і в попередньому випадку), але замість поступального реалізується обертальний вертикальний рух, утворюючи в просторі сферу (рис. 4.14);

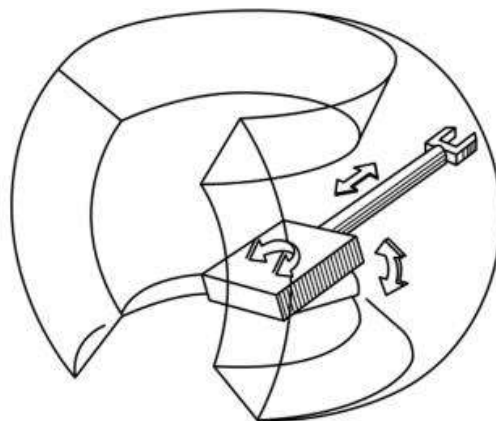


Рисунок 4.14 – Сферична система координат руху робота.

– прямокутна (декартова) система координат. Цей тип робота має три взаємно перпендикулярними осями переміщень. До його складу входить стійка,

яка рухається вгору і вниз, і поперечна балка, до якої підвішена стійка. Стійка може рухатися вліво і вправо. Балка, в свою чергу, здатна переміщатися вперед і назад. Таким чином забезпечується рух по осях x , y і z (рис. 4.15);

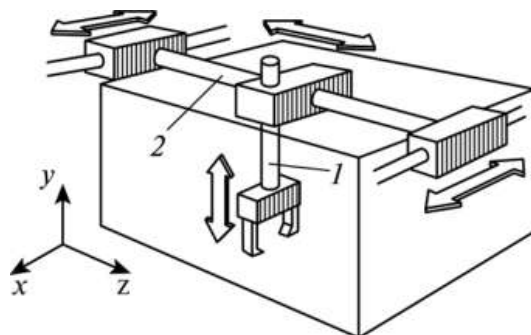


Рисунок 4.15 – Прямокутна система координат руху робота.

– кутова система координат. Він складається з здатних обертатися сполук, які називаються плечем і ліктем закріплених на стійці, що складається з обертової підстави, яке забезпечує третю ступінь свободи (рис. 4.16).

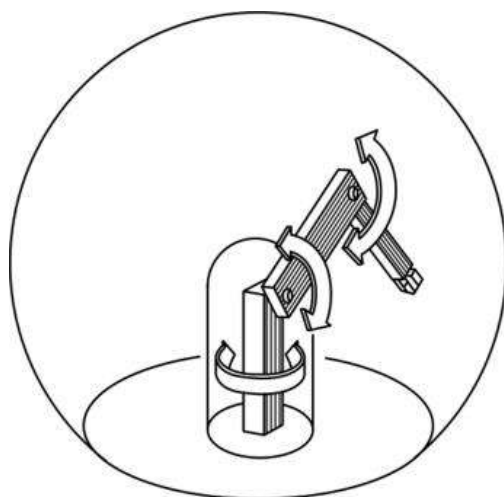


Рисунок 4.16 – Кутова система координат руху робота.

Декартові (лінійні, порталні) промислові роботи-маніпулятори порівняно прості по кінематиці і надійні в експлуатації. Ця конструкція складається з декількох лінійних осей переміщення, розташованих перпендикулярно один до одного.

Картезіанська конструкція – найпростіший варіант стаціонарної системи.

Завдяки малій кількості шарнірів і зчленувань така система забезпечує високу жорсткість конструкції, що при точних роботах дає високі показники точності. Наприклад, роботи цього типу при довжині робочого ходу 2500 мм, здатні точно повторювати однотипні рухи в межах 0,05 мм. Поступальні

кінематичні пари забезпечують мінімальність навантаження на кінематичні елементи і, отже, високу зносостійкість, кінематичну стабільність і довговічність. Простота конструкції зумовлює низьку вартість обладнання.

Мінуси даної конструкції – не найзручніша форма обслуговується простору, низьке співвідношення обслуговуємого і займаного роботом просторів, істотно обмежені можливості орієнтування інструмента, і, як наслідок, неможливість виконання робіт в місцях з обмеженим доступом.

Даний тип роботів досить широко застосовується в цілях автоматизації обладнання для переробки пластичних мас. Безпосередньо для застосування на ливарних машинах вони були розроблені на початку 1980-х рр., і до теперішнього часу їх електричні та пневматичні приводи значною мірою стандартизовані. Промислові роботи цього типу легко інтегруються в систему управління литтєвою машиною.

Колінно-важільні роботи відносяться до маніпуляторів в сферичній системі координат.

Основні особливості колінно-важільного робота:

- наявність тільки обертальних рухів від серводвигунів, сукупність яких обумовлює просторову траєкторію руху схвата;
- робочий об'єм має сферичну форму;
- більш високе (в порівнянні з лінійним роботом) можливе число ступенів свободи схвата;
- ускладнене програмування і обслуговування робота.

Задана траєкторія руху схвата досягається за рахунок шести обертальних рухів робочої руки на контрольовані кути повороту. При цьому кожен рух здійснюється за допомогою окремого, автономного сервоприводу з високою швидкодією, завдяки чому забезпечується практично миттєва зупинка схвата в необхідній позиції.

Необхідно враховувати, що управління рухом осей колінно-важільних роботів набагато складніше, ніж лінійних роботів, тому обслуговування і усунення несправностей колінно-важільних роботів найчастіше можливо тільки за допомогою фахівців-виробників роботів. Необхідна більш висока кваліфікація персоналу і значні витрати на програмування збільшують вартість використання колінно-важільних роботів. Так, наприклад, витрати на програмування і адаптацію колінно-важільних роботів до литних машин приблизно в три рази вище, ніж для лінійних роботів.

Щоб колінно-важільний робот виконував рух схвата зі збереженням просторового положення, все його шість осей повинні регулюватися одночасно.

Таким чином реалізується управління траєкторією, яка вважається ідеальною при допусках на відхилення в міліметровому діапазоні.

Така траєкторія руху вимагає від колінно-важільного робота значно більшої продуктивності в порівнянні з лінійним роботом.

До цієї групи належать шарнірні промислові роботи маніпулятори. За принципом дії цей тип маніпулятора нагадує людську руку. У його кінематичній системі присутні не менше трьох поворотних з'єднань, які утворюють полярну систему координат. Основні шарніри забезпечують поворот маніпулятора, нахил в плечовому з'єднанні і згинання в лікті (осі 1 – 3); ще 3 поворотні осі (4 – 6) і одне панорамне зчленування забезпечують додаткові ступені свободи (наклон, кутовий рух відносно вертикальної осі, кутовий рух щодо головної поперечної осі інерції), що дозволяє руці вибирати будь-які напрямки і вільно витягуватися по радіусу в будь-якому напрямку.

Відмінна риса цієї конструкції – дуже висока гнучкість, що дозволяє роботу обходити будь-які перешкоди. Рука може приймати будь-яке положення всередині обслуговуваної зони. Крім того, робот цього типу досить компактний, і зона яку він обслуговує набагато більше, ніж займане роботом місце. Як правило, типовий шарнірний маніпулятор має 5 або 6 програмованих ланок. Незважаючи на очевидні переваги конструкції, управляти таким роботом досить складно.

Головною перевагою даної конструкції є паралельне з'єднання зчленувань маніпулятора, в результаті чого рука може досить вільно рухатися по горизонталі, зберігаючи при цьому вертикальну жорсткість. Важливо, що конструкція, що складається з двох ланок, може витягнутися, розпрямивши лікоть, а може згорнутися, звільнивши займаний простір. Це зручно в робочій зоні з обмеженим простором, і під час роботи, коли деталі переміщуються з одного виробничого модуля в інший. Зона обслуговування такого робота має вигляд сегмента полого криволінійного циліндра.

Промислові роботи цього типу мають ряд переваг: велика швидкість пересування по горизонталі за рахунок підсумовування щодо вертикальних осей; висока точність позиціонування; особлива жорсткість конструкції щодо вертикальної осі; велика зона обслуговування; компактність.

У виробництві виробів з пластмас промислові роботи такого типу використовуються для транспортування рулонів плівки масою 5 кг – 25 кг, труб діаметром до 500 мм, листів шаруватих пластиків розміром до 2000 мм х 4000 мм, листів жорсткого полівінілхлориду, рулонів лінолеуму, тощо, беруть участь в заключних операціях.

У промисловості часто зустрічаються операції, де робот повинен взяти деталь, перенести в потрібному напрямку і опустити в інше місце. Для виконання таких операцій були розроблені паралельні стрижневі роботи.

Суть їх конструкції така: до базової установки кріпляться чотири маніпулятора, з'єднані паралельно, утворюючи кінематичну схему паралелограма (це така схема, при якій два паралельно з'єднаних важеля рухаються синхронно, коректуючи напрям один одного; приклад такої системи – спиця парасольки). Особливістю цієї системи є те, що вихідна ланка пов'язана з базою декількома кінематичними ланцюгами, які паралельно повідомляють інструменту рух. В результаті в кожного кінематичного ланцюга є вільні від приводів зчленування, на які можна встановити різні датчики, а також додаткові приводи. Такий механізм дозволяє паралельно управляти швидкістю і зусиллям вихідної ланки по одній координаті. Паралельна структура має високу точність, що дозволяє використовувати маніпулятор для виконання операцій, що вимагають високої точності (вимірювальні роботи). Маніпулятор має малу вагу, тому здатний працювати на надшвидких швидкостях.

В даний час широко поширені апарати, об'єднані під загальною назвою «дельта-роботи», які представляють собою типову паралельну конструкцію: три маніпулятори, що приводяться в рух карданною передачею, кріпляться до бази, розташованої вгорі у вигляді підвісної конструкції; маніпулятори, що сходяться вниз, з'єднані невеликою трикутною платформою, яка під час роботи зміщується по осях X, Y або Z, а центральний, четвертий важіль дає додатковий ступінь свободи – обертальну. Завдяки тому що приводи встановлені в основі базової конструкції, а «руки» виконані з легкого композитного матеріалу, робот може здійснювати до 150 захоплень в хвилину. Найчастіше такі роботи використовуються для упаковки і сортування виробів.

В даний час класифікація промислових роботів за типом конструкції стає все більш умовною. Все частіше різні конструкції компонуються один з одним – шарнірні маніпулятори ставляться на декартові конструкції, а декартові роботи обзаводяться додатковими ступенями свободи.

Як відомо, сучасні промислові роботи маніпулятори мають набір датчиків, що дозволяють їм відслідковувати розташування робочого органу і кожної ланки руки механізму в просторі в будь-який момент часу, а також контролювати ряд технологічних операцій (наприклад, при лазерному різанні металу неприпустимий фізичний контакт робочого органу і поверхні оброблюваного матеріалу).

Інформація, що надходить від датчиків, одразу надходить в бортовий комп'ютер, який і віддає коригувальні сигнали, або зовсім зупиняє виробництво, якщо який-небудь з параметрів вийшов за рамки встановлених меж.

З точки зору техніки датчик – це первинний перетворювач (аналогового сигналу в дискретний-електронний). Функція пристосування – вимір деякого параметра, в сукупності з обчислювальним центром і системою зворотного зв'язку – в усуненні відхилень, контрольованого їм параметра. Так само датчик може сигналізувати про зміну контрольованого параметра, передавати його в обчислювальний процесор, попередньо посиливши сигнал, тощо

В даний час під цим поняттям ховається сукупність всіх систем, що працюють в єдиній зв'язці керуючими пристроями і каналами передачі сигналів.

Зазвичай під датчиком мається на увазі пристрій для вимірювання деякого параметра навколишнього середовища: температури, вологості, атмосферного тиску або тиску робочого тіла (рідини або газу).

Такі датчики можуть використовуватися в окремих елементах конструкції промислового робота (наприклад, в гідравлічній або пневматичній системі), але, як правило, не є пристосуваннями, що безпосередньо впливають на роботу системи, а тільки контролюючими та регулюючими якісь внутрішніх параметрів. Передбачається, що тиск, вологість і температура повітря в зоні роботи знаходиться на прийнятному рівні.

Однак, цим переліком не обмежується типаж використовуваних пристроїв. Сюди потрапляють і контролери радіоактивності, вібрації. Подібні пристрої потрібні практично повсюди, вони вважаються універсальними приладами і призначені для контролю над загальними параметрами. Вони забезпечують загальні умови роботи.

Сюди відносяться, в першу чергу, датчики переміщення – їх основне призначення полягає в контролі над лінійними переміщеннями ланок руки робота. Так само існує спеціальні пристрої за контролем переміщення заготовки, від найпростіших фотоелементів, що повідомляють про факт перетину заготівлею лінії, до складних відеокамер з пристроями розпізнавання обрисів об'єкта.

У разі якщо заготовка в своєму становищі має неприпустимі відхилення, то управляючий пристрій подає сигнал або про зміну орієнтації деталі (якщо є така можливість), або про припинення виробництва і зупинці технологічного процесу.

Такі пристрої дозволяють уникнути випуску шлюбу, істотно підвищуючи якість продукції, що випускається і знижуючи витрати від зіпсованих заготовок.

Так само датчики температури і тиску можуть бути вкрай корисними безпосередньо в робочій зоні, де не допускається істотне перевищення температури і тиску над встановленим максимумом, інакше неминуче пошкодження заготовки.

Нарешті, існують прилади, що відстежують факт контакту робочого органу і поверхні заготовки. Останнє може привести до псування не тільки заготовки, але і робочого інструмента промислового маніпулятора, що обіцяє додаткові витрати.

Система управління сучасного маніпулятора складається з декількох підсистем, що виконують певні інформаційні, керуючі, захисно-попереджувальні, обмежувальні та інші функції.

Розрізняють три основних види управління: циклове, позиційне, контурне. Циклове управління програмує послідовність виконання рухів і умови початку і закінчення рухів; становище, до якого йде рух, задаються на самому маніпуляторі, а не в програмі; швидкість переміщення визначається характеристиками приводу і також не задається в програмі. При позиційному управлінні команди подаються так, що переміщення робочого органу походить від точки до точки, причому положення точок задаються програмою. Швидкість переміщення між точками не контролюється і не реалізується. При контурному управлінні рух робочого органу відбувається по заданій траєкторії і задається швидкістю. У програмі задаються самі траєкторії і режими руху. Контурне управління використовується в технологічних роботах.

Пристрій управління та інші блоки системи управління при цикловому, позиційному і контурному управліннях можуть бути реалізовані на однакових або різних принципах і елементних базах. В особливий вид зазвичай виділяється адаптивне управління, при якому здійснюється автоматична зміна керуючих програм. Зокрема адаптація або пристосованість системи управління може полягати в тому, що пристрої системи управління за допомогою спеціальних датчиків визначають конфігурацію об'єкта і його положення.

Об'єктом управління є виконавчий пристрій (маніпулятор плюс пристрій пересування, якщо він є). У виконавчий пристрій також входять приводи. Все інше обладнання робота призначене для формування і видачі керуючих впливів виконавчому пристрою. Таким чином, пристрій управління отримує сигнали (від датчиків) і видає сигнали (на приводи маніпулятора).

Для пульта ручного керування основними є зв'язки з пристроєм управління. З пульта ручного керування можуть здійснюватися введення програм, настройка. На пульт управління надходять сигнали про виконання

різних рухів, а також про можливі порушення режимів роботи та про відмови. Слід мати на увазі, що в пристрій управління зазвичай надходять сигнали від зовнішніх (по відношенню до робота) датчиків і систем (наприклад, від систем управління устаткуванням, що обслуговується). Пристрій управління роботом також може бути пов'язаний з ЕОМ, яка координує роботу декількох одиниць обладнання, наприклад всього обладнання технологічного ділянки або лінії. У цих випадках ця ЕОМ як би знаходиться на більш високому поверсі, на наступному рівні управління. Така багаторівнева система управління типова для сучасних гнучких виробничих систем.

Robotino – це автономний мобільний робот, який пересувається на трьох роликівих колесах типу "omni wheel" («omnidirectional» – всепрямований).

У рух, робота приводять три двигуни постійного струму, осі яких розташовані під кутом 120 градусів один до одного.

Така особливість конструкції характерна для мобільних платформ на базі всепрямованих коліс, і, завдяки їй, робот може переміщатися в усіх напрямках, а так само повертатися навколо своєї осі.

Так як Robotino, який представлений на рис. 4.17, призначається, головним чином, для навчання, то він виконаний модульно; всі технічні компоненти (електроприводи, датчики, камера), можна відключити від робота і вивчити окремо.



Рисунок 4.17 – Робот Robotino

Підсистеми Robotino:

- корпус робота – сталевий корпус з бампером, що забезпечує легкий доступ до батарей, двигунів і портів робота;

- підсистема живлення – дві акумуляторні батареї (12 В / 5 Ач кислотно-свинцевий (lead-acid) або 12 В / 9 Ач NiMH), що дозволяє роботу працювати в автономному режимі кілька годин;

– рухова підсистема – три двигуни постійного струму, редуктор і ремінна передача на роликові колеса, що дозволяють роботу рухатися в різних напрямках.

Для стабілізації швидкостей обертання валів двигунів робота, використовуються вбудовані ПД-регулятори з попередньо встановленими коефіцієнтами;

– підсистема одометра – мінкрементні енкодери на валах двигунів;

– підсистема введення/виведення – плата, яка виконує комунікаційний зв'язок між комп'ютером робота і його датчиками, двигунами і інтерфейсом вводу / виводу;

– підсистема бездротового зв'язку із зовнішнім керуючим комп'ютером (Wi-Fi точка доступу);

– бортовий обчислювач – одноплатний промисловий комп'ютер формату PC / 104 + (має шини ISA і PCI) з процесором на 300 МГц. У якості пам'яті, використовується картка Compact Flash (1024 MB), на яку встановлена вбудована версія операційної системи Linux;

– різні датчики робота – 9 інфрачервоних датчиків відстані (на бампері), USB web-камера).

Інтерфейс представлений на рис. 4.18.



Рисунок 4.18 – Інтерфейси робот Robotino

Robotino підтримує інтерфейси:

- USB;
- Ethernet;
- VGA;
- 8 цифрових портів введення-виведення;
- 8 аналогових портів введення.

Управляти роботом Robotino, можна як за допомогою програм, записаних в пам'ять робота, так і дистанційно – за допомогою бездротової технології Wireless LAN (WLAN).

Крім того, можливо автономне програмування робота, через підключення до нього монітор і клавіатури.

У комплекті з роботом йде програма Robotino View – програмне забезпечення для інтерактивного графічного програмування робота.

Робот може бути запрограмований в графічному режимі – простим вибором необхідних функціональних блоків з наявного списку і їх подальшим комбінуванням і налаштуванням (візуальне програмування).

Програмне забезпечення робота можна розділити на дві частини: внутрішнє програмне забезпечення і зовнішнє.

Внутрішньо ПЗ робота – це його операційна система (Linux), яка здійснює взаємодію програмного і апаратного забезпечення робота, обробку внутрішніх команд і обмін даними з зовнішнім керуючим комп'ютером.

Команди управління, що надходять на робота з зовнішнього програмного забезпечення, виконуються послідовно і тому робот не може виконувати кілька команд одночасно.

Тобто після передачі відповідної команди роботу, програма управління припиняється і чекає поки робот не виконає задану дію.

4.6 Аналіз та обґрунтування основних складових маніпулятора

Структурна схема маніпулятора представлена нижче (рис. 4.19).

У ході аналізу було виявлено основні компоненти, які потрібні для збірки робота-маніпулятора:

- системна плата;
- серводвигуни;
- плата розширення;
- модуль Bluetooth.

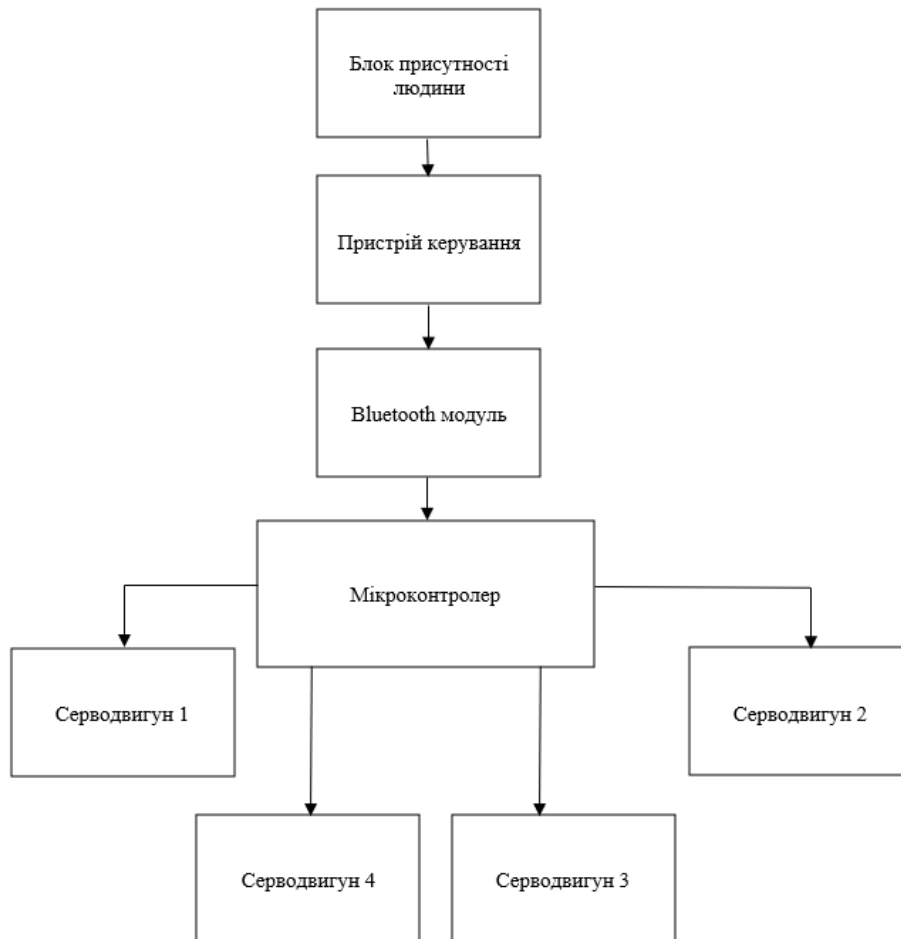


Рисунок 4.19 – Структурна схема маніпулятора

У якості системної плати було обрано плату Arduino Uno R3 (рис. 4.20).



Рисунок 4.20– Плата Arduino Uno R3

Arduino Uno – це пристрій на основі мікроконтролера ATmega328. До його склад входять все необхідне для зручної роботи з мікро контролером: 14 цифрових входів / виходів (з них 6 можуть використовуватися в якості ШІМ-виходів), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм для внутрішньо схемного програмування (ICSP) і кнопка

скидання. Для початку роботи з пристроєм досить просто подати живлення від AC / DC-адаптера або батарейки, або підключити його до комп'ютера за допомогою USB-кабелю.

Характеристики:

- мікроконтролер – ATmega328;
- робоча напруга – 5 В;
- напруга живлення (рекомендована) – від 7 В до 12 В;
- напруга живлення (граничне) – від 6 В до 20 В;
- цифрові входи / виходи – 14 (з них 6 можуть використовуватися в якості ШИМ-виходів);
- фналогові входи – 6;
- максимальний струм одного виведення – 40 мА;
- максимальний вихідний струм виводу 3,3V – 50 мА;
- Flash-пам'ять – 32 КБ з яких 0,5 КБ використовуються загрузчиком SRAM – 2 КБ;
- EEPROM – 1 КБ;
- тактова частота – 16 МГц.

Arduino Uno може бути живиться від USB або від зовнішнього джерела живлення – тип джерела обирається автоматично.

В якості зовнішнього джерела живлення (не USB) може використовуватися мережевий AC / DC-адаптер або акумулятор / батарея. Штекер адаптера (діаметр – 2,1 мм, центральний контакт – позитивний) необхідно вставити у відповідний роз'єм живлення на платі. У разі живлення від акумулятора / батареї, її проводу необхідно під'єднати до висновків Gnd і Vin роз'єму POWER.

Напруга зовнішнього джерела живлення може бути в межах від 6 В до 20 В. Однак, зменшення напруги живлення нижче 7 В призводить до зменшення напруги на виводі 5V, що може стати причиною нестабільної роботи пристрою. Використання напруги більше 12 В може призводити до перегріву стабілізатора напруги і виходу плати з ладу. З огляду на це, рекомендується використовувати джерело живлення з напругою в діапазоні від 7 В до 12 В.

Нижче перераховані виводи живлення, розташовані на платі:

– VIN. Напруга, що надходить в Arduino безпосередньо від зовнішнього джерела живлення (не пов'язане з 5 В від USB або іншим стабілізованою напругою). Через цей вивід можна як подавати зовнішнє живлення, так і споживати струм, коли пристрій живиться від зовнішнього адаптера;

– 5V. На вивід надходить напруга 5 В від стабілізатора напруги на платі, поза незалежності від того, як живиться пристрій: від адаптера (від 7 В до 12 В),

від USB (5 В) або через вивод VIN (від 7 В до 12 В). Живити пристрій через виводи 5 В або 3,3 В не рекомендується, оскільки в цьому випадку не використовується стабілізатор напруги, що може привести до виходу плати з ладу;

- 3.3V. 3,3 В, що надходять від стабілізатора напруги на платі. Максимальний струм, споживаний від цього висновку, становить 50 мА;

- GND, виводи землі;

- IOREF. Цей вивод надає платам розширення інформацію про робочій напрузі мікроконтролера Arduino. Залежно від напруги, ліченого з виведення IOREF, плата розширення може переключитися на відповідний джерело живлення або задіяти перетворювачі рівнів, що дозволить їй працювати як з 5 В, так і з 3,3 В пристроями.

Об'єм флеш-пам'яті ATmega328 становить 32 КБ (з яких 0,5 КБ використовуються загрузчиком). Мікроконтролер також має 2 КБ пам'яті SRAM і 1 КБ EEPROM (з якої можна зчитувати або записувати інформацію за допомогою бібліотеки EEPROM).

З використанням функцій `pinMode ()`, `digitalWrite ()` і `digitalRead ()` кожен з 14 цифрових висновків може працювати в якості входу або виходу. Рівень напруги на висновках обмежений 5 В. Максимальний струм, який може віддавати або споживати один вивод, становить 40 мА. Всі виводи пов'язані з внутрішніми підтягуються резисторами (за замовчуванням відключеними) номіналом від 20 кОм до 50 кОм. Крім цього, деякі виводи Arduino можуть виконувати додаткові функції:

- послідовний інтерфейс: виводи 0 (RX) і 1 (TX). Використовуються для отримання (RX) і передачі (TX) даних по послідовному інтерфейсу. Ці виводи з'єднані з відповідними висновками мікросхеми ATmega8U2, яка виконує роль перетворювача USB-UART;

- зовнішні переривання: виводи 2 і 3. Чи можуть служити джерелами переривань, що виникають при фронті, спаді або при низькому рівні сигналу на цих висновках. Для отримання додаткової інформації див. Функцію `attachInterrupt ()`;

- ШИМ: виводи 3, 5, 6, 9, 10 і 11. За допомогою функції `analogWrite ()` можуть виводити 8-бітові аналогові значення в вигляді ШИМ-сигналу;

- інтерфейс SPI: виводи 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Із застосуванням бібліотеки SPI дані виводи можуть здійснювати зв'язок по інтерфейсу SPI;

– світлодіод: 13. Вбудований світлодіод, приєднаний до висновку 13. При відправці значення HIGH світлодіод включається, при відправці LOW – вимикається.

В Arduino Uno є 6 аналогових входів (A0 – A5), кожен з яких може уявити аналогову напругу у вигляді 10-бітного числа (1024 різних значення). За замовчуванням, вимір напруги здійснюється щодо діапазону від 0 В до 5 В. Проте, верхню межу цього діапазону можна змінити, використовуючи вивод AREF і функцію `analogReference()`. Крім цього, деякі з аналогових входів мають додаткові функції:

– TWI: вивод A4 або SDA і вивод A5 або SCL. З використанням бібліотеки `Wire` дані виводи можуть здійснювати зв'язок по інтерфейсу TWI;

– AREF. Опорна напруга для аналогових входів. Може бути задіяний функцією `analogReference()`;

– Reset. Формування низького рівня (LOW) на цьому висновку призведе до перезавантаження мікроконтролера. Зазвичай цей вивод служить для функціонування кнопки скидання на платах розширення.

Arduino Uno надає ряд можливостей для здійснення зв'язку з комп'ютером, ще одним Arduino або іншими мікроконтролерами. У ATmega328 є приймач UART, що дозволяє здійснювати послідовну зв'язок за допомогою цифрових висновків 0 (RX) і 1 (TX). Мікроконтролер ATmega16U2 на платі забезпечує зв'язок цього приймача з USB-портом комп'ютера, і при підключенні до ПК дозволяє Arduino визначатися як віртуальний COM-порт. Прошивка мікросхеми 16U2 використовує стандартні драйвера USB-COM, тому установка зовнішніх драйверів не потрібно. На платформі Windows необхідний тільки відповідний .inf-файл. У пакет програмного забезпечення Arduino входить спеціальна програма, що дозволяє зчитувати і відправляти на Arduino прості текстові дані. При передачі даних через мікросхему-перетворювач USB-UART під час USB-з'єднання з комп'ютером, на платі будуть мигати світлодіоди RX і TX.

Бібліотека `SoftwareSerial` дозволяє реалізувати послідовну зв'язок на будь-яких цифрових висновках Arduino Uno.

У мікроконтролері ATmega328 також реалізована підтримка послідовних інтерфейсів I2C (TWI) і SPI. У програмне забезпечення Arduino входить бібліотека `Wire`, що дозволяє спростити роботу з шиною I2C.

Arduino Uno програмується за допомогою програмного забезпечення Arduino. ATmega328 в Arduino Uno випускається з прошитим загрузчиком, що дозволяє завантажувати в мікроконтролер нові програми без необхідності

використання зовнішнього програматора. Взаємодія з ним здійснюється за оригінальним протоколу STK500 (опис, заголовки C).

Проте, мікроконтролер можна прошити і через роз'єм для внутрисхемного програмування ICSP (In-Circuit Serial Programming), не звертаючи уваги на завантажувач; більш докладно про це див. відповідні інструкції.

Вихідний код прошивки мікроконтролера ATmega16U2 знаходиться у вільному доступі. Прошивка ATmega16U2 включає в себе DFU-завантажувач (Device Firmware Update), що дозволяє оновлювати прошивку мікроконтролера.

Щоб кожен раз перед завантаженням програми не було потрібно натискати кнопку скидання, Arduino Uno спроектований таким чином, що дозволяє здійснювати його скидання програмно з підключеного комп'ютера. Один з висновків ATmega16U2, який бере участь в управлінні потоком даних (DTR), з'єднаний з висновком RESET мікроконтролера ATmega328 через конденсатор номіналом 100 нФ. Коли на лінії DTR з'являється нуль, вивод RESET також переходить в низький рівень на час, достатній для перезавантаження мікроконтролера. Дана особливість використовується для того, щоб можна було прошивати мікроконтролер всього одним натисненням кнопки в середовищі програмування Ардуіно. Така архітектура дозволяє зменшити таймаут завантажувача, оскільки процес прошивки завжди синхронізований зі спадом сигналу на лінії DTR.

В Arduino Uno є відновлювані запобіжники, що захищають USB-порт комп'ютера від коротких замикань і перевантажень. Незважаючи на те, що більшість комп'ютерів мають власний захист, такі запобіжники забезпечують додатковий рівень захисту. Якщо від USB-порту споживається струм більше 500 мА, запобіжник автоматично розірве з'єднання до усунення причин короткого замикання або перевантаження.

Максимальна довжина і ширина друкованої плати Uno становить 6,9 см і 5,4 см відповідно, з урахуванням роз'єму USB і роз'єму живлення, які виступають за межі плати. Чотири кріпильних отвори дозволяють прикріплювати плату до поверхні або корпусу. Зверніть увагу, що відстань між цифровими висновками 7 і 8 не кратне традиційним 2,54 мм і становить 4 мм [9].

У якості серводвигунів було обрано серводвигуни MG90S (рис. 4.21).



Рисунок 4.21– Серводвигун MG90S

Серводвигун MG90S з металевим редуктором широко використовується в автомоделях для управління поворотом передніх коліс, в моделюванні для повороту керма і закрилків моделі. Серводвигун MG90S застосовується для повороту деталей різних механізмів. Завдяки редуктора з металевими шестернями на вихідному валу розвивається зусилля достатню для застосування в рухомих роботів. Низька швидкість повороту вала дозволяє більш точно отримати необхідний кут повороту.

Усередині корпусу знаходиться невеликий модуль управління, який під дією вхідного сигналу подає живлення відповідної полярності на електродвигун. Вхідний сигнал управління містить дані про необхідному положенні вала. Для визначення поточного положення вала редуктор з'єднаний з датчиком положення. Електроніка MG90S обчислює різницю між поточним становищем редуктора і необхідним. Модуль управління орієнтуючись на датчик положення подає живлення необхідної полярності на двигун для повороту редуктора приводить у відповідність положення передане сигналом управління і поточний.

Інформація про необхідне положення вала міститься в скважності імпульсів керуючого сигналу. Частота сигналу повинна бути постійна і складати 50 Гц. Шпаруватість – відношення тривалості імпульсу до періоду. Найчастіше

при аналізі параметрів керуючого сигналу розглядають тривалість імпульсу. Для формування такого сигналу зручно використовувати мікроконтролер має функцію широтно-імпульсної модуляції вихідного сигналу.

У якості плати розширення було обрано Arduino Uno R3 Sensor Shield V5.0 (рис. 4.22).



Рисунок 4.22 – Палата розширення Arduino Uno R3 Sensor Shield V5.0

Sensor Shield для підключення Ардуіно Uno R3 – плата розширення V5.0 версії для підключення і монтажу великої кількості сенсорів до контролера і проведення монтажу, пайки з використанням перемичок і без них. Ще називається як Сенсор Шилд для підключення датчиків або плата розширення. Дуже зручно для безлічі сервоприводів. Сенсор Шилд монтується прямо на контролер Ардуіно і не потребує додаткових комутацій і пайку, що дуже зручно для експериментів. На платі Arduino Sensor Shield V5.0 розташовано безліч інтерфейсів для підключення різних зовнішніх пристроїв і живлення.

Контактні інтерфейси на Sensor Shield Arduino V5.0:

- посадка на Uno R3;
- клеми живлення Arduino мають стандартні параметри і позначення позначення висновків: 3V3, 5V, GND, VIN, RESET;
- розширення для 6 аналогових входів позначених 0 – 5;
- порти аналогових входів-виходів Analog IO ports – 6 шт, позначення висновків: A0 – A5. Комутація проходить через контакти GND, VCC, Signal;
- колодки інтерфейсу RS232 (послідовна шина COM);
- I2C (ІС) port (Inter-Integrated Circuit) – послідовна шина розширення даних для зв'язку інтегральних схем. Позначення контактів: SCL (Serial CLock), SDA (Serial DAta);

– Digital IO ports (колотка цифрових входів-виходів) має 16 виводів, позначення висновків: 0 (Rx), 1 (Tx) – 13, GND, AREF. Кожен вхід-вихід має 3 контакту GND, VCC, Signal. Перший контакт позначений на схемі;

– аналогічно транслюються на плату розширення Входи-виходи Arduino контролера 0 – 13, GND, AREF. Вивод AREF (опорна напруга для аналогових входів);

– бездротової інтерфейс RF RF-інтерфейса APC220;

– паралельний інтерфейс 12864 LCD.

Параметри і електричні характеристики Sensor Shield Arduino:

– напруга живлення плати розширення: 5 В;

– живлення: від Arduino і зовнішніх джерел живлення (бажано зовнішнє і для ШІлд і для контролера);

– I2C (ІІС) port (Inter-Integrated Circuit) – послідовна шина даних для зв'язку інтегральних схем;

– інтерфейс RS232 (послідовна шина COM);

– інтерфейс Analog Buckled ports для зручного і надійного з'єднання з 6 аналоговими входами;

– вхідна напруга (рекомендований): від 7 В до 12 В;

– вхідна напруга (граничне): від 6 В до 20 В;

– габарити плати: 57 мм х 68 мм х 19 мм [11].

У якості Bluetooth модуля було обрано модуль HC-05 (рис. 4.23).

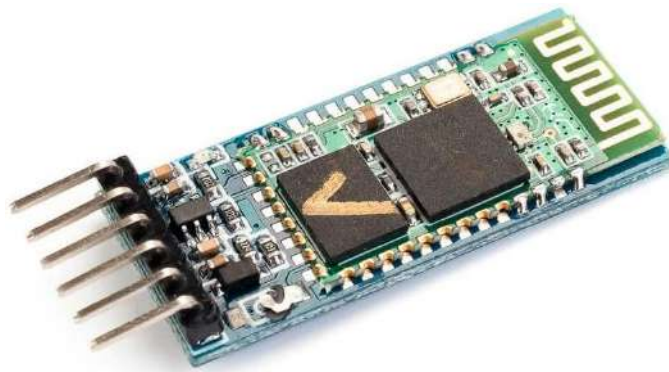


Рисунок 4.23– Bluetooth модуль HC-05

На рис. 4.24 представлена схема підключення Bluetooth модуля до плати розширення.

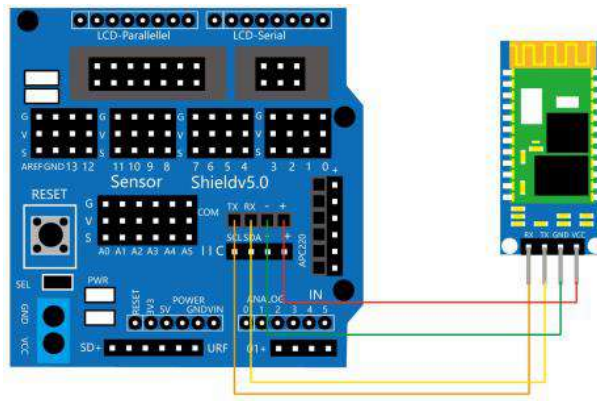


Рисунок 4.24 – Схема підключення Bluetooth модуля до плати розширення

На рис. 4.25 зображено схему підключення серводвигунів.

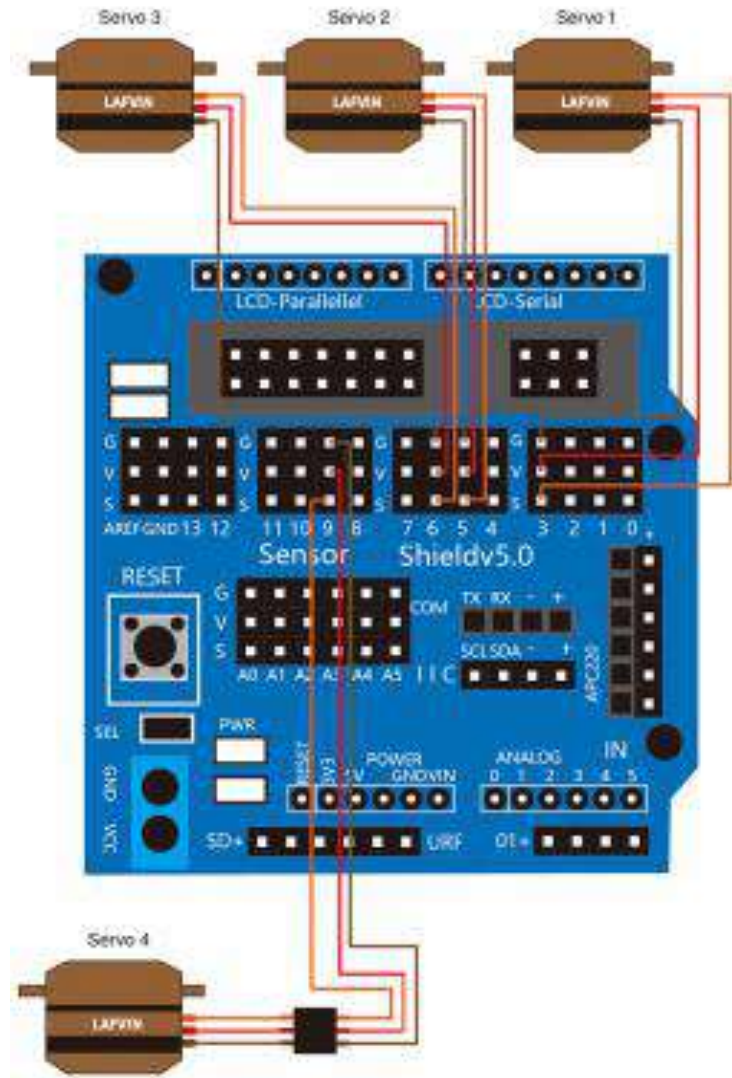


Рисунок 4.25 – Схема підключення серводвигунів

4.7 Аналіз та принцип роботи системи дозування для відмивки плат

Запропонована система дозування рідини для відмивки друкованих плат призначена для управління процесами дозування і обліку витрат компонентів. Вона дозволяє дозувати речовини в заданій тимчасовій або логічній послідовності.

Принцип роботи системи представлено нижче:

- подача живлення для запуску системи;
- появлення бігучого напису SERVICE на дисплеї (при затриманні кнопки);
- запуск помпи і таймера через кнопку енкодера;
- вибір режиму (автоматичний і ручний);
- кладемо плати в ванни для відмивки;
- виконується заповнення ванни рідиною для відмивки згідно з обраним режимом.

Для розробки проекту було обрано середовище Arduino. Інтерфейс містить такі основні елементи:

- текстовий редактор для написання коду;
- область для виведення повідомлень;
- текстова консоль;
- панель інструментів з традиційними кнопками;
- головне меню.

Даний софт дозволяє комп'ютеру взаємодіяти з Ардуіно як для передачі даних, так і для прошивки коду в контролер. Для роботи плати і майбутнього проекту необхідно написати і завантажити на Arduino скетч.

Вибір середовища Arduino був зроблений виходячи з переваг роботи в ньому, а саме:

- просте і зрозуміле середовище програмування;
- низька вартість. Ці плати відносно дешеві;
- плата має вбудований програматор (не потрібен компілятор);
- присутня велика кількість бібліотек для модулів, сенсорів, дисплеїв тощо, доступних для вільного використання;
- процеси живлення, програмування і обмінювання повідомленнями з Arduino можна здійснювати за допомогою одного USB кабелю.

Пристрої на базі Arduino можуть отримувати інформацію про навколишнє середовище за допомогою різних датчиків, а також можуть управляти різними

виконавчими пристроями.

Для розробки макету була обрана версія плати Arduino Nano. Незважаючи на свій скромний розмір, вона практично нічим не поступається Arduino Uno по функціоналу і може використовуватися в проектах, де габарити відіграють важливу роль. В цій платі передбачені штирові колодки. Це зручно для макетування, але при бажанні їх можна не встановлювати. В такому випадку дроти до потрібних висновків припаюються безпосередньо.

Також штирові колодки потрібні при використанні в проекті спеціалізованих плат розширення, яких для даної модифікації Arduino придумано безліч (для мого проекту це було непотрібно, але це також перевага даної плати).

Для заощадження місця, радіоелементи розташовані по обидва боки плати. З лицьового боку нанесена вся інформативна трафаретна печать, встановлений МК ATmega328, кварцовий резонатор, роз'єм Mini-USB, кнопка скидання і чотири індикаторних світлодіода (TX, RX, PWR і L). Перші два світлодіода загоряються при обміні даними плати Arduino Nano з іншими пристроями через послідовний серійний порт. Індикатор PWR відображає надходження на плату живлення, а світлодіод L є індикатором загального призначення і спалахує в разі подачі високого сигналу на висновок (рис. 4.26).

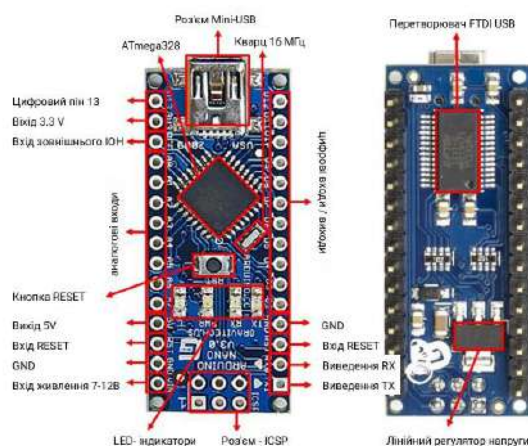


Рисунок 4.26– Складові плати Arduino Nano

Незважаючи на дуже скромний розмір (4.3 см x 1.85 см), Arduino Nano є повнофункціональним закінченим пристроєм, який в більшості випадків базується на мікроконтролері ATmega328. Плата дозволяє полегшити процес макетування та враховуючи її популярність, успішно справляється з цим завданням протягом довгого часу. Універсальність Arduino Nano в комплексі з

безліччю бібліотек і різноманітністю технічної літератури є досить потужним інструментом.

Для проектування та збірки макету була обрана плата Arduino Nano. МК на платі програмується за допомогою мови Arduino .

У процесор Arduino було завантажено програму, яка допомагає дозувати рідину та автоматично розливати вже розраховану чітку кількість.

На платі існує пара десятків контактів, до яких були підключені необхідні компоненти для роботи програми, а саме:

- адресна стрічка;
- модуль поворотного енкодера WAVGAT;
- кнопка;
- дисплей;
- кінцеві вимикачі;
- драйвер двигуна постійного струму L298N;
- мікросервопривод WAVGAT MG90S;
- роз'єм адаптера Micro USB;
- водяний насос 12 В з двигуном постійного струму;
- дроти для з'єднання.

Адресні світлодіодні стрічки схожі зі звичайними led смугами. Вони мають робочу напругу 5 В, 12 В, в деяких моделях 220 В. Стрічка оснащена SMD світлодіодами RGB з контролерами на кожному. Таке поєднання діода з керуючим елементом і називають адресним діодом, тобто, задана програма може керувати кожним діодом окремо. Якщо розбирати конструкцію адресного світлодіода SMD 5050 RGB, то це конструкція складається з керуючої мікросхеми, трьох кристалів світіння (червоний, зелений, синій), і трьох транзисторних висновків. І це все всередині кожного діода. Адресна стрічка представлена рис. 4.27.



Рисунок 4.27 – Адресна стрічка

Дана стрічка має подвійну лінію передачі даних. Якщо раптово один світлодіод припинив працювати по несправності, інші діоди не перестають функціонувати. Функціонує стрічка від безпечної напруги 5 Вольт. Для її служби буде потрібна напруга 5 В.

Таблиця 4.1 – Характеристики адресної стрічки

Тип діода	SMD 5050
Потужність	14,4 W
Кількість світлодіодів	30 Led / m
Робоча напруга	5 V
ІР захист	IP20
Кратність різання	1 Led
Ширина	10 мм
Висота	2 мм
Робоча температура	від -40 до до +60 градусів за Цельсієм
Країна виробник	Китай

Модуль датчика обертання KY-040 являє собою плату з простим механічним енкодером – поворотним датчиком EC11, який відстежує ступінь повороту осі і в якому напрямку вона обертається. Під віссю енкодера розташована кнопка. Конструкція модуля дозволяє легко додати його до будь-якої макетної платі. Кут повороту енкодера не обмежений, тому застосування енкодера часто дає більш зручний і зрозумілий інтерфейс управління. Може

використовуватися в якості цифрового потенціометра. Енкодер зображен на рис. 4.28.



Рисунок 4.28 – Модуль поворотного енкодера

Енкодер перетворює кут повороту в електричний сигнал. При обертанні ручки модуля ми отримуємо два сигналу (А і В), які протилежні по фазі. Сигнали А і В залежать один від одного при обертанні енкодера Ардуіно за годинниковою або проти годинникової стрілки. Для зчитування сигналу А та В з енкодера, можна використовувати, як цифрові, так і аналогові порти мікроконтролера.

Таблиця 4.2 – Характеристики енкодера KY-40 (EC11)

Довжина деталі, що обертається вала	12 мм
Повна довжина вала	20 мм
Діаметр вала	6 мм
Розмір (Д x Ш x В)	13 x 11 x 25 мм
Closed Circuit опір	максимальне значення – 3 Ом
Максимальне значення току	10 мА при 5 В постійного струму
Діапазон робочих температур	від -30 до +70 градусів за Цельсієм
Діапазон температур зберігання	від -40 до +85 градусів за Цельсієм
Кількість циклів обертання	30000 циклів (мінімум)
Кількість циклів перемикання	20000 циклів (мінімум)

За допомогою даної кнопки здійснюється запуск системи, а так само при утриманні, вона відповідає за перемикання режимів роботи, з автоматичного на ручний. Кнопка представлена на рис. 4.29.



Рисунок 4.29– PBS10C-2 green, Кнопка без фіксації ON- (OFF)

Таблиця 4.3 – Технічні характеристики PBS10C-2

Функціональне призначення	кнопковий вимикач
Фіксація	відсутня
Алгоритм роботи	ON- (OFF);
Робоча напруга	250 В
Робочий струм	1 А

На передній частині модуля МТ1637 встановлений чотирьохзначний 7-сегментний дисплей з двокрапкою в центрі, на зворотному боці встановлена мікросхема ТМ1637 виробництва Titan Micro Electronics. Сам 7-сегментний дисплей (рис. 4.30) складається з семи світлодіодів, позначених буквами: від «а» до «g», які спалахують незалежно від один одного і можуть відобразити числа і букви. Так само, плата сумісна з 3,3 - 5 В, ніяких додаткових резисторів або конденсаторів не потрібно.



Рисунок 4.30– Семисегментний дисплей ТМ1637

Світлодіодний семисегментний індикатор являє собою групу світлодіодів, розташованих і об'єднаних в певному порядку. Світлодіодні контакти промарковані мітками від а до g (і додатково dp – для відображення десяткового дробу), і один загальний висновок, який визначає тип підключення індикатора. Запалюючи одночасно кілька світлодіодів, можна формувати на індикаторі символи цифр.

Щоб відобразити на індикаторі цифру або букву слід засвітити кілька сегментів. Наприклад, для відображення одиниці 1 задіюються сегменти b і c. При відображенні вісімки 8 задіюються всі символи від a до g (рисунок 4.31).

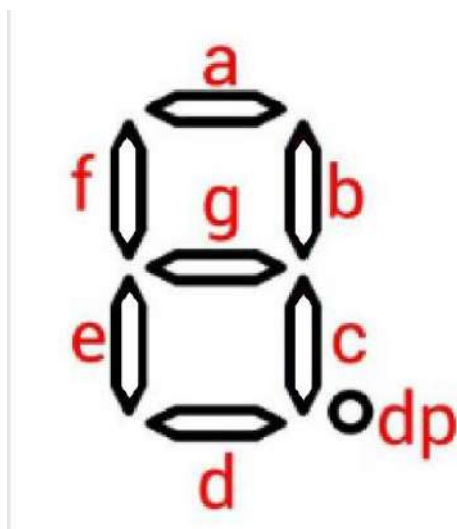


Рисунок 4.31 – Семисегментний індикатор

L298N Motor Controller Module міні – компактна версія драйвера для звичайних і крокових двигунів постійного струму (рис. 4.32). В основі модуля лежить мікросхема L298N виробництва ST. Дозволяє легко керувати швидкістю і напрямком крокової двигуна. Основний чіп модуля це мікросхема L298N, що складається з двох H-міст (H-Bridge), один для виходу А, другий для виходу В.

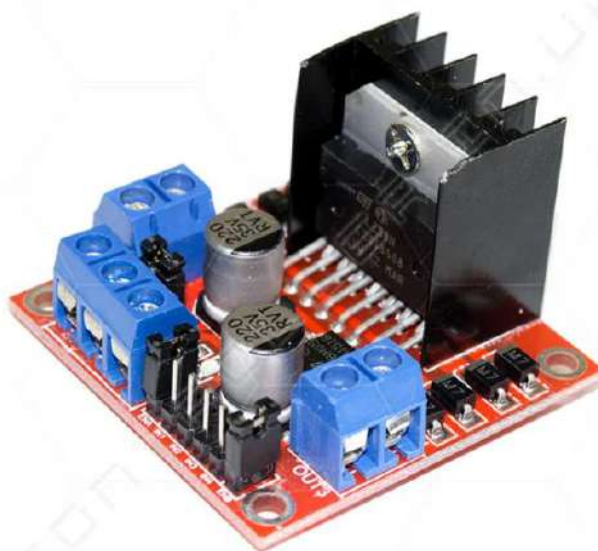


Рисунок 4.32 – Драйвер двигуна постійного струму L298N

H-міст широко використовується в електроніці і служить для зміни обертання двигуном, схема H-моста містить чотири транзистора (ключа) з

двигуном в центрі, утворюючи H-подібну компоновку. Принцип роботи простий, при одночасному закриття двох окремих транзистора змінюється полярність напруги, прикладеного до двигуна. Це дозволяє змінювати напрямок обертання двигуна.

Tower Pro SG90 micro servo 9g – легкий і якісний мікро сервопривід за маленьку ціну. Управління за допомогою Arduino, AVR, PIC, ARM і інших мікроконтролерів. В своєму макеті управління сервопривідом виконувалось завдяки Arduino Nano. Кольори проводів стандартні. Сервопривід коштує недорого, він не забезпечує точних налаштувань початкових і кінцевих позицій. Для того, щоб уникнути зайвих перевантажень і характерного тріску в положенні 0 і 180 градусів краще виставляти крайні точки в 10 ° і 170 °. При роботі пристрою важливо стежити за напругою живлення. При сильному завищенні цього показника можуть пошкодитися механічні елементи зубчастих механізмів.



Рисунок 4.33– Мікро сервопривід SG90

Даний роз'єм адаптера Micro USB має дуже міцну конструкцію, яка дозволяє виконувати велику кількість підключень і відключень, без ризику виходу з ладу як гнізда, так і електричної розетки. Саме тому, до цих пір, він є еталоном для підключення комп'ютерної периферії, такий як: мишки, клавіатури, веб-камери, флешки та інші.



Рисунок 4.34 – Роз'єм адаптера Micro USB

Мініатюрний мембранний насос являє собою насос, який дозволяє тривалий час працювати на холостому ході без наповнення трубки рідиною (рис 4.35). Механічні деталі в корпусі насоса відсутні. Дана деталь призначена для всмоктування бензину, води, масла та інших схожих рідин.

Всмоктування насоса реалізується при обертанні двигуна за допомогою колеса ексцентрикового вала двигуна. Привідна гумова петля за допомогою зворотньо-поступального руху здійснює всмоктування і розрядження всередині формування порожнини. Циркуляція насоса виконується через закритий зворотний клапан і відкритий клапан для досягнення всмоктування і розрядки газу і рідини.

Даний водяний насос складається усередині з пластика та силікону, працює за рахунок перекошу головки і клапанів. Була обрана саме ця водяна помпа, тому що вона маленька, більш тиха, має швидку подачу і сильний напір струменю.

Управління цією помпою та живленням на привід буде відбувається за рахунок драйверу двигуна постійного струму.



Рисунок 4.35 – Водяний насос 12В з двигуном постійного струму

Механічний кінцевик представляє собою простий перемикач (кнопку), який спрацьовує при зіткненні з кареткою. При спрацьовуванні кінцевика на порт мікроконтролера надходить сигнал. За цим сигналом МК вмикає кроковий двигун і відкочує каретку на кілька мм назад. Кінцеві вимикачі натискної дії бувають з роликком або без.

Мікровимикачі характеризуються допустимим струмом і напругою. При перевищенні максимального струму такий вимикач нагріється вище допустимої температури. Але так як буде подаватися невисока напруга в розробтаній системі, даний кінцевик досить добре підійде.

Звичайний кінцевий вимикач, має контакти COM, NO, NC.

У розімкнутому положенні:

- контакти COM і NC знаходяться під напругою + 5V;
- контакт NO – земля (GND);
- контакти COM і NC замкнуті.

У замкнутому положенні:

- контакти COM (+ 5V) і NO (GND) замикаються;
- контакти COM і NC розмикаються.

При спрацьовуванні чути клацання.

Для макету було використано чотири мікровимикача (кінцевика) з роликком (рис. 4.36).



Рисунок 4.36 – Мікровимикач (кінцевик) KW12-3

Була розроблена схема підключення елементів для системи дозування рідини. Дана структурна схема підключення компонентів дозволяє розглянути принцип роботи пристрою в найзагальнішому вигляді (рис. 4.37). На ній зображені основні, функціональні частини, які будуть використані для розробки макету, а саме:

- діоди (4 штуки);
- кінцеві вимикачі (4 штуки);

- модуль поворотного енкодера;
- кнопка;
- дисплей;
- драйвер приводу двигуна постійного струму L298N;
- мікро сервопривід WAVGAT MG90S;
- роз'єм адаптера Micro USB;
- водяний насос 12В з двигуном постійного струму.

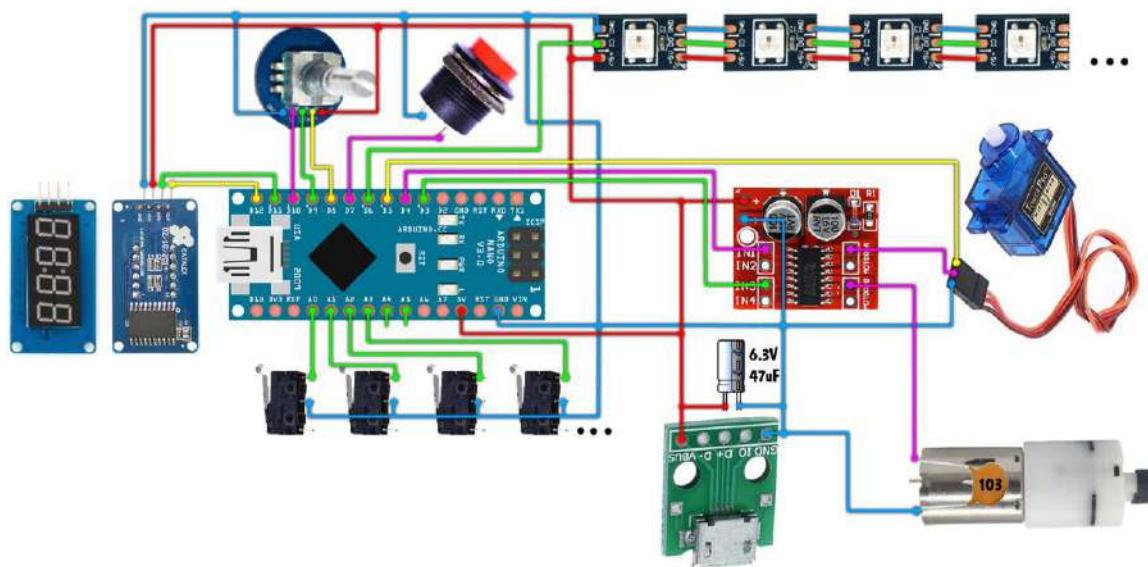


Рисунок 4.37 – Схема підключення компонентів системи дозування рідини

На схемі зображені усі зв'язки між даними компонентами. Такий етап проектування схеми дозволяє на ранньому етапі виявити помилки проектування, перерозподіляти вимоги до вузлів радіоелектронного пристрою. В результаті значно скорочуються зусилля при розробці принципової схеми виробу.

Реальне розташування блоків (конструкція) на структурній схемі зазвичай не враховується. Крім того, не уточнюється спосіб зв'язку між блоками (дротова, індуктивна, оптична тощо). Дана структурна схема дає наочне уявлення про послідовність взаємодії функціональних частин у виробі.

На схемі підключення, живлення сервопривіду SG90 було розв'язане через драйвер двигуна постійного струму. Зроблено це для того, щоб можна було керувати живленням і відключати його. Таким чином, можна знижувати енергопостачання та припиняти шуми від праці. Дана система може підтримувати роботу від однієї до 6 ємностей. Для цього потрібно припаяти необхідну кількість світлодіодів і кнопок, скільки необхідно ємностей. Усі інші настройки будуть в програмі.

4.8 Аналіз методів інтелектуального керування складними системами

Система керування на промисловості (СКП) – це особливий термін, який використовується для визначення комбінації програмного та апаратного забезпечення з мережею для балансування критично важливої промислової інфраструктури (рис. 4.39).

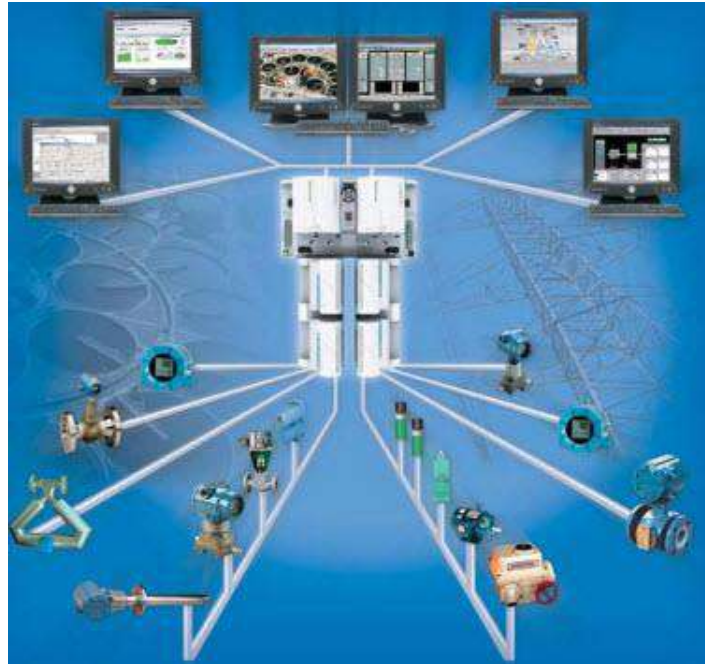


Рисунок 4.39– Система керування на промисловості

Він використовує різні технології, такі як програмовані логічні контролери (ПЛК), диспетчерське управління та збирання даних (SCADA), системи промислової автоматизації та управління (СПАУ), віддалені термінали (ВТ), інтелектуальні електронні пристрої (ІЕП), керуючі сервери та датчики.

Таким чином, інформаційна система, що використовується для керування різними промисловими процесами, такими як обробка продукції, виробництво та розподіл. Ці механізми включають системи збору даних і диспетчерський контроль для управління географічно розосередженими активами.

Спочатку для керування технологічним оснащенням використовувалися так звані локальні панелі. Ця система, однак, потребувала персоналу. На наступному етапі розвитку було введено центральні диспетчерські передачі всіх заводських вимірів. Цей процес централізації виявився дещо успішним. Головним недоліком є недостатня гнучкість, оскільки кожен контур управління має апаратне забезпечення контролера, що додатково вимагає реконфігурації сигналу шляхом повторної проводки або повторного розведення. Також система потребує постійного руху оператора, що може створити труднощі.

Електронні процесори були використані в якості заміни цих дискретних контролерів комп'ютерними алгоритмами. Вони розміщені на стійках введення-виводу разом із процесорами керування. Таким чином, це свого роду розподілена система, що була реалізована на той час. Ця політика була прийнятною, оскільки вона ввела автоматичну реєстрацію подій, уможливила складну обробку аварійних сигналів і усунула необхідність фізичного запису, такої як самописці.

Було розроблено різні форми стійок керування, яким багато виробників привласнили комерційну назву розподіленої системи керування (РСК) для керування великими системами. Ці системи також були замінені на ПЛК, щоб використовувати керуючі дисплеї та мережі сигналів. Вперше вони були прийняті у автомобільній промисловості, де використання послідовної логіки було складним. Потім були представлені в інших програмах, керованих подіями.

Також можна відстежувати системи SCADA за допомогою різних розподільних додатків, таких як трубопроводи природного газу, електроенергії та води, де необхідно збирати видалені дані по каналах з високою затримкою. Зазвичай вони використовують ВТ (віддалені термінали) для передачі даних у центри управління.

Отже, SCADA, ПЛК чи РСК ефективно розвивалися в системах керування з відкритим чи замкнутим контуром. Збільшення швидкості процесорів призвело до подальшого підвищення ефективності процесу. У 1993 році галузь перейшла до вдосконалення програмного забезпечення з посиленням стандартизації коду. Впровадження ООП (об'єктно-орієнтованого програмування) уможливило розробку як промислових ПК, так і програмованих контролерів автоматизації (ПКА). Крім того, це призвело до багатьох ефективних поліпшень, а також економії коштів у порівнянні з традиційними ПЛК.

Нові технології та платформи зробили значний внесок у розвиток систем SCADA та РСК, ще більше стираючи прогалини у визначеннях.

СКП є важливими. Основними цілями використання СКП є:

- полегшення зв'язку через різні протоколи;
- автоматизація енергосистем;
- автоматизація будівель;
- автоматизація процесів тощо;
- використання протоколів забезпечує сумісність між виробниками.

Переваги систем управління технологічними процесами є:

- підвищення продуктивності, безпеки та надійності;
- повітряний проміжок сприяє зменшенню проблем з безпекою;

- підтримка кібер-фізичних систем (КФС) як четверту промислову революцію, коли все підключається до Інтернету;
- ефективне електронне керування завданнями;
- зниження втрати енергії з покращенням екологічних характеристик.

Також присутні і обмеження промислових систем управління. Існує декілька проблем, з якими вам доведеться зіткнутися під час запровадження систем керування на промисловості на суттєвому рівні:

- багато факторів, таких як невизначеність процесу, перешкоди, динаміка процесу та насичення приводу, вимагають розгляду;
- результати можуть бути ускладнені за допомогою обчислювальних інструментів для проектування системи управління;
- помилки, що виникають, можуть бути небезпечними або пошкодити всю систему.

Багато галузей промисловості використовували для різних видів електронних програм СКП. Практично вся критична інфраструктура та галузі промислового виробництва, водопідготовки, транспорту та керування електроенергією вимагають того чи іншого типу АСУ ТП разом із пов'язаними з ним процедурами та пристроями.

СКП, організовані в кілька класів залежно від відносної складності їхніх дій, що управляють, по відношенню до загальних функцій. Загальні типи цих систем управління включають в себе:

- програмовані логічні контролери (ПЛК);
- інтелектуальні електронні пристрої (ІЕП);
- диспетчерське керування та збір даних (SCADA);
- програмовані контролери автоматизації (ПКА);
- людино-машинний інтерфейс (ЛМІ);
- промислова автоматизація та системи керування (ПАСК);
- видалені термінальні пристрої (ВТП);
- розподілені системи керування.

ПЛК – це напівпровідникові системи управління з міцною та надійною конструкцією та набором спеціальних функцій. Деякі з цих функцій включають інтуїтивно зрозумілий інтерфейс програмування, керування введенням-виведенням, трирежимне (ПД) управління, арифметичні, рахункові та тимчасові механізми, а також послідовне управління.

ПЛК виготовлені таким чином, щоб бути дуже стійкими, здатними витримувати екстремальні погодні умови, такі як дуже високі та низькі температури, висока вологість, електричні перешкоди та сильні вібрації. Ці

контролери призначені для моніторингу та керування великою кількістю приводів та датчиків, і вони виділяються в порівнянні з типовими комп'ютерними та процесорними системами завдяки великій кількості налаштувань введення-виведення.



Рисунок 4.40 – Зовнішній вигляд ПЛК

ПЛК використовують у:

- системах лиття під тиском для пластмас;
- системах керування машинами;
- системах контролю безпеки у туристичній індустрії;
- роботі ескалаторів та ліфтів;
- кульових млинах, шахтних та вугільних пічках при виробництві цементу;
- при контролі обробки листового скла та співвідношення матеріалу скла у скляній промисловості;
- засобах керування виробництвом газет чи книжкових сторінок у процесі рулонного офсетного друку;
- пральних машинах;
- керуванні світлофором та його роботою тощо.

Інтелектуальні електронні пристрої (ІЕП) – це електронні пристрої, призначені для багатьох цілей у галузі зв'язку, вимірювання, контролю та управління потужністю. ІЕП вбудовані в структури СКП, щоб забезпечити найскладніші можливості автоматизації енергопостачання (рис. 4.41).

Зазвичай це електронні компоненти з мікропроцесорами, такі як контролер напруги та регулятор. Ці електронні пристрої можуть обмінюватися даними за декількома різними протоколами, наприклад, через Ethernet в реальному часі або шині цифрового зв'язку. ІЕП можна знайти у багатьох процесах та промислових системах управління, таких як РСК та SCADA.



Рисунок 4.41– Зовнішній вигляд інтелектуального електронного пристрою

Існує безліч програм для інтелектуальних електронних пристроїв, таких як:

- металургійні заводи;
- інструменти екологічного контролю;
- фармацевтичне виробництво;
- їжа та харчова промисловість;
- очисні споруди;
- контроль якості;
- автомобільне виробництво;
- шахтарська справа;
- водоочисні споруди;
- паперова фабрика;
- агрохімікати та добрива;
- цукропереробні заводи;
- нафтохімія та нафтопереробка;
- керування водними ресурсами тощо.

Диспетчерське управління та збір даних – це комп'ютерна система, яка обробляє та збирає дані та здійснює оперативний контроль на значних відстанях [7]. Системи SCADA (рис. 4.42) були розроблені для вирішення проблем зв'язку, зокрема проблем із цілісністю даних та затримок, пов'язаних із різними засобами зв'язку.

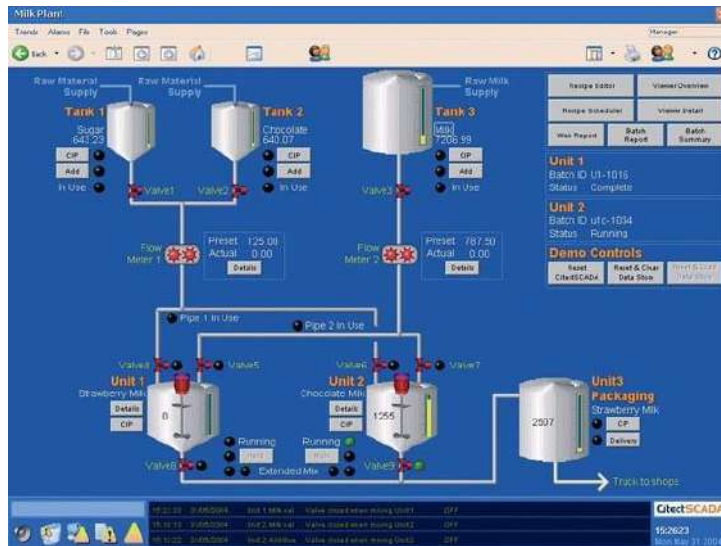


Рисунок 4.42– Приклад SCADA-системи

Системи диспетчерського управління та збору даних використовуються в багатьох додатках, деякі з яких:

- трубопровідні системи;
- передача та розподіл енергії
- мікрохвилі;
- супутники;
- телефонні лінії тощо.

Програмовані контролери автоматизації (ПКА) дозволяють системам надавати складні інструкції вищого рівня різного автоматизованого устаткування. Ці типи контролерів добре підходять для різних промислових та інфраструктурних секторів та додатків. ПКА часто мають більше можливостей підключення та більш широкі можливості керування системами, ніж більшість систем ПЛК. ПКА складаються із п'яти основних елементів: одна багатопрофільна платформа розробки; сумісність із корпоративною мережею; адаптовані програмні засоби, що покращують технологічний процес; багатодоменна функціональність; та модульна, відкрита архітектура (рис. 4.43).

ПКА використовують у:

- системах технічного зору (обробка сигналів, реєстрація даних);
- контролі процесу;
- системах руху;
- логічних системах.



Рисунок 4.43 – Приклад програмованого контролеру автоматизації

Людино-машинні інтерфейси (ЛМІ) – це апаратні та програмні засоби зв'язку та обміну інформацією між людьми-операторами та машинами та/або комп'ютеризованими системами. За допомогою ЛМІ можна контролювати, візуалізувати та керувати різними процесами пристрою за допомогою взаємодії між людьми та машинами.

Існує безліч застосувань ЛМІ, у тому числі:

- портативні портативні пристрої;
- керування машинами та процесами для заводських цехів;
- різні типи машин;
- цифрові вивіски;
- медичні прилади;
- централізовані диспетчерські;
- будівельна та промислова автоматизація;
- автомобільна техніка тощо.



Рисунок 4.44– Людино-машинні інтерфейси

Промислова автоматизація та системи керування (ПАСК) є набір напівпровідникових, електромеханічних та механічних пристроїв, які виконують кілька процесів управління, моніторингу та приведення в дію багатьох логічних пристроїв та складних систем технологічного типу. ПАСК може відслідковувати

та контролювати багато процесів за допомогою датчиків на машинах, інтелектуальних пристроїв та іншого програмного та апаратного забезпечення, яке перетворює інформацію датчиків у різні вихідні дані управління.



Рисунок 4.45 – Зовнішній вигляд ПАСК

Програми ПАСК включають:

- засоби управління двигуном та приладдя (гальма змінного та постійного струму, контролери двигунів, приводи з регульованою швидкістю та частотні перетворювачі);
- автотрансформатори, реактори, перетворювачі;
- системи управління технологічними процесами;
- комплексне виробництво;
- різні пристрої людино-машинного інтерфейсу;
- системи керування машинною логікою;
- послуги системної інтеграції;
- автомобільний транспорт;
- керування об'єктами.

Віддалений термінал (ВТП) – це промислова система управління, заснована на мікропроцесорі [13]. ВТП в електронному вигляді підключають різні види обладнання до інших систем управління, таких як SCADA або РСК. Ці електронні блоки несуть дані датчиків через вхідні та вихідні потоки, які передаються в централізовану СКП через контур керування. ВТП також керують з'єднаннями з віддаленими або місцевими органами управління, тому вони відомі як віддалені блоки телекерування або віддалені блоки телеметрії (рис. 4.46).



Рисунок 4.46 – Зовнішній вигляд віддаленого терміналу

Основне використання ВТП:

- морські платформи та нафтові свердловини;
- трубопровідні насосні станції (водопостачання, водовідведення);
- зовнішні попереджувальні сирени;
- мережі природного газу та супутнє обладнання;
- устаткування повітряного контролю;
- системи моніторингу довкілля (викиди, забруднення, якість повітря)

тощо.

У розподіленій системі керування (РСК) існують різні системи, розподілені таким чином, що ними можна керувати окремо. Розподілені системи керування – це системи контролерів, датчиків та спеціалізованих комп'ютерів, розподілені на промислові підприємства. Кожен компонент РСК має унікальну функцію, таку як графічний дисплей, управління процесом, збирання та зберігання даних. РСК служить центральним мозком промислового підприємства, у своїй елементи системи взаємодіють друг з одним через централізовану керуючу комп'ютерну мережу, свого роду локальну мережу. РСК може відслідковувати виробничі процеси та приймати автоматизовані рішення в режимі реального часу, використовуючи виробничі матриці та тенденції, які система відзначає на всьому підприємстві.

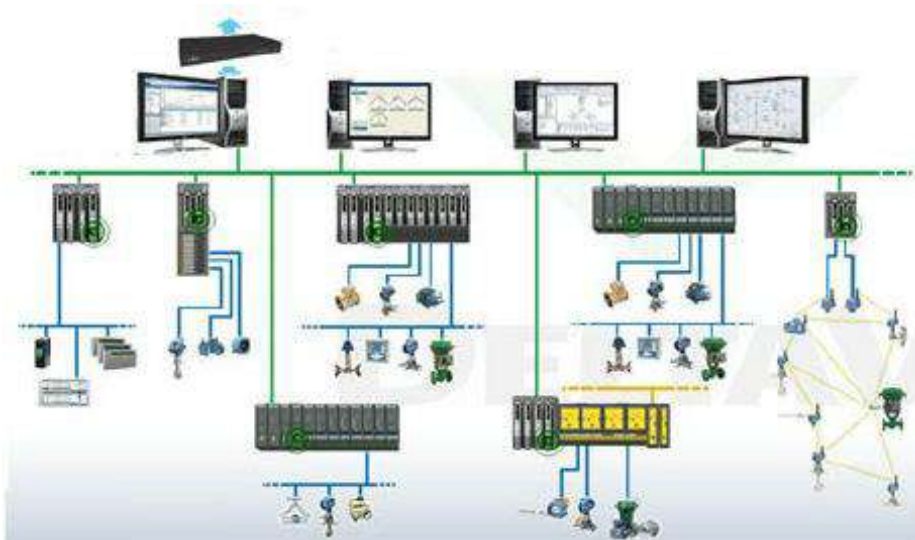


Рисунок 4.47– Розподілена система керування

РСК може використовуватися в пакетному та безперервному режимах і має таке використання:

- хімічні заводи;
- очисні споруди;
- водоочисні споруди;
- атомні електростанції;
- сільськогосподарські програми;
- системи керування котлами;
- целюлозно-паперові комбінати;
- системи контролю якості;
- виробництво автомобілів;
- системи екологічного контролю.

4.9 Аналіз використання комп'ютерного зору на сучасному виробництві

Індустрія 4.0 – це створення розумної фабрики, виробничого майданчика, яка використовує дані від різних типів датчиків та всіх доступних джерел для оптимізації процесів. Комп'ютерний зір (КЗ) є частиною комплексного підходу до інтелектуального виробництва, який дозволяє комп'ютерам та машинам бачити фізичний світ, дозволяючи їм вилучати, обробляти та аналізувати інформацію з візуальних вхідних даних.

Ринок комп'ютерного зору зазнає невтомних перетворень, постійно створюючи нові рішення та технологічні досягнення. Аналіз Forrester показує, що ми знаходимося на вершині стадії комерціалізації розробки КЗ і

використовуємо технологію для розпізнавання осіб, аналітики контенту та механізмів рекомендацій з аналітики та багато іншого.

Основними перевагами використання комп'ютерного зору на виробництві є:

- ефективність часу – повністю автоматизована система не тільки працює набагато швидше, але й за необхідності може працювати цілодобово та без вихідних;

- точність – впровадження рішень на основі КЗ дозволяє компаніям-виробникам досягти значно вищого рівня точності в межах прийнятого допуску, а у поєднанні зі спеціальним обладнанням та передовими алгоритмами, дозволяє досягти майже ідеального рівня точності у виробництві та контролі якості;

- повторюваність – коли справа доходить до роботи, що повторюється, рішення на основі КЗ більш ефективні, коли йдеться про монотонні завдання, зокрема, повністю автоматизована система прискорює час виробництва та знижує вартість виробництва на багатьох рівнях (наприклад, немає необхідності навчати чи переучувати персонал);

- зниження витрат – крім зниження витрат на робочу силу (оскільки для контролю процесу потрібно менше працівників), менше місця для помилок або відхилень від стандартів, тому загальна якість продукту краща, а відходів менше.

- постпандемічне значення – соціальне дистанціювання залишиться, тому скорочення кількості працівників на виробничих лініях допоможе роботодавцям дотримуватися санітарних норм і допомогти співробітникам зберегти здоров'я.

Ключові приклади того, як КЗ і машинне навчання (МН) можуть поліпшити процеси, підвищити продуктивність і збільшити дохід і можуть бути використані у:

- роботи, керовані зором (vision-guided robots) – найчастіше системи КЗ використовуються для позиціонування інструментів та/або деталей на виробничих лініях, а також для їх переміщення та розміщення, тобто система визначає точне розташування об'єкта або інструменту і відправляє ці координати роботу;

- виявлення аномалій використовується для аналізу нових зображень та порівняння їх з вже існуючим набором даних, щоб знайти аномалії та запобігти небезпечним ситуаціям на виробничих майданчиках, виробничих лініях тощо;

- зменшення дефекту – коли на виробничій лінії необхідно перевірити велику кількість виробів, комп'ютерний зір може бути чудовою технологією, яка допоможе виробникам автоматизувати цей процес, тобто коли кількість дефектних виробів досягає певної допустимої кількості, система може

повідомити про це менеджера/інспектора або навіть зупинити виробництво до проведення додаткової перевірки;

- перевірка упаковки – більшість виробників залежить від певної якості своєї продукції, тому дуже важливо мати систему, яка може автоматично виявляти будь-які відхилення від стандарту;

- сканування штрих-кодів та текстових етикеток – впровадження комп'ютерного зору на виробничому майданчику може: а) покращити процес управління деталями; б) прискорити обробку замовлень та в) покращити систему відстеження;

- маркування, відстеження – допомагає компаніям-виробникам ідентифікувати предмети (а також зміщені або зім'яті етикетки), зіставляти їх з базою даних та відстежувати їх.

Як і будь-яке технологічне рішення в наші дні, комп'ютерний зір у виробництві може мати досить універсальний технічний стек. У цьому розділі ми розглянемо деякі з інструментів і бібліотек, що підтримують КЗ:

- OpenCV – один з найбільш широко відомих, використовуваних та улюблених інструментів комп'ютерного зору. Він написаний на C++ і має багато прив'язок до мов: Python, Java, Matlab, JavaScript. Він має бібліотеку функцій та алгоритмів для комп'ютерного зору, обробки зображень та чисельних алгоритмів загального призначення з відкритим вихідним кодом, які дозволяють користувачам інтерпретувати зображення, калібрувати камеру за заданим стандартом, зменшувати або навіть усувати оптичні спотворення та аналізувати рух об'єкту. Він також поставляється з низкою інших зручних функцій, таких як 3D-реконструкція, сегментація об'єктів, розпізнавання жестів тощо;

- VisionWorks – ще один відмінний інструмент від Nvidia, який допомагає користувачам створювати конвеєри КЗ, забезпечуючи роботу багатьох програм, таких як інтелектуальна відеоаналітика, алгоритми локалізації та багато іншого. Nvidia VisionWorks також надає потокобезпечний API, що спрощує і прискорює відстеження та аналіз кількох сцен;

- фреймворки AForge.NET/Accord.NET та Computer Vision Sandbox – гарний набір інструментів для рішень .NET. Обидва поставляються з широким спектром доступних інструментів, надаючи користувачам доступ до КЗ, бібліотек обробки зображень та відео, розширених алгоритмів аналізу та багато іншого. КЗ Sandbox, програмний пакет з відкритим вихідним кодом, дозволяє вирішувати різні завдання, пов'язані з КЗ (наприклад, автоматизація на основі машинного зору, обробка зображень/відео, перегляд з декількох камер, сканування штрих-кодів тощо);

– SimpleCV – це фреймворк для створення КЗ-програм. Завдяки швидкому створенню прототипів та сумісності з багатьма іншими інструментами SimpleCV є чудовим варіантом для технології комп'ютерного зору у виробництві. Однак його продуктивність може бути не такою високою, як OpenCV, особливо в середовищі з високим навантаженням. Іншою проблемою є досить невелика спільнота, якій може бракувати підтримки з погляду документації та вирішення проблем;

– бібліотека Vision Workbench – версія КЗ-інструментів NASA, написана в основному на C++. І хоча це ще не найсучасніший елемент, ця бібліотека призначена для використання як база для прикладних досліджень у галузі комп'ютерного зору. Крім аналізу зображень та покращення, його також можна використовувати для дослідницьких цілей та робототехніки;

– PyTorch — це бібліотека машинного навчання з відкритим кодом, яка часто використовується для комп'ютерного зору та обробки природної мови. Він має інтерфейси Python та C++.

Комп'ютерний зір є міждисциплінарною областю. Незважаючи на те, що ці інструменти та платформи мають більш широку сферу застосування, вони також можуть стати потужним інструментом для рішень Computer Vision в Індустрії 4.0:

– CUDA – це платформа для паралельних обчислень, яка допомагає користувачам запускати складні процеси на графічних процесорах, а чи не на центральних процесорах. Його набір інструментів включає бібліотеку NVIDIA Performance Primitives, яка є набором різних функцій обробки зображень, відео та сигналів. CUDA добре працює з різними бібліотеками та інструментами, що робить його ідеальним інструментом для прискорення алгоритмів CV;

– TensorFlow – крім деяких хороших інструментів для обробки/класифікації зображень, він також допоможе застосувати глибоке навчання до комп'ютерного зору. За допомогою Python API можна виконувати розпізнавання облич та виразів. Серед недоліків TensorFlow: а) високе енергоспоживання; б) потреба в лозі ресурсів; і в) потрібна більш висока кваліфікація розробників;

– Keras – це бібліотека нейронних мереж з відкритим вихідним кодом, яка може працювати поверх TensorFlow, Microsoft Cognitive Toolkit, R, Theano або PlaidML. Це зручний, модульний та розширюваний інструмент, який розширює ваші можливості під час роботи з глибокими нейронними мережами;

– FastAI для PyTorch – ця бібліотека, розроблена некомерційною групою дослідників, спрощує навчання швидких та точних нейронних мереж з упором на глибоке навчання;

– бібліотека завантаження даних Nvidia (DALI) – це портативна бібліотека з відкритим вихідним кодом, яка забезпечує декодування та доповнення зображень, відео та мови для програм глибокого навчання;

– H2O.ai – це платформа машинного навчання з відкритим вихідним кодом та лінійною масштабованістю. Він підтримує більшість статистичних алгоритмів та алгоритмів машинного навчання, а також пропонує функції AutoML;

– OpenVINO (Visual Inference and Neural Network Optimization) – це інструментарій, написаний на C++ та Python. Він складається з безлічі попередньо навчених моделей, що дозволяє оптимізувати та розгорнути їх, а не розробляти;

– Kubeflow – це безкоштовна платформа машинного навчання з відкритим вихідним кодом, яка була розроблена для спрощення та координації робочих процесів машинного навчання у кластерах Kubernetes;

– Microsoft Cognitive Toolkit (колишній CNTK) – це набір інструментів з відкритим вихідним кодом для вирішення глибокого навчання. Це також один із перших наборів інструментів, що підтримують формат Open Neural Network Exchange ONNX;

– Matlab – чудовий інструмент для створення програм для обробки зображень. Цей інструмент допомагає створювати чіткий та лаконічний код (у порівнянні з C++), спрощуючи його читання та налагодження, а отже, і підтримку.

4.10 Аналіз методів отримання керуючих впливів для біонічних рук

Величезний прогрес в галузі матеріалознавства та технологій призвів до значних досягнень в області протезування кінцівок. Дослідники в даний час просто намагаються відтворити функціональність і діапазон рухів здорової людської кінцівки, що є дуже складним завданням.

Наприклад, для того, щоб підняти руку і зігнути-розігнути один палець декілька разів виконується дій.

По-перше, потрібно зігнути лікоть, одночасно піднімаючи передпліччя, щоб воно знаходилося в правильному положенні. Потім потрібно повернути передпліччя на потрібний кут, щоб палець міг виконати певні дії. Все це робиться, прикладаючи необхідне зусилля – створення роботизованої кінцівки,

яка б виконувала всі ці завдання плавно, легко і швидко, є досить складним завданням.

Нерви, м'язи, синапси, кора головного мозку – все це повинно працювати без збоїв, для виконання подібних завдань, тому взаємодія між думкою, дією і реакцією є актуальним завданням для розробників-дослідників всього світу, що намагаються відтворити це все в своїх біонічних технологіях.

Вже доступний ряд біонічних протезів, які починають імітувати деякі функціональні можливості втрачених кінцівок, але вони ще знаходяться на стадії досліджень і розробок, хоча і демонструють великі перспективи.

Існує п'ять типів біонічних систем управління рукою / пензлем:

- міоелектричне пряме керування;
- розпізнавання міоелектричних образів;
- розпізнавання рухів за допомогою хірургічно імплантованих міоелектричних датчиків;
- міоневральний інтерфейс агонист-антагоніст;
- нейронні інтерфейси.

У більшості використовуваних сьогодні біонічних рук використовуються міоелектричні системи прямого управління – один або два міоелектричних датчика вбудовані в гніздо в руці, так що вони прилягають до шкіри частини кінцівки, що залишилась після ампутації. Основне призначення цих датчиків – виявляти рух м'язів, призначених для (рис. 4.48):

- відкриття біонічної руки;
- закриття біонічної руки;
- спільного скорочення м'язів в обох напрямках.

Цей тип системи описується як пряме керування, тому що користувач повинен явно скорочувати певні м'язи, щоб безпосередньо керувати схватом, а також діями відкриття і закриття. Цей тип не є ідеальним для керування біонічною рукою вище ліктя – не генерує досить розпізнавальних сигналів для керування суглобами кисті, зап'ястя, ліктя і плеча.

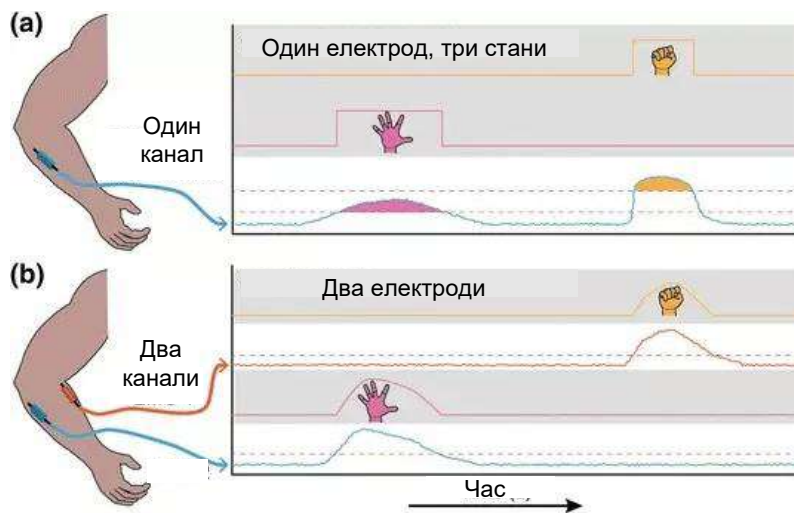


Рисунок 4.48 – Міоелектричне пряме керування

Зворотний зв'язок в цьому типі системи також обмежений – включає в себе розміщення датчиків тиску в біонічних пальцях і запуск невеликих вібрацій, щоб вказати контакт і/або ступінь опору. Однак також можна вказати на ковзання, температуру або різкість об'єкта, стимулюючи залишкову шкіру кінцівки іншими способами.

Нові версії біонічних рук / кистей тепер пропонують розпізнавання міоелектричних образів в якості альтернативи методам прямого керування. Для зчитування показань з датчиків (рис. 4.49), використовується зазвичай від 8 до 16.

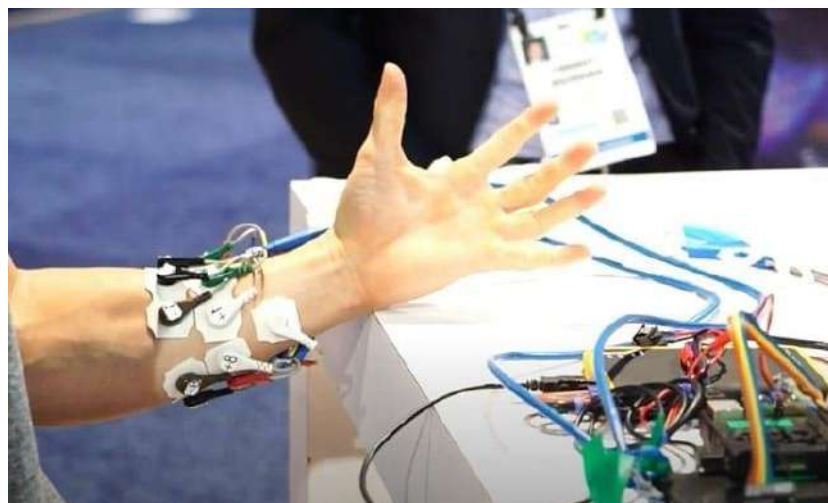


Рисунок 4.49 – Зчитування показань датчиків з руки

З такою кількістю датчиків можна виявити кілька моделей руху м'язів, а не тільки окремі рухи. Кожному шаблону можна зіставити певну дію. Наприклад, один шаблон може бути зіставлений з використанням схвату, інший – із

захопленням, третій – для переміщення ліктьового суглоба тощо. У більш просунутих системах шаблони можна навіть порівняти з рухами окремих пальців.

На прикладі біонічної руки процес відображення відбувається наступним чином:

- користувача просять виконати дію фантомної рукою, наприклад взяти олівець, що лежить на столі;

- це дія зазвичай вимагає захоплення об'єкту великим, вказівним і середнім пальцями;

- спроба користувача зробити це своєю фантомної рукою запускає м'язові рухи в частині кінцівки, що залишилась;

- цей шаблон співвідноситься з необхідними діями біонічної руки.

Отримані в результаті елемент управління набагато більш інтуїтивно зрозумілий, оскільки користувачеві не потрібно явно управляти рукою. Він просто думає про те, щоб рухати своєю фантомної рукою точно так, як якщо б вона все ще була присутня, і біонічна кінцівка буде відображати його дії.

Розпізнавання рухів за допомогою хірургічно імплантованих міоелектричних датчиків. Щоб уникнути проблем з підтриманням належного контакту між датчиками і шкірою, можна хірургічним шляхом імплантувати датчики в цільові м'язи залишкової кінцівки.

Цей підхід часто використовується в поєднанні з процесом, званім цілеспрямована реіннервація м'язів, коли нерви, які колись контролювали м'язи, згодом були втрачені в результаті ампутації, хірургічним шляхом перепризначувалися на нові м'язи. Так як пацієнту все одно знадобиться операція, є сенс вбудувати датчики (рис. 4.50).

Однак, крім фізичного розміщення датчиків, немає ніякої різниці в характері системи управління, тобто це все ще система розпізнавання міоелектричних образів.

Міоневрального інтерфейс агонист-антагонист – це технологія, яка в першу чергу пов'язана з хірургічним відновленням м'язових пар агонист-антагонист, які контролюють наші кінцівки. В основному, відрізані м'язи частини кінцівки, що залишилась повторно з'єднуються, щоб сформувати таку пару (рис. 4.51).

Це відновлює природне сприйняття мозком положення і рух відсутньої частини кінцівки. Це називається проприоцепція.

Потім суглоб калібрується так, щоб його руху відповідали рухам відновленої пари м'язів – мозок може керувати біонічної кінцівкою за допомогою думок, а також відчувати її положення в просторі.

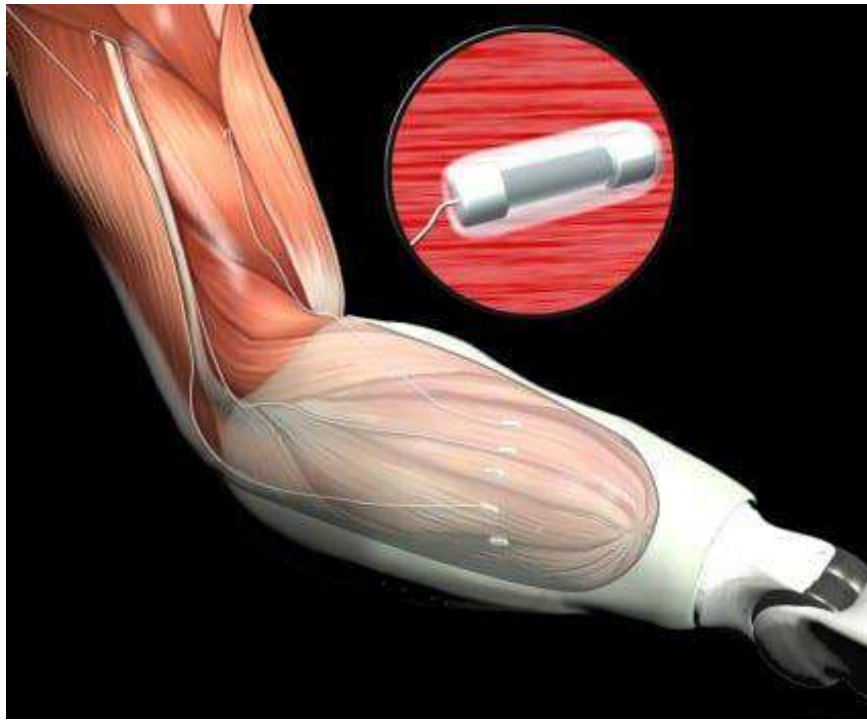


Рисунок 4.50 – Розпізнавання рухів за допомогою хірургічно імплантованих міоелектричних датчиків

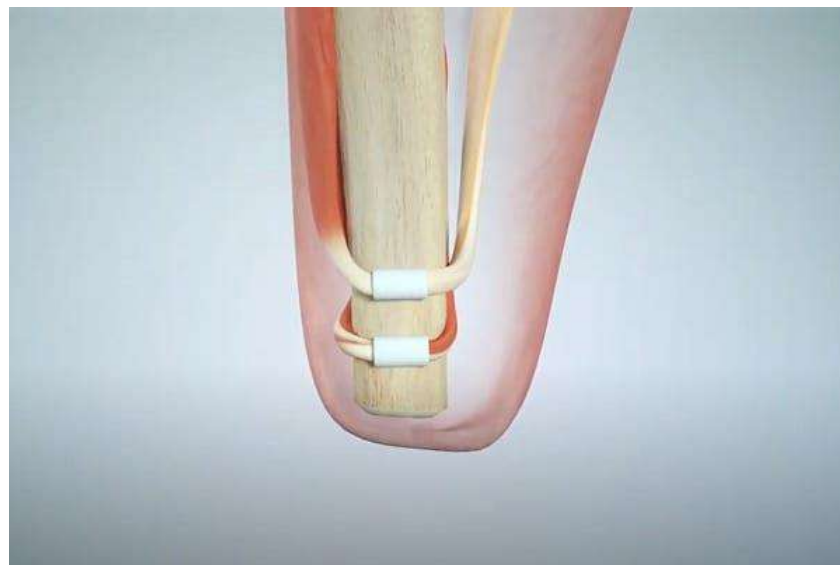


Рисунок 4.51 – З'єднання м'язів кінцівки, що залишилась

Справа в тому, що з точки зору управління біонічної кінцівкою це все ще міоелектрична система у якій набагато кращий зв'язок між мозком і рештою м'язів кінцівки, а також зменшується атрофія м'язів, тому рухи стають сильнішими. Цей інтерфейс використовують для генерації більш чітких міоелектричних сигналів, як з точки зору наміри користувача, так і з точки зору сили сигналу.

Як тільки біонічна кінцівка налаштована так, що її руху відповідають рухам пари м'язів, візуальні підказки потім підкріплюють цю модель, тобто користувач намагається підняти передню частину стопи до своєї гомілки, пара м'язів реагує відповідно, біонічна нога відображає цей рух, і очі користувача підтверджують це. Це створює систему управління зі зворотним зв'язком.

Основна мета повноцінного нейронного інтерфейсу – полегшити двосторонню зв'язок між користувачем і його біонічної рукою.

Є кілька способів зробити це. Проста модель керування досягається за допомогою імплантованих міоелектричних датчиків, в той час як сенсорний зворотний зв'язок передається від датчиків на біонічних пальцях до імплантованим електродів, прикріплених до нервової системи. Потім електроди стимулюють нерви, щоб змусити мозок відчувати правильні відчуття.

Реальний виграш тут досягається за рахунок сенсорного зворотного зв'язку системи. Користувачі можуть не тільки отримати базову зворотний зв'язок, наприклад, контакт і тиск. Також ведуться роботи по відтворенню інших аспектів дотику, таких як форма, прослизання і навіть температура.

Цей зв'язок також може значно поліпшити керування більш точними рухами рук, оскільки вона дозволяє виконувати невеликі інтерактивні налаштування між мозком і біонічної рукою на основі цієї зворотного зв'язку, що зменшить залежність користувача від візуального керівництва (рис. 4.52).

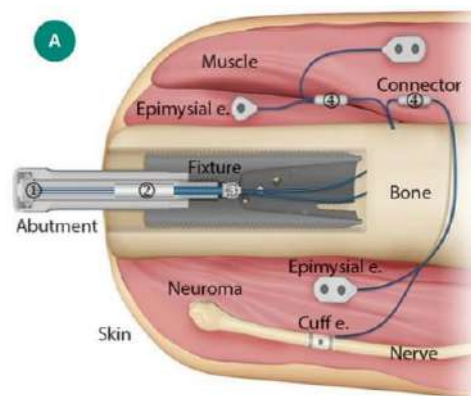


Рисунок 4.52 – Нейронний інтерфейс

М'язи скелетної мускулатури, скорочення яких приводять в рух кінцівки, отримують сигнал на скорочення або розслаблення від головного мозку. Провідником імпульсів служать нервові волокна, сама ж природа сигналів електрична. При цьому окремі ділянки м'язів іннервуються, тобто управляються, окремими нервовими волокнами, що йдуть безпосередньо в мозок. Це означає те, що навіть при ампутації нервові закінчення зберігаються в культі і можуть

проводити сигнали в обидві сторони. Саме остання обставина призводить іноді до фантомних болей, коли виникає імпульс в нервовому волокні, раніше пов'язаному з сенсорними клітинами, наприклад, кисті.

Біонічний протез руки являє собою складний комплекс. У гільзі розташовуються численні найточніші датчики, які здатні вловити електричні імпульси, передані на нервові закінчення. Вловити і передати на мікропроцесор, який обробляє та інтерпретує їх в зрозумілі для сервоприводів команди, завдяки чому відбувається рух кисті або руки в суглобі саме таким чином, яким його хоче зробити користувач. Хоча на сьогоднішній день швидкість роботи міоелектричних протезів рук поступається природним кінцівкам, інженери постійно працюють над удосконаленням та оптимізацією всіх процесів.

Протез руки відкриває перед користувачами нові горизонти активності, хоча і має ряд особливостей, які потрібно враховувати при виборі. Наприклад, він вимагає певного часу на адаптацію і навчання роботі з ним. Кожен такий протез налаштовується індивідуально, під конкретного носія. У цьому процесі беруть участь як протезисти, так і програмісти й інженери. Освоїться з таким пристроєм може бути досить непросто, в деяких випадках процес ускладнюється анатомічними і фізіологічними особливостями пацієнта (рис. 4.53).



Рисунок 4.53 – Біонічна рука

Зусилля, яке можна буде давати на протез, також буде в цьому випадку максимальним – адже більша частина м'язів збережена. Найскладніше протезувати за допомогою біонічних протезів пацієнтів з виокремлення плечового суглоба. Ціна протеза руки також залежить від рівня ампутації.

Зосередженість інженерів переважно на поліпшенні розпізнавання сигналів призвела до суттєвого підвищення точності рухів протеза, яка досягає часом 90%. Але це ж одночасно і дуже здорожує пристрою. Альтернативне рішення пропонують розробники з Університету Ньюкасла. Вони забезпечили

протез відеокамерою і програмою розпізнавання образів, що набагато простіше з технічної точки зору і дешевше. При тому, що піднесло протеза до предмету камера передає зображення, обробивши яке, процесор приймає рішення, яким чином краще взяти його.

Оскільки існує певна кількість основних схватів, які програмують в більш традиційних біонічних протезах, зробити цілком можливо.

Розпізнавання образів і їх інтерпретація в команди для сервоприводів проходять значно швидше, ніж зчитування і інтерпретація сигналів з нервових волокон. Людина ж тільки повинен піднести «руку» до предмету, який він хоче використовувати.

Ще більш широкі перспективи відкриває перед біонічної протезами, оснащеними відеокамерами, машинне навчання і нейромережі. Так, всі об'єкти, які «запам'ятовувала» система обробки чином, потім категоризувати за типом захоплення - і відповідно, в одній категорії могли виявитися зовсім різні об'єкти, але з однаковим способом хвата. Це дуже спрощувало звернення з протезом. При цьому використовується свёрточная нейросеть, найбільш підходяща саме для сприйняття візуальної інформації. Образи для засвоєння бралися з амстердамського банку зображень, що містить велику кількість фотографій предметів побуту для потреб дизайнерів.

Основна механічна функція людського пальця – відкривати і закривати. Це відтворюється в біонічному пальці, що використовує таку конструкцію (рис. 4.54).



Рисунок 4.54 – Біонічний палець

Електродвигун на батарейках надає руху систему зубчастих коліс для переміщення головного шарніру, який потім переміщує другий і третій шарніри через систему важелів тяг. Зверніть увагу, що існує безліч варіантів цього,

наприклад використання систем шківів для моделювання сухожилів замість систем стрижневих важелів.

Ключовою особливістю цього типу конструкції є те, що існує тільки один незалежний шарнір: шарнір (головний суглоб). Два інших суглоба автоматично слідує за цим суглобом, створюючи таку фіксовану схему руху (рис. 4.55).

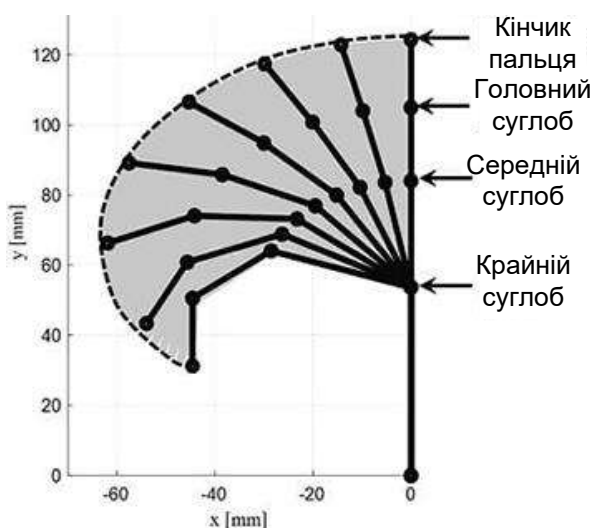


Рисунок 4.55 – Схема руху пальця

Для порівняння, природні пальці можуть рухати середній суглоб незалежно, а крайній суглоб – майже незалежно. Крім того, натуральні пальці можуть розкладатися, скручуватися і повертатися в безлічі комбінацій.

Рух природного великого пальця ще складніше. Крім відкривання і закривання, він може повертатися вниз і трохи назовні, щоб стосуватися всієї нижньої сторони будь-якого пальця і навіть частин долоні. Це дозволяє природним рукам хапати предмети будь-якої форми.

Незважаючи на свою механічну складність, цей тип великого пальця все ще має безліч обмежень (рис. 4.56). Так, він може позиціонувати себе, щоб стати протидією силою для багатьох типів схватів, але йому не вистачає спритності природного великого пальця.

Отже, біонічні руки – вражаюче досягнення інженерної думки, але вони набагато простіші, ніж природні руки. З одного боку, це не має значення, тому що вони все ще можуть виконувати широкий спектр завдань.

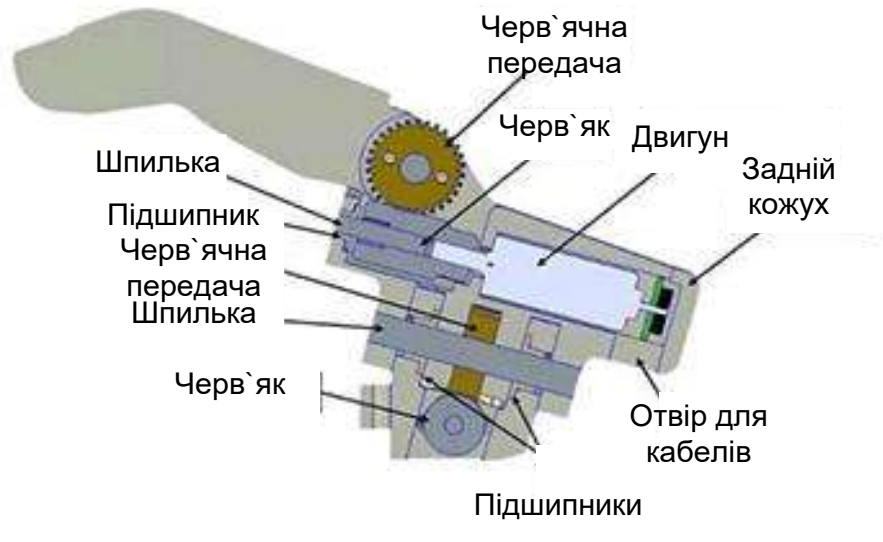


Рисунок 4.56 – Палець біонічної руки

4.11 Аналіз засобів комп'ютерного моделювання робочого простору транспортних роботів у сучасних системах

4.11.1 Microsoft Robotic Studio

Microsoft Robotics Developer Studio (Microsoft RDS, MRDS) - Windows-орієнтоване середовище для управління роботами та їх симуляції. Цей продукт призначений для академічної, аматорської або комерційної розробки та підтримує велику кількість різноманітного апаратного забезпечення роботів.

RDS заснована на бібліотеці CCR (Concurrency and Coordination Runtime), .NET-реалізації бібліотеки для роботи з паралельними та асинхронними потоками даних, використовуючи обмін повідомленнями, та DSS (Decentralized Software Services) – полегшений засіб створення розподілених додатків на основі сервісів, що передбачає управління безліччю сервісів для коригування поведінки загалом [4].

Серед особливостей:

мова візуального програмування Microsoft Visual Programming Language для;

створення та налагодження програмних додатків для роботів; - веб-орієнтовані та windows-орієнтовані інтерфейси;

3D симуляція (включаючи апаратне прискорення);

спрощений доступ до датчиків та виконавчих механізмів робота;

підтримку кількох мов, включаючи C#, Visual Basic .NET, JScript та IronPython.

RDS складається з 4-х основних компонентів:

CCR (Concurrency and CoordinationRuntime)

DSS (Decentralized SoftwareServices);

VPL (Visual Programming Language) ;

VSE (Visual Simulation Environment).

CCR і DSS також доступні окремо для застосування в комерційних додатках, що вимагають високого рівня паралелізму та/або можливості взаємодії між безліччю вузлів у мережі. Цей пакет називається CCR and DSS Toolkit (CCR та DSS інструментарій). На рис. 4.57 – структура програми Microsoft Robotic Studio.

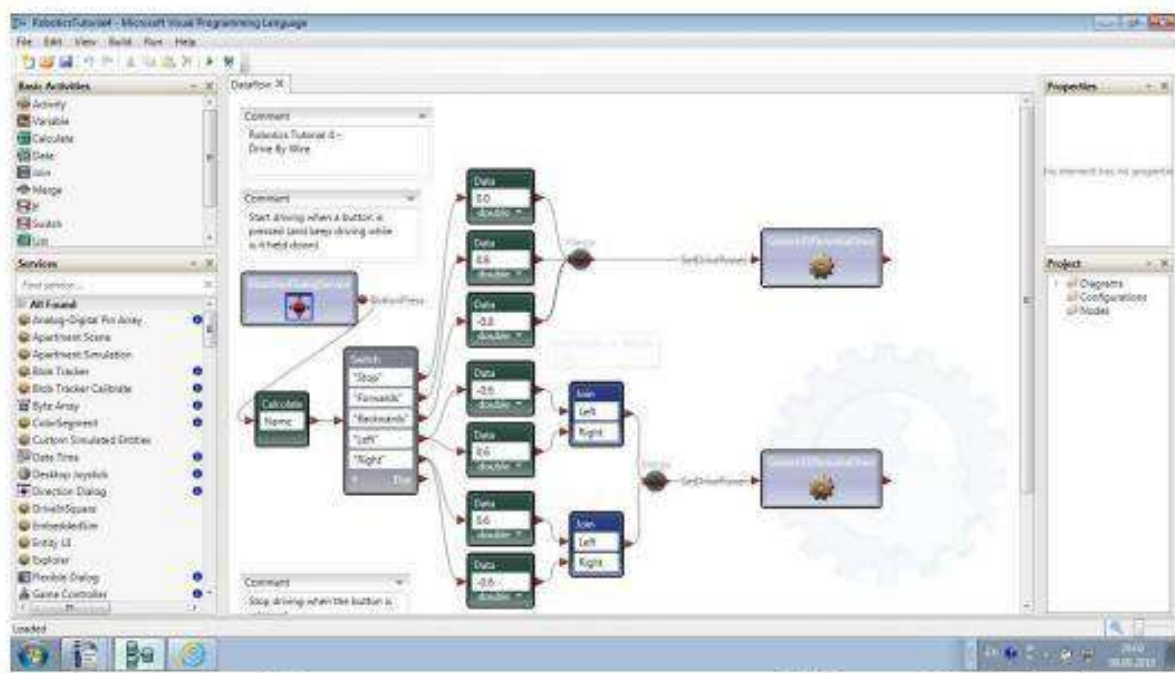


Рисунок 4.57 – Структура програми Microsoft Robotic Studio

Інструменти, що дозволяють розробляти MRDS-додатки, включають графічне середовище (Microsoft Visual Programming Language: VPL); утиліти командного рядка, що дозволяють взаємодіяти з проектами Visual Studio (версії VS Express цілком достатньо) мовами C#, C++ або VB.NET; підтримку скриптових мов, наприклад IronPython; та інструменти симуляції 3D.

Visual Programming Language - графічне середовище розробки, що використовує каталог сервісів та дій.

Вони можуть бути пов'язані графічним способом: обслуговування або дія представляються у вигляді блоку, у якого є введення та виведення, і його потрібно лише перетягнути з каталогу на діаграму. На рис. 4.58 – приклади відображення роботів у Microsoft Robotic Studio.

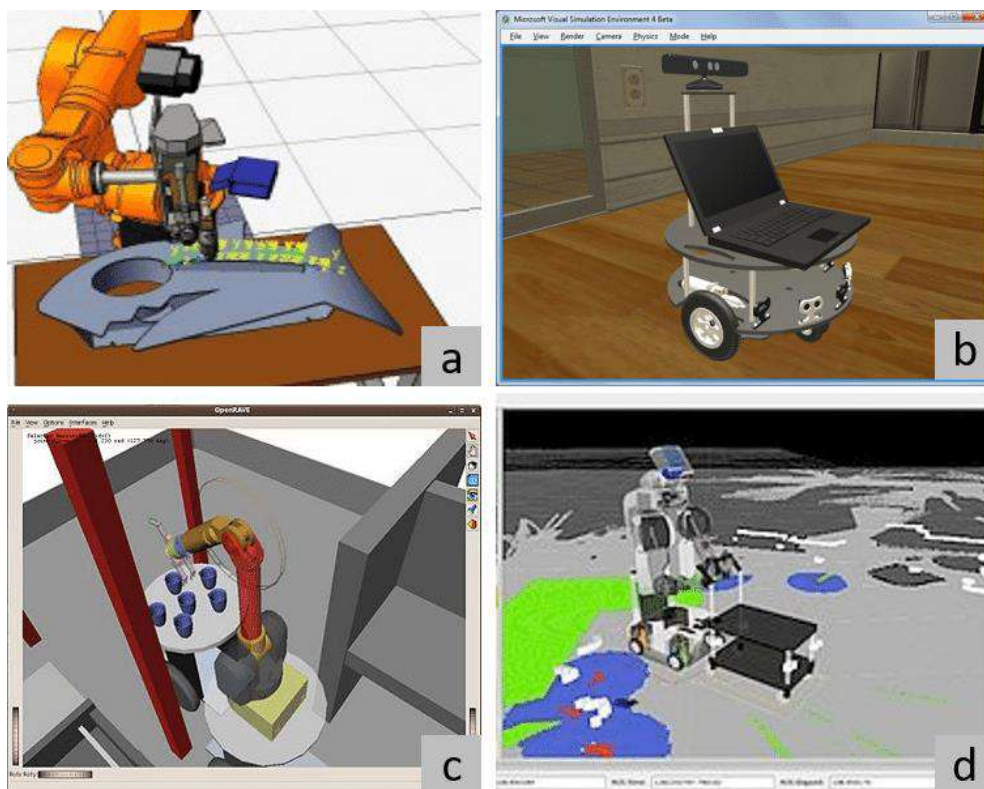


Рисунок 4.58 – Приклади відображення роботів у Microsoft Robotic Studio

Зв'язування можна зробити за допомогою миші, це дозволяє вам визначити, є сигнали синхронними чи ні, дозволяє вам виконувати операції, пов'язані з передачею значень та ін.

VPL також дозволяє вам генерувати код нових "макро"-сервісів на основі діаграм, створених користувачами.

У VPL можливе просте налаштування сервісів для різних апаратних елементів.

Середовище симуляції 3D в MRDS дозволяє вам симулювати поведінку роботів у віртуальному світі, використовуючи технологію NVIDIA PhysX (3D-двигок, який спочатку був написаний компанією Ageia), що дозволяє використовувати просунуту фізичну модель.

Безліч прикладів та посібників, доступних для різних інструментів, що прискорюють розуміння та засвоювання MRDS. У комплект додано кілька програм, серед них, наприклад, згадані вище Maze Simulator або Soccer Simulation, розроблені корпорацією Microsoft.

Також у стандартному та академічному виданнях MRDS присутні 3 невеликі симуляційні середовища, які є обмеженими версіями більших узагальнених симуляційних середовищ, розроблених компанією SimplySim.

Недоліки. Немає обліку та підтримки реального середовища експлуатації робота (тип поверхні його переміщення, погодні умови та ін.)

При керуванні реальним роботом використовується його симуляція, яка може повністю збігатися з реальним прототипом. Чим точніше модель, тим більше налаштувань вона потребує. Незважаючи на використання PhysX, фізика в MRDS є дуже спрощеною.

Інформація про найбільш значущі програми наводиться нижче.

Автономна машина Принстонського університету, створена в рамках програми DARPA Urban Grand Challenge, була повністю запрограмована у

MRDS. MySpace використовує MRDS для розробки додатків (не призначених для роботів), що використовуються в серверній (back end) частині їхнього сайту.

Університет Індіани використовує MRDS для розробки додатків (не призначених для роботів), які застосовуються для координації мережі високовитратних обчислень.

У 2008 році Microsoft запустила змагання з симуляції робототехніки під назвою RoboChamps, що використовує MRDS, з чотирьох категорій: універсальна, сумо, місто, марсохід. Симульоване середовище та роботи, задіяні у змаганні, були створені компанією SimplySim, а саме змагання спонсорували Kia Motors.

У категорії робототехніки та алгоритмізації змагання з програмного забезпечення Imagine Cup у 2009 році використовувалося середовище візуальної симуляції MRDS. Завдання цього змагання були розроблені компанією SimplySim і по суті були покращеними версіями завдань RoboChamps.

4.11.2 MatLab

MatLab(скорочення від англ. Matrix Laboratory) - пакет прикладних програм для вирішення завдань технічних обчислень. Пакет використовують більше мільйона інженерних та науковців, він працює на більшості сучасних операційних систем, включаючи Linux, macOS, Solaris (починаючи з версії R2010b, підтримка Solaris припинена) та Windows.

MatLab, як мову програмування, було розроблено Клівом Моулером (англ. Cleve Moler) наприкінці 1970-х років, коли він був деканом факультету комп'ютерних наук в Університеті Нью-Мексико. Метою розробки було завдання дати студентам факультету можливість використання програмних бібліотек Linpack та EISPACK без необхідності вивчення Фортрану. Незабаром нова мова поширилася серед інших університетів і була з великою цікавістю

зустріта вченими, які працюють у галузі прикладної математики. Досі в Інтернеті можна знайти версію 1982 року, написану на Фортрані, яка розповсюджується з відкритим вихідним кодом. Інженер Джон Літл (англ. John N. (Jack) Little) познайомився з цією мовою під час візиту Кліва Моулера до Стенфордського університету у 1983 році. Зрозумівши, що нова мова має великий комерційний потенціал, він об'єднався з Клівом Моулером і Стівом Бангертом (Steve Bangert). Спільними зусиллями вони переписали MATLAB на C і започаткували в 1984 компанію The MathWorks для подальшого розвитку. Ці переписані на бібліотеки довгий час були відомі під ім'ям JASCRAS. Спочатку MATLAB призначався для проектування систем управління (основна спеціальність Джона Літтла), але швидко завоював популярність у багатьох інших наукових та інженерних галузях. Він також широко використовувався і в освіті, зокрема для викладання лінійної алгебри та чисельних методів.

Опис мови. Мова MATLAB є високорівневою мовою програмування, що інтерпретується, що включає засновані на матрицях структури даних, широкий спектр функцій, інтегроване середовище розробки, об'єктно-орієнтовані можливості та інтерфейси до програм, написаних іншими мовами програмування [6].

Програми, написані на MATLAB, бувають двох типів – функції та скрипти. Функції мають вхідні та вихідні аргументи, а також власний робочий простір для зберігання проміжних результатів обчислень та змінних. Скрипти використовують загальний робочий простір. Як скрипти, і функції зберігаються як текстових файлів і компілюються в машинний код динамічно.

Існує також можливість зберігати так звані pre-parsed програми - функції та скрипти, оброблені у вигляді, зручний для машинного виконання. Загалом такі програми виконуються швидше за звичайні, особливо якщо функція містить команди побудови графіків.

Основною особливістю мови MATLAB є його широкі можливості для роботи з матрицями, які творці мови висловили у гасла «думай векторно» (англ. Think vectorized).

Набори інструментів. Для MATLAB є можливість створювати спеціальні набори інструментів (toolbox), що розширюють його функціональність. Набори інструментів є колекцією функцій та об'єктів, написаних мовою MATLAB для вирішення певного класу завдань. Компанія Mathworks постачає набори інструментів, які використовуються в багатьох областях, включаючи. На рис. 4.59 – MatLab.

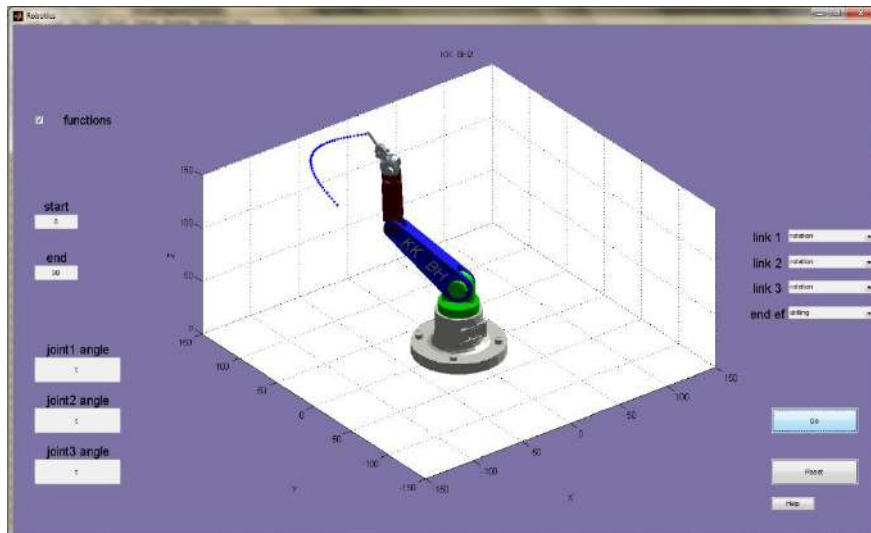


Рисунок 4.59 – Моделювання маніпулятора засобами MatLab

Цифрова обробка сигналів, зображень і даних: Signal Processing Toolbox (з'явився в 1987), DSP System Toolbox, Image Processing Toolbox (з'явився в 1993), Wavelet Toolbox, Communications System Toolbox - набори функцій і об'єктів, дозволяють вирішувати широкий спектр завдань обробки сигналів, зображень, проектування цифрових фільтрів та систем зв'язку.

Системи управління: Control Systems Toolbox, Robust Control Toolbox, System Identification Toolbox, Model Predictive Control Toolbox, Model-Based Calibration Toolbox - набори функцій та об'єктів, що полегшують аналіз та синтез динамічних систем, проектування, моделювання та ідентифікацію систем управління, включаючи сучасні алгоритми управління, такі як робастне управління, H_∞ -управління, ЛМН-синтез, μ -синтез та інші.

Фінансовий аналіз: Econometrics Toolbox, Financial Instruments Toolbox, Financial Toolbox, Datafeed Toolbox, Trading Toolbox - набори функцій та об'єктів, що дозволяють швидко та ефективно збирати, обробляти та передавати різноманітну фінансову інформацію.

Аналіз та синтез географічних карт, включаючи тривимірні: Mapping Toolbox.

Збір та аналіз експериментальних даних: Data Acquisition Toolbox, Image Acquisition Toolbox, Instrument Control Toolbox, OPC Toolbox – набори функцій та об'єктів, що дозволяють зберігати та обробляти дані, отримані в ході експериментів, у тому числі в реальному часі. Підтримується широкий спектр наукового та інженерного вимірювального обладнання.

Візуалізація та подання даних: Virtual Reality Toolbox – дозволяє створювати інтерактивні світи та візуалізувати наукову інформацію за допомогою технологій віртуальної реальності та мови VRML.

Засоби розробки: MATLAB Builder for COM, MATLAB Builder for Excel, MATLAB Builder for NET, MATLAB Compiler, HDL Coder — інструменти, що дозволяють створювати незалежні програми із середовища MATLAB.

Взаємодія із зовнішніми програмними продуктами: MATLAB Report Generator, Excel Link, Database Toolbox, MATLAB Web Server, Link for ModelSim – набори функцій, що дозволяють зберігати дані різних видів таким чином, щоб інші програми могли працювати з ними.

Бази даних: Database Toolbox - інструменти роботи з базами даних. Наукові та математичні пакети: Bioinformatics Toolbox, Curve Fitting

Toolbox, Fixed-Point Toolbox, Optimization Toolbox, Global Optimization Toolbox, Partial Differential Equation Toolbox, Statistics and Machine Learning Toolbox, RF Toolbox та інженерних завдань, включаючи розробку генетичних алгоритмів, розв'язання задач у приватних похідних, цілі проблеми, оптимізацію систем та інші.

Нейронні мережі: Neural Network Toolbox – інструменти для синтезу та аналізу нейронних мереж.

Нечітка логіка: Fuzzy Logic Toolbox - інструменти для побудови та аналізу нечітких множин.

Символьні обчислення: Symbolic Math Toolbox (з'явився в 1993)

- інструменти для символьних обчислень з можливістю взаємодії з символьним процесором програми Maple.

Крім перерахованих вище, існують тисячі інших наборів інструментів для MATLAB, написаних іншими компаніями та ентузіастами.

4.11.3 Robotino SIM Demo

Robotino SIM Demo – ідеальне віртуальне середовище для роботи з Robotino. Виявлення відмінностей від реальної поведінки лежить в основі засвоєння, аналізу та розуміння нових фізичних та технічних явищ. За допомогою програми можна створювати будь-які віртуальні тривимірні робочі середовища для Robotino, а потім моделювати створену програму. ПЗ доступне чотирма мовами (de/en/es/fr); Вибрати мову можна безпосередньо у процесі роботи. На рис. 4.60 – приклад моделювання у Robotino SIM Demo.

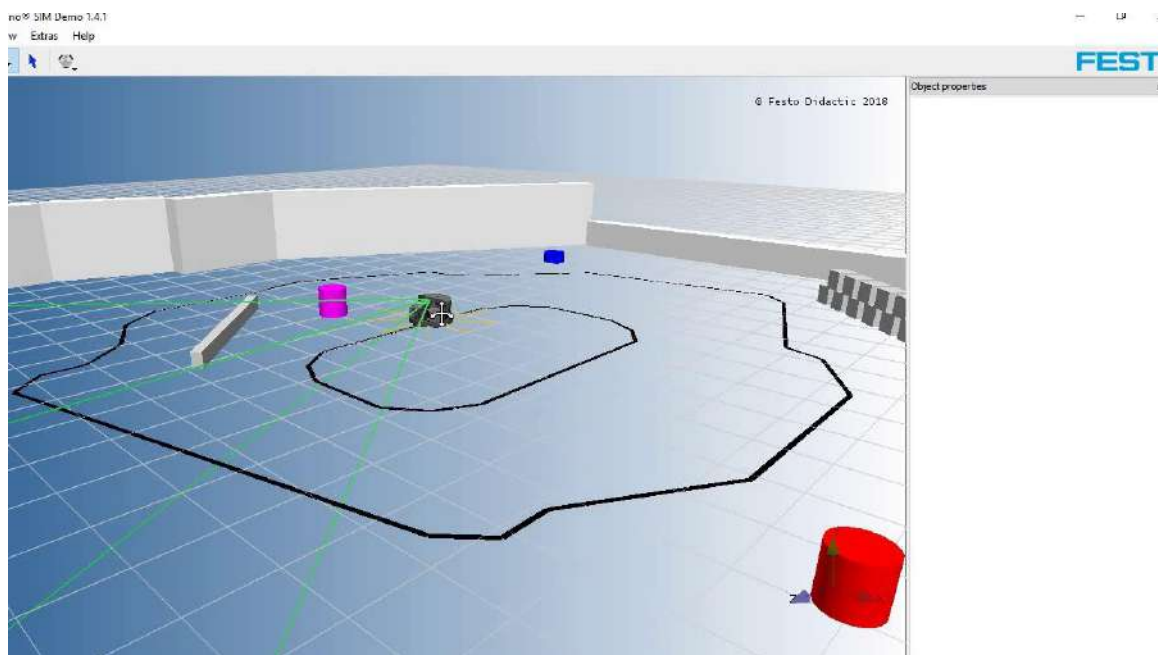


Рисунок 4.60 - Приклад моделювання у Robotino SIM Demo

Програмна модель Robotino включає геометричну модель з трьома всеспрямованими приводами, двома індуктивними аналоговими датчиками, двома цифровими оптичними датчиками, дев'ятьма датчиками визначення відстані, однією камерою, датчиком зіткнень у захисній накладці рами.

Бібліотека Robotino містить додаткові елементи, такі як захвати, лотки, лазерні датчики та заготовки. Про наявність нових елементів у бібліотеці користувач повідомляється через Інтернет.

Редактор та багата бібліотека моделей програми LabCreator забезпечують чудове робоче середовище для простого та швидкого створення привабливих віртуальних сценаріїв для Robotino®.

Програмування Robotino® може здійснюватися за допомогою ПЗ Robotino View, а також однією з мов високого рівня C, C++, C# або Java. Для цього потрібний відповідний інтерфейс API.

Для використання ПЗ потрібна операційна система Windows 2000, XP, VISTA або Windows 7. Для тривимірної візуалізації потрібен графічний адаптер з не менше ніж 128 МБ RAM та підтримкою OpenGL.

4.12 Розробка програмного забезпечення для мобільного робота Festo Robotino

4.12.1 Особливості розробки програмного забезпечення Festo Robotino

Особливістю управління програмно-апаратним комплексом Robotino полягає в його універсальності. Його програмувати можна як у візуальній середовищі розробки «Robotino View», так і за допомогою великої кількості бібліотек. Такі бібліотеки є практично для всіх популярних мов програмування. Компанія «Festo-Didactic» надає продукт для візуальної симуляції роботи робота Robotino. У вільному доступі є одна стандартна сцена, на якій можна проводити випробування.

Особливістю управління програмно-апаратним комплексом Robotino полягає в його універсальності. Його програмувати можна як у візуальній середовищі розробки «Robotino View» так і за допомогою великої кількості бібліотек. Такі бібліотеки є практично для всіх популярних мов програмування. Компанія «Festo – Didactic» надає продукт для візуальної симуляції роботи робота Robotino. У вільному доступі є одна стандартна сцена, на якій можна проводити випробування/

Етапи проведення досліджень:

- аналіз алгоритму JPS;
- модифікація алгоритму JPS;
- аналіз ПЗ робота;
- програмування модифікованого алгоритму.

Розглянемо алгоритм JPS. Даний алгоритм - покращення алгоритму пошуку шляху A*. Він покращується за рахунок «перескокування» тих позицій, які мають бути переглянуті. Також слід зазначити, що алгоритм JPS не вимагає додаткової обробки та витрат пам'яті. Розглянемо рішення, яке дає цей алгоритм (рис. 4.61 а).



Рисунок 4.61 – Розв'язок елементів JPS його модифікація

Як видно з рисунка 4.61 а, шукається оптимальний шлях з точки SP1 до точки DP 1. Сірим кольором показані стрибкові точки. алгоритм JPS дозволяє знаходити рішення лише для матеріальної точки.

Як очевидно, здійснюється пошук оптимального шляху з точки SP1 в точку DP1. Сірий колір показує стрибкові точки. Алгоритм JPS дозволяє шукати рішення лише матеріальної точки. Для Robotino цей алгоритм не підійде, тому потрібна його модифікація. Вона полягає в тому, щоб уникнути перешкод з урахуванням фізичних розмірів робота без зіткнення з ними. Модифікація алгоритму JPS полягає в тому, що аналізуються всі діагональні переходи, і якщо зустрічаються перешкоди, шукається альтернативний шлях для даного відрізка.

У візуальному середовищі розробки Robotino view запрограмувати робота можна за допомогою функціональних блоків, які в свою чергу розбиті на категорії. На рис. 4.62 зображено вікно даного середовища розробки.

На правій панелі є категорії з базовими елементами: Functional block library, Robotino, Variables. У категорії "Functional block library" є набір логічних, математичних операцій, обробка навігації тощо. Категорія Robotino містить список пристроїв робота, які можна використовувати. Наприклад, у "Drive system" є можливість керувати кожним двигуном окремо. Елементами категорії "Variables" є змінні проекту. При створенні великого проекту його можна розділити на складові, які розміщують на окремих вкладках. У центральній сцені користувач формує схему окремої компоненти проекту як послідовність елементів керування. Під час аналізу програмного забезпечення Robotino було виділено 2 групи програмних засобів: – Robotino View для візуального програмування; – Robotino SIM для моделювання та візуалізації поведінки. За описом системи програмування Robotino була запропонована ієрархія побудови програмного забезпечення (рис. 4.63).

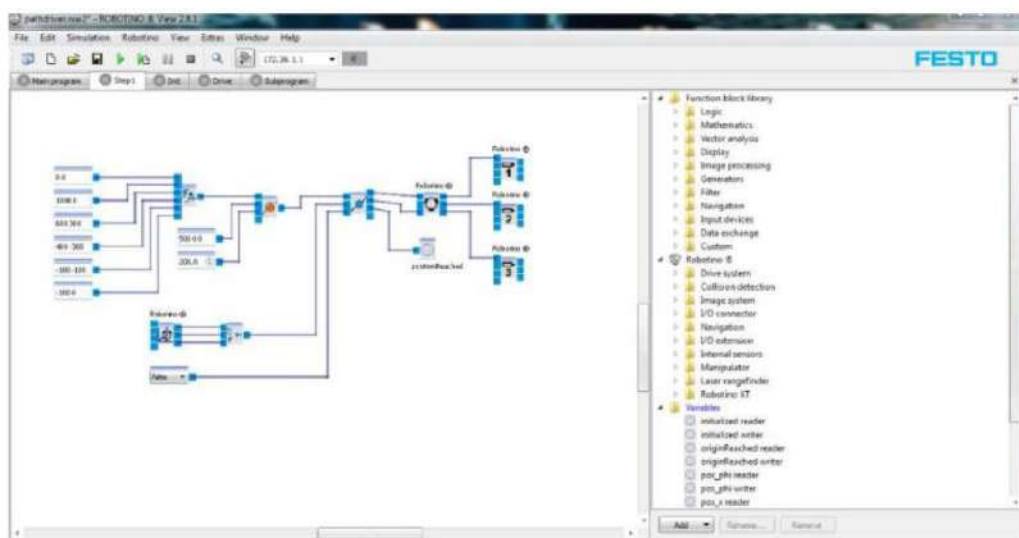


Рисунок 4.63 – Вікно "Robotino view"

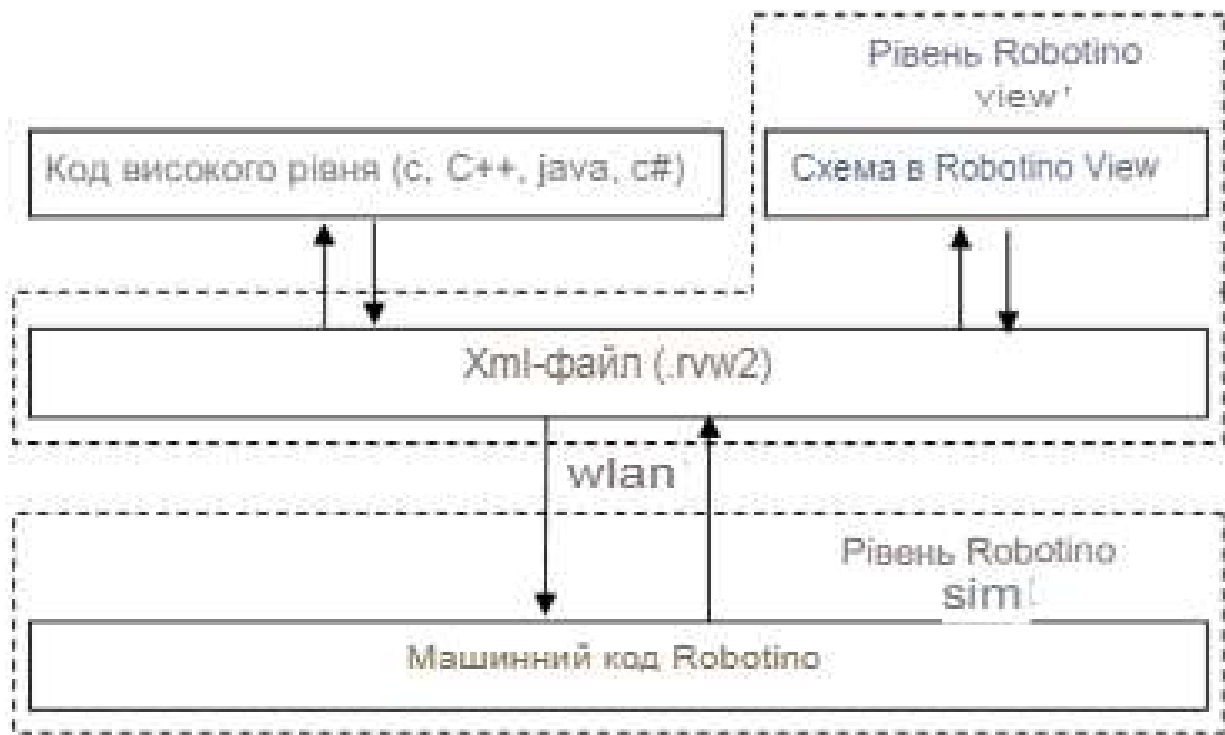


Рисунок 4.63 – Ієрархія побудови ПЗ для Robotino

У цій схемі є два рівні: рівень Robotino view – рівень побудови схеми для управління роботом, і рівень Robotino SIM – рівень представлення даних. Взаємодія між машинним кодом робота та кодом високого рівня здійснюється за допомогою .rvw2-файлу. Цей файл є версткою xml, причому кожен елемент має свою позицію у файлі. Керування роботом здійснюється дистанційно по бездротовому каналу W LAN (взаємодія між рівнем xml-файлу і машинного коду Robotino). Слід також зазначити, що дані з робота (наприклад дані з датчиків) передаються тим же каналом [3]. У цьому роботі програмування здійснювалося з допомогою мови C#. Компанія-виробник "Festo-Didactic" надає засоби для програмування Robotino у вигляді бібліотеки. Програмування модифікованого алгоритму виконувалось у Microsoft Visual Studio 2008 засобами мови C#. Тестування проводилися в оболонці Robotino SIM, вона дозволяє симулювати переміщення робота. Оцінка даного алгоритму та його модифікації виглядає так (див. табл. 4.1):

Таблиця 4.1 – Оцінки алгоритму (N, M - розмірність карти, K - кількість кроків)

Оцінка алгоритму JPS	Оцінка алгоритму JPS з модифікацією
$F(N, M, K) = (4N + 2M)^K$	$F(N, M, K) = (4N + 2M)^K + K - 1$

Було проаналізовано ієрархію побудови програмного забезпечення для робота Robotino. Також було розроблено програму та реалізовано алгоритм JPS з модифікацією, що дозволило його застосувати для робота. Виконано оцінку складності даних алгоритмів

4.12.2 Розробка загального алгоритму програми

Robotino має три двигуни, кожен з цих входів визначено поруч із двигунами, як показано на рис. 4.64.

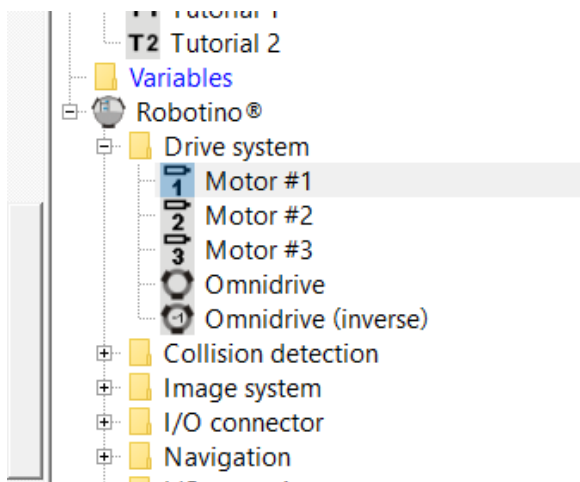


Рисунок 4.64 – Інструкції двигуна Robotino в меню Robotino View

На рисунку 4.65 ми будемо використовувати елемент OmniDrive, який допомагає керувати двигунами, а потім перемістивши панель керування, як на рисунку 4.66, і підключивши спочатку її напрямок X, який є плаваючою точкою, до плаваючої точки у напрямку X пристрою. omni drive і клацніть на VY і перетягніть його до VY, потім до Omega і перетягніть його, як показано на рисунку 4.67.

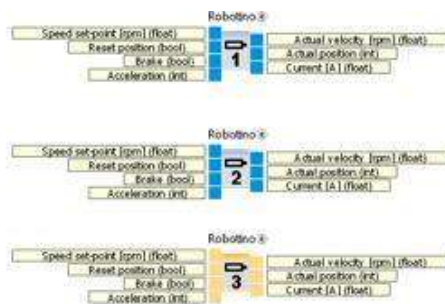


Рисунок 4.66 – Вхідні/вихідні дані двигунів Omnidrive

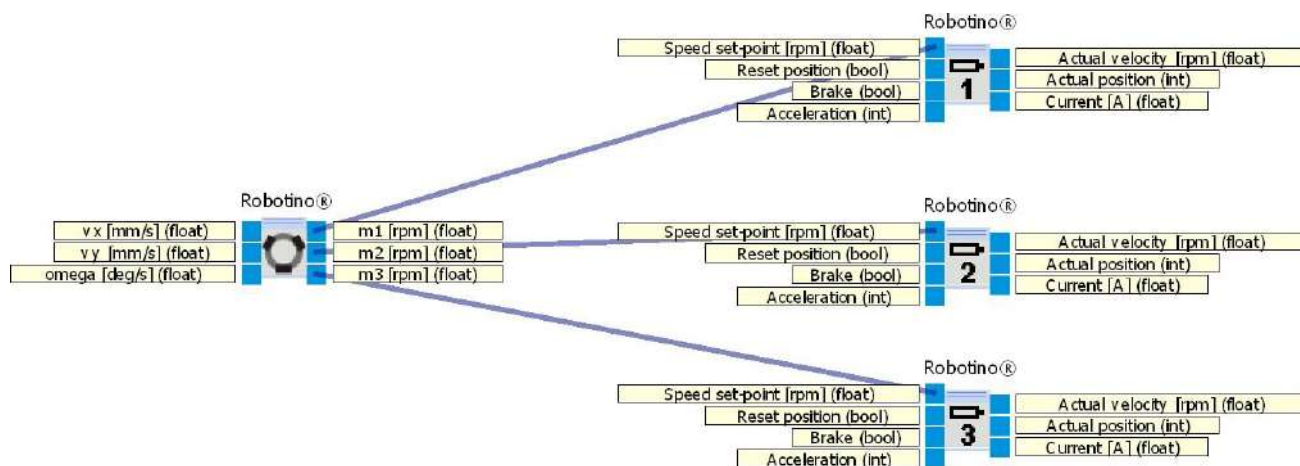


Рисунок 4.67 – Підключення Omnidrive з двигунами

Для відображення режиму ручного керування двигуни та Омні-привод повинні бути підключені до елемента панелі керування, вибраного з підпункту «Пристрої введення», як на рисунку 4.68.

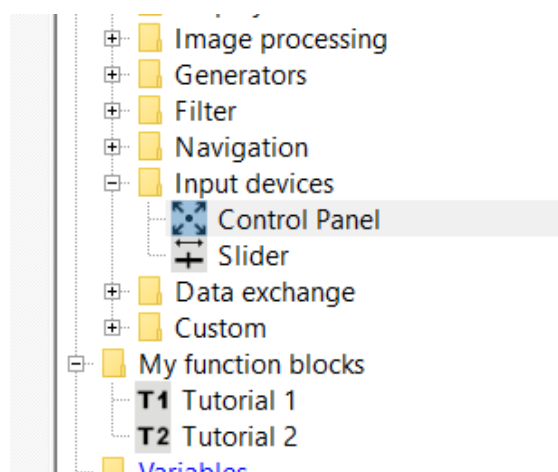


Рисунок 4.68 – Бібліотека елементів керування

Натиснувши на Панель керування, користувач може отримати панель керування вручну (див. рисунок 4.69).

Після того, як ми завершили підключення панелі керування та універсального приводу з двигунами, тож тепер у нас є невелика програма, і ми можемо почати перше завдання та керувати Роботіно, нам потрібно відкрити симулятор (ROBOTINO SIM Demo) і підключити цю програму до цього імітованого робота, і ми виберемо IP-адресу симулятора 127.0.0.1:8080, як показано на рисунку 4.70. Результат ручного керування можна побачити за допомогою Robotino SIM (див. рис. 4.71).

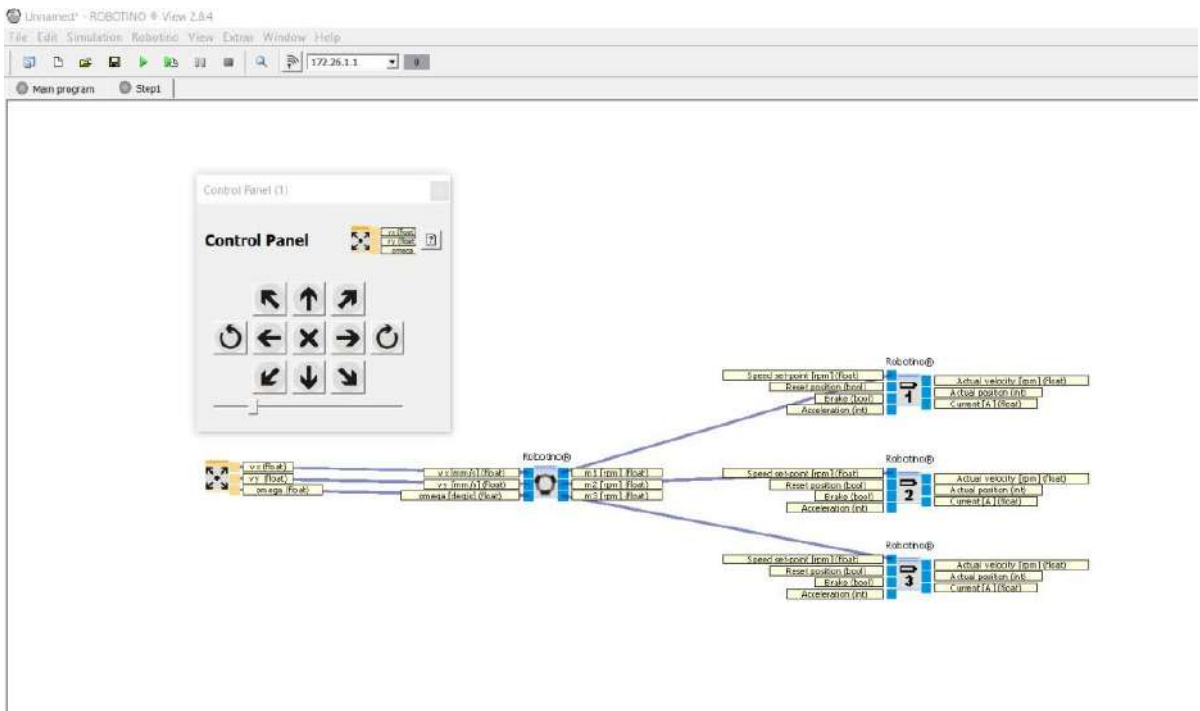


Рисунок 4.69 – Підключення панелі керування до OmniDrive

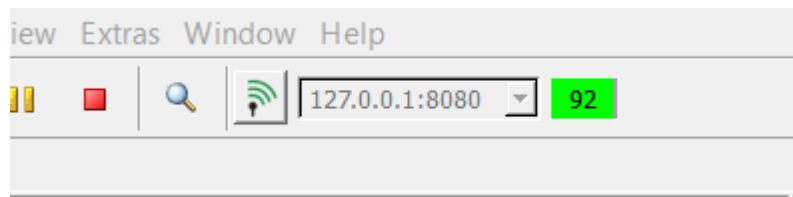


Рисунок 4.70 – Підключення до симулятора

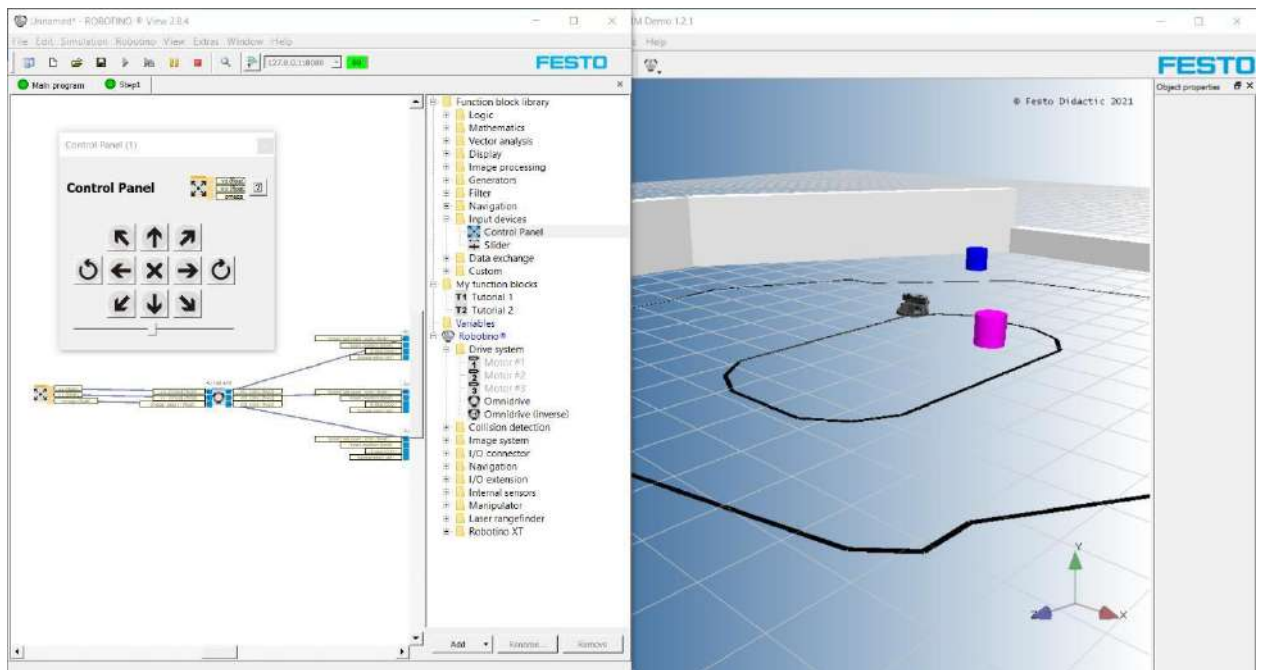


Рисунок 4.71 – Підключений Robotion View і Robotino SIM

Ми можемо використовувати ці стрілки на панелі керування, щоб переміщати Robotino в деяких напрямках і обертати його, а також ми можемо контролювати швидкість роботи Robotino.

Для різного контролю робочих просторів, а також слідування дорожніх знаків, Robotino може використовувати вбудовану веб-камеру, яка представлена спеціальним елементом «Камера» підпункту «Система зображень» (див. рис. 4.72).



Рисунок 4.72 – Камера елемента та її вікно з Robotino SIM

4.12.3 Розробка програми пересування робота за заданою траєкторією в Robotino View

Завдання – визначити програму керування мобільним роботом за заданою траєкторією. Суть програми полягає у встановленні конкретних координат, за якими робот буде рухатися відносно вихідної позиції.

Порядок розробки програми для визначення траєкторії руху робота:

Визначте змінні, які описуватимуть траєкторію руху робота. Наприклад, дві точки pos1 і pos2 повинні бути описані компонентами:

- pos1_x - x-компонента точки pos1;
- pos1_y - y-компонент точки pos1;
- pos1_phi - кутова складова точки pos1;
- pos2_x - x-компонента точки pos2;
- pos2_y - y-компонент точки pos2;
- pos2_phi - кутова склад'ова точки pos2.

Опишіть поточне положення змінної робота. pos у компоненті pos_x, pos_y, pos_phi. Для перенесення значень точок траєкторії в поточне положення створіть підпрограми типу setPos1 setPos2 тощо згідно з рисунком 4.73.

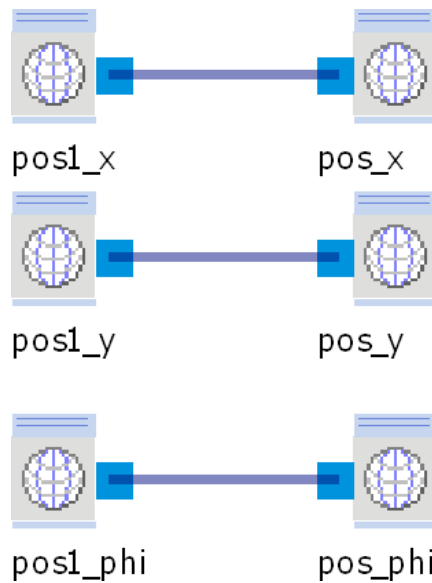


Рисунок 4.73 – Передача даних точки траєкторії pos1 до поточної позиції pos

Для забезпечення руху робота реалізувати підпрограму Drive (рис. 4.74). Основна ідея цієї процедури полягає у встановленні поточної точки «pos» (з внутрішніми компонентами x, y і phi) як вхідної інформації для драйвера позиції, керованого елементом Odometry. Вихід драйвера положення стає вхідним для OmniDrive, а потім для окремих двигунів платформи.

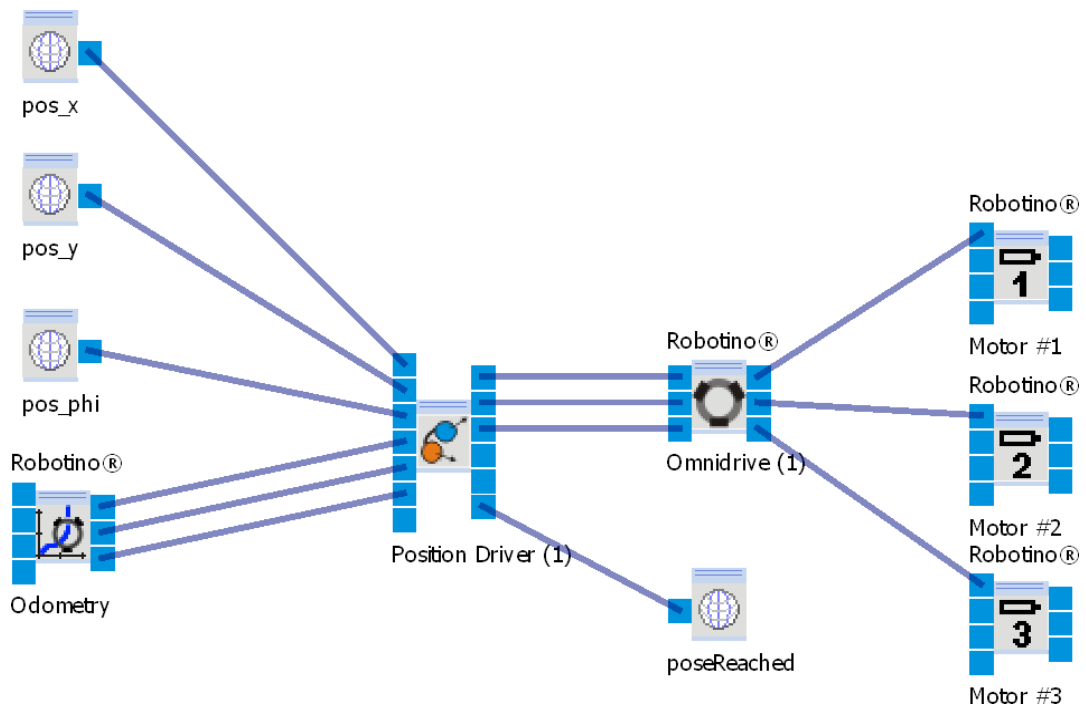


Рисунок 4.74 – Вид процедури Drive для керування Robotino

Визначте процедуру ініціалізації ініціалізації згідно з рисунком 4.75 елементів константа, одометрія, таймер, змінна пауза (тайм-аут).

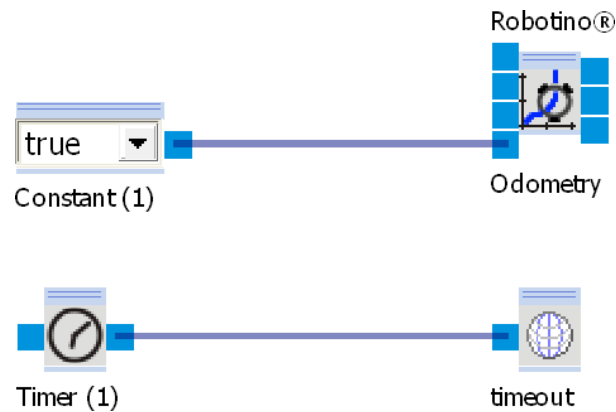


Рисунок 4.75 – Процедура ініціалізації Robotino

Підпрограми слід зв'язати шляхом визначення елементів основної програми у вигляді послідовної схеми (рисунок 4.76).

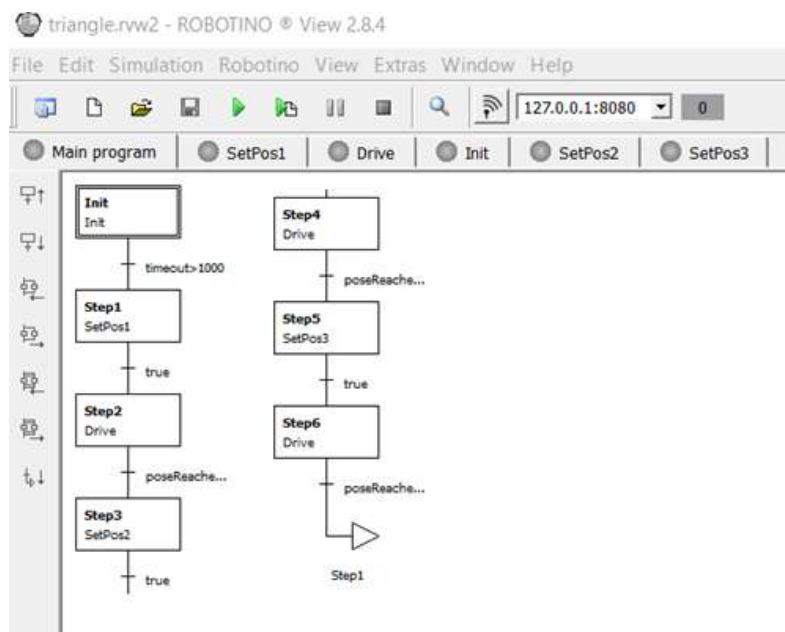


Рисунок 4.76 – Опис алгоритму основної програми

4.12.4 Тестування розроблених алгоритмів та процедур у Robotino SIM

Для моніторингу робочих областей різного типу можна визначити декілька шляхів. Одним з найпростіших є використання трикутної траєкторії. У цьому завданні ми змусимо робота пройти три точки, щоб сформувати трикутник. Спочатку ми визначимо змінні програми, тому в Robotino View будемо

використовувати «Додати змінну», щоб додати точки у складі X, Y і Z - координат.

Отже, ми розділимо змінні, такі як компоненти X-Pos1, Y-Pos1 і те саме для позиції 2 і 3.

Порядок розробки програми для визначення траєкторії руху робота:

- визначити змінні, які описують траєкторію руху робота, наприклад, дві точки pos1 і pos2 повинні бути описані компонентами: pos1_x - x-компонента точки pos1; pos1_y - y-компонент точки pos1; pos1_phi - кутова складова точки pos1; pos2_x - x-компонента точки pos2; pos2_y - y-компонент точки pos2; pos2_phi - кутова складова точки pos2;
- описати поточну позицію робота є змінною pos у компоненті pos_x, pos_y, pos_phi;
- щоб перенести значення точок траєкторії в поточну позицію, створити підпрограми типу setPos1 setPos2 і т. д. згідно з рисунком 4.78;
- вибрати для трикутника точки, як показано на рис. 4.77 для pos1_y = 1000 , для pos2_x = 500 і для pos2_y = 1000. Рисунок 4.78 – алгоритм програми трикутника (у презентації Robotino View).

Name	Type	Value
pos1_phi	float	0.000000
pos1_x	float	0.000000
pos1_y	float	1000.000000
pos2_phi	float	0.000000
pos2_x	float	500.000000
pos2_y	float	1000.000000
pos3_x	float	0.000000
pos3_y	float	0.000000
pos_3_phi	float	0.000000
pos_phi	float	0.000000
pos_x	float	0.000000
pos_y	float	0.000000
poseReached	float	0.000000
timeout	float	0.000000

Рисунок 4.77 – Таблиця значень точок траєкторії переміщення

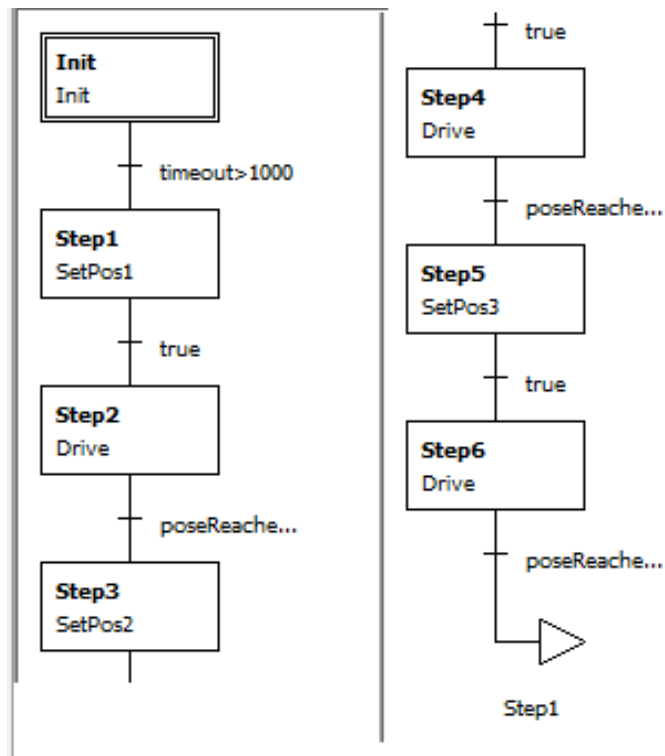


Рисунок 4.78 – Алгоритм програми трикутника (у презентації Robotino View)

4.13 Вибір апаратного та програмного забезпечення мобільної платформи

4.13.1 Розробка архітектури мобільної платформи

Для того, щоб розробити архітектуру мобільної платформи, необхідно включити всі наведені компоненти. Нижче буде описано за які функції вони відповідають.

Рама платформи – фізична система, яка утримує двигуни, електроніку, датчики та акумулятори.

Живлення – більшість мобільних роботів використовують літій-іонні акумулятори, які можна перезаряджати.

Двигуни та сервоприводи – роботи переміщуються і виконують дії, пов'язані з рухом (динамічні системи). Для цих цілей зазвичай використовуються двигуни постійного струму і сервоприводи.

Вбудована процесорна плата: всім роботам потрібна або плата мікроконтролера, або плата мікропроцесора, або перша і друга. Потрібно використовувати контакти вводу / виводу (аналогові та цифрові) для керування датчиками та сигналами управління, сигналів ШІМ для управління двигунами та сервоприводами, а також можливість використання послідовних зв'язків.

Датчики: це пристрої, які вимірюють характеристики реального світу і надають інформацію процесору, щоб змінити його поведінку або будь-яким чином вплинути на навколишнє середовище. Наприклад, такі датчики:

- датчик відстані;
- датчик зображення (камера);
- інерційний датчик / магнітний датчик;
- датчик температури.

Програмне забезпечення: всі мобільні роботи повинні мати унікальне для кожного робота програмне забезпечення, яке працює в реальному часі. Зазвичай це програмне забезпечення написано з використанням C, C ++, Python та ін. Зазвичай використовується інтегрована середовище розробки (IDE) для написання, налагодження і тестування коду. Приклади: Eclipse, Microsoft Visual Studio, Arduino Sketch Pad, Atmel Studio і інструменти на основі Linux.

Двигун (привід, мотор) – невід'ємна частина робота, яка призводить не тільки робота в рух, але і різні механізми або маніпулятори, якими оснащений робот. Коротше кажучи, двигун робота перетворює електричну енергію в кінетичну.

По потужності можна класифікувати двигуни так:

- двигуни постійного струму з редуктором. найпотужніший мотор, який можна використовувати практично в будь-якому типі роботів;
- серводвигуни. використовується для роботів вагою менше 2,5 кг. і в типах роботів з ногами;
- крокові двигуни. мабуть, найслабші з них використовуються в маленьких і легких роботах.

Кожен з цих трьох типів двигунів має свої переваги і недоліки:

Переваги електродвигунів постійного струму: легко підключити, найпотужніші, широкий спектр двигунів легко доступні на ринку.

Недоліки електродвигунів постійного струму: Занадто швидкі, необхідний редуктор, велике споживання електроенергії, важко встановити колеса, велика вартість.

Через це, можна зробити висновок що електродвигуни постійного струму найкраще всього підходять для великих роботів.

Переваги серводвигунів: середнє споживання енергії, простота установки, відповідна потужність для маленьких роботів, не такі дорогі.

Недоліки серводвигунів: не підходять для великих роботів, досить низька швидкість.

Серводвигуни найкраще за все підходять для невеликих роботів або роботів з ногами.

Переваги крокових двигунів: хороша швидкість, точний контроль, не велика вартість.

Недоліки крокових двигунів: складне управління, важко встановити колеса, об'ємні, велике споживання енергії, занадто маленька потужність.

Для роботів, які їздять по заданій лінії, найкраще підходять крокові двигуни.

Сервопривод – це механічний привід будь-якого типу, до складу якого входить блок управління приводом, який автоматично підтримує необхідні параметри датчика відповідно до заданого зовнішнім значенням. До складу сервоприводу входять:

- привід;
- датчик зворотного зв'язку;
- блок живлення і управління;
- вхід / конвертер / датчик сигналу.



Рисунок 4.79 – Структурна схема транспортного модуля



Рисунок 4.80 – Функціональна схема мобільного робота

4.13.2 Визначення складових апаратного забезпечення мобільної платформи

На сьогоднішній день існує дуже велика кількість мікроконтролерів, був проведений аналіз і порівняння найпоширеніших і найпопулярніших з них в порівнянні з Arduino Uno R3, яку будемо використовувати в мобільному роботі.

Raspberry Pi Zero. Одноплатний комп'ютер, який підходить для самих різних проектів – від побутової техніки з виходом в Інтернет до будівельних роботів (рис. 4.82). Система працює на чіпі Broadcom BCM2835. Оснащений слотом micro-SD, роз'ємом miniHDMI і 512 МБ оперативної пам'яті LPDDR2. Також пристрій має два роз'єми micro-USB для передачі даних і потужності, 40-контактний нерозпаяний GPIO, композитний відеовихід.

Zero володіє всіма перевагами оригінального Raspberry Pi A+. Він може з комфортом запускати Raspbian. Але Raspberry Pi Zero має два недоліки в двох основних областях: мережі і USB-підключення. У нього всього два порти micro USB, і тут немає ні Ethernet, ні Bluetooth, ні Wi-Fi. Щоб додати або розширити цю функціональність, потрібно буде купити адаптер.

Raspberry Pi Zero пропонує найбільшу обчислювальну потужність. Як результат, його найкраще використовувати для більш інтенсивних процесорів, де Arduino Uno буде важко забезпечити обробку даних та візуалізацію.



Рисунок 4.81 – зовнішній вигляд Raspberry Pi Zero

STM32 – відладочна плата, яка заснована на базі мікроконтролера STM32F103C8T6 (рис 4.83). Її робоча частота – 72 МГц (на відміну від 16 МГц у Arduino), обсяг флеш-пам'яті 64 Кб (в два рази більше). RAM у STM32 становить цілих 20 Кб. При цьому плата оснащена 12-бітним АЦП і 16-бітним апаратним ШІМ.

В порівнянні з Ардуіно можна виділити наступні недоліки:

- високий поріг входу, для нормального користування необхідний хороший базис;
- бібліотеки також присутні, але більшість з них застаріли або ж зроблені кожним особисто для себе, швидше за все доведеться писати свою власну;
- складні семантичні особливості мови програмування.

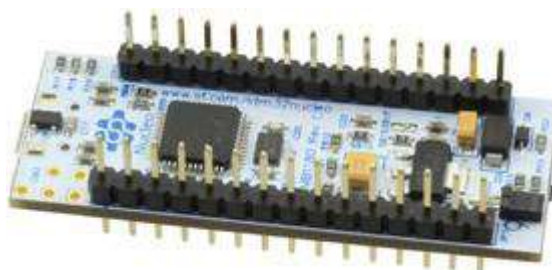


Рисунок 4.82 – Зовнішній вигляд STM32L4

Arduino Uno – це широко використовувана плата мікроконтролерів з відкритим кодом на базі мікроконтролера ATmega328P (рис. 4.83). У його склад

входить все необхідне для зручної роботи з мікроконтролером: 14 цифрових входів/виходів (з них 6 можуть використовуватися в якості ШІМ-виходів), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм для програмування всередині схеми (ICSP) і кнопка скидання. Для початку роботи з пристроєм досить просто подати живлення від AC/DC-адаптера або батарейки, або підключити його до комп'ютера за допомогою USB-кабелю. До переваг Arduino Uno можна віднести наступне:

- проста і зручна середовище програмування. оболонка програми є легкою в застосуванні для початківців програмістів, однак має достатню гнучкість для роботи професіоналів;
- можливість функціонування на різних видах операційних систем;
- невисока вартість;
- невеликі розміри плати.

Саме ці переваги виявилися найважливішими, тому в своїй роботі використано Arduino Uno.



Рисунок 4.83 – Зовнішній вигляд Arduino UNO

4.13.3 Вибір програмного забезпечення для керування мобільною платформою

Programino – платне середовище розробки, але його можна випробувати протягом 14-ти днів безкоштовно. Programino, як і інші середовища розробки, вимагає, однак, щоб на комп'ютері була встановлена Arduino IDE. Мова, яка використовується в даному середовищі розробки - така же, як і в оригінальній Arduino IDE – C. Дана IDE пропонує такий зручний спосіб швидкої розробки як автодоповнення коду. В даному середовищі розробки є кілька додаткових корисних інструментів, наприклад: блокнот, дизайнер LCD символів,

перетворювач між DEC-BIN-HEX, термінал послідовного порту, аналоговий плоттер.

B4R (Basic for Arduino) унікально тим, що використовує мову Basic, а не C. Воно також підтримує функцію автодоповнення коду. Крім того, воно повністю безкоштовна. Це середовищі розробки використовує іншу, більш об'єктно-орієнтована мова, ніж класична Arduino IDE, з іншим синтаксисом.

Оскільки буде перевагою використання мови C, а не Basic, то будемо використовувати Arduino IDE.

Саме Arduino IDE буде використовуватися для розробки та реалізації програмного коду мікроконтролера (рис. 4.85). Це середовище розробки було обрано через його великий функціонал і набір бібліотек для написання і налагодження програм.

Мова програмування Arduino є стандартним C ++ (використовується компілятор AVR-GCC) з деякими особливостями.

Програми, написані на Arduino, називаються начерки (або іноді sketch) і зберігаються в файлах з розширенням `.ino`. Ці файли перед компіляцією обробляються препроцесором Ардуіно. Також існує можливість створювати і підключати до проекту стандартні файли C ++.

Обов'язкову в C ++ функцію `main` препроцесор Arduino створює сам, вставляючи туди необхідні «чорнові» дії.

Далі необхідно написати дві обов'язкові для Arduino функції: `setup` і `loop`. Перша викликається одноразово при старті, друга виконується в нескінченному циклі.

В текст своєї програми (скетчу) програміст не зобов'язаний вставляти заголовки використовуваних стандартних бібліотек. Ці заголовки додають препроцесор Arduino відповідно до конфігурації проекту. Однак призначені для користувача бібліотеки потрібно вказувати.

Менеджер проекту Arduino IDE має нестандартний механізм додавання бібліотек. Бібліотеки у вигляді вихідних текстів на стандартному C ++ додаються в спеціальну папку в робочому каталозі IDE. При цьому назва бібліотеки буде додано до списку бібліотек в меню IDE. Програміст зазначає потрібні бібліотеки, і вони вносяться до списку компіляції.

Arduino IDE не пропонує ніяких налаштувань компілятора і мінімізує інші настройки, що спрощує початок роботи для новачків і зменшує ризик виникнення проблем.

Найпростіша Arduino-програма складається з двох функцій:

– `setup`-функція викликається одноразово при старті мікроконтролера;

Ця IDE найзручніша з усіх доступних, так як є безкоштовною, в ній можна писати на мові C, а також ця IDE не перевантажена різними непотрібними функціями. Також до її переваг можна віднести:

- підсвічування коду;
- швидка заливка скетчу в плату arduino;
- можна швидко присвоїти номер порту до певної змінної;
- так само можна швидко позначити стан порту (low або high).

4.14 Розробка апаратного забезпечення мобільної платформи

4.14.1 Привідна система коліс

Щоб зібрати мобільний робот, знадобляться компоненти, описані нижче. Для управління рухом мобільного робота використовуються 4 мотор-редуктора (рис 4.86). Управління моторами здійснюється незалежно один від одного. Характеристики мотора наведені у таблиці 4.4. Розміри наведені на рисунку 4.85.



Рисунок 4.85 – Мотор-редуктор DC-6v

Мікро-електродвигун постійного струму встановлений в пластмасовий корпус, в якому знаходиться понижуючий швидкість обертання редуктор з пластмасових шестерень і збільшуючий зусилля на валу механізму. На вал мотор-редуктора насаджується колесо з гумовою покришкою. Вал знаходиться з двох сторін корпусу редуктора. Колесо може встановлюватися з будь-якого боку корпусу. На другу частину валу-редуктора встановлюють диск з отворами, що дозволяє працювати оптичному датчику контролю параметрів обертання валу.

Таблиця 4.4 – Характеристика мотора постійного струму РП

Характеристики	Значення
Робоча напруга	3-12В
Рекомендована напруга	6-8В
Передавальне число редуктора (без навантаження при 3В)	1:48
Номінальний струм	70мА
Максимальний струм (при 6В)	250мА
Швидкість обертання (при 6В)	90 об/хв
Обертальний момент	2кг/см
Шум	65 dB

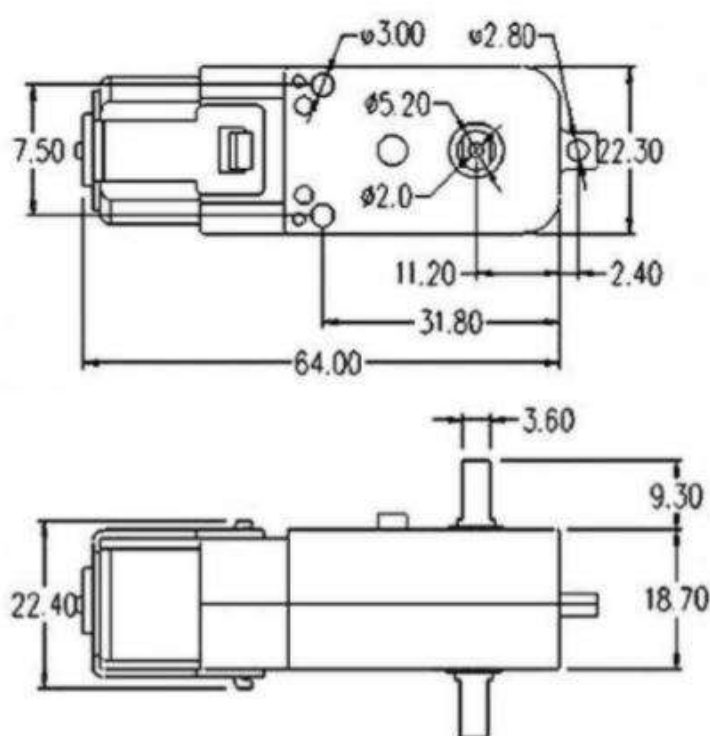


Рисунок 4.86 – Розміри мотор-редуктора DC-6v

4.14.2 Сервопривод SG90

Привод – електромотор з редуктором. Щоб перетворити електрику в механічний поворот, необхідний електромотор. Однак найчастіше швидкість обертання мотора буває занадто великою для практичного використання. Для зниження швидкості використовується редуктор: механізм з шестерень, який передає і перетворює крутний момент (рис 4.87).

Вмикаючи і вимикаючи електромотор, можна обертати вихідний вал – кінцеву шестерню сервоприводу, до якої можна прикріпити щось, чим ми хочемо

керувати. Однак, для того щоб положення контролювалося пристроєм, необхідний датчик зворотного зв'язку – енкодер, який буде перетворювати кут повороту назад в електричний сигнал. Для цього часто використовується потенціометр. При повороті бігунка потенціометра відбувається зміна його опору, пропорційне куту повороту. Таким чином, з його допомогою можна встановити поточний стан механізму.

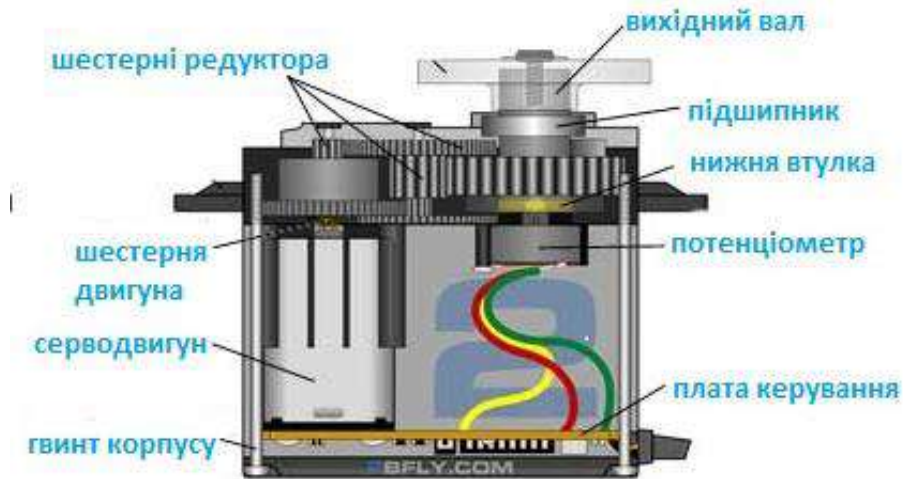


Рисунок 4.87 – Пристрій сервоприводу

Крім електромотора, редуктора і потенціометра, сервопривод має електронну частину, що відповідає за прийом зовнішніх параметрів, зчитування даних потенціометра, їх порівняння, включення/вимикання двигуна. Вона відповідає за підтримання негативного зворотного зв'язку. Сервопривод SG90 представлено на рис. 4.88, 4.89, характеристики – в таблиці 4.5.



Рисунок 4.88 – Сервопривод SG90

Таблиця 4.5 – Характеристики сервоприводу SG90

Характеристики	Значення
Робоча напруга	4.8-6В
Матеріал редуктора	Пластмасовий
Маса	9 гр.
Номінальний струм	12 – 100 мА
Пусковий струм	До 1А
Швидкість обертання (при 6В)	60°/0.08s
Обертальний момент (при 4.8В)	1.8 кг/см
Обертальний момент (при 6В)	2.2 кг/см

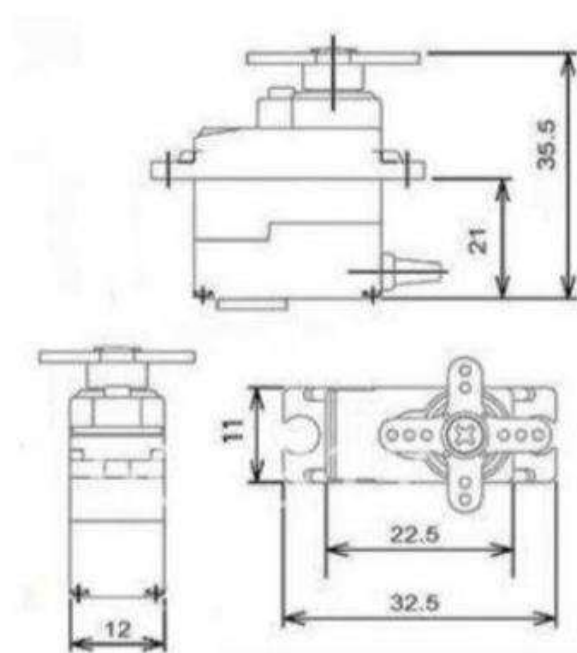


Рисунок 4.89 – Розміри сервопривода SG90

4.14.3 Реверсивний драйвер двигуна L298N

Мікросхема включає в себе драйвер двигуна для управління слабкострумними моторами. Мікросхема драйвера двигуна зображена на рисунку 4.90.

До переваг даної мікросхеми відноситься роздільне живлення логічної частини мікросхеми, напруга живлення якої лежить в межах 4.5-5 вольт (VSS), і силовий частини живлення двигунів (VS).

За допомогою даної мікросхеми, ми можемо керувати двигунами з широким діапазоном напруги живлення від 4.5 до 46 вольт. Також L298N видає

2А тривалого струму навантаження на кожен канал. Максимальний струм короткочасно може досягти 4А.

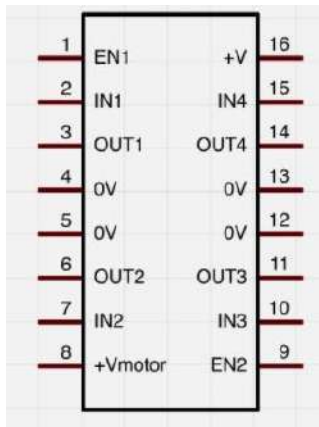


Рисунок 4.90 – Мікросхема L298N

Дана мікросхема невибаглива до напруги вхідних сигналів, що подаються на виходи INPUT. Логічний “0” розпізнається мікросхемою, коли вхідна напруга <1.5 вольт. Логічна “1” з’являється при вхідній напрузі, що лежить в межах від 2.3 до 7 вольт. Діапазон робочих температур: від -40°C до +150°C.

4.14.4 Arduino Uno R3

Arduino Uno – це пристрій на основі мікроконтролера ATmega328. У його склад входить все необхідне для зручної роботи з мікро контролером: 6 аналогових входів, кварцовий резонатор на 16 МГц, роз’єм USB, роз’єм живлення, роз’єм для внутрисхемного програмування (ICSP) і кнопка скидання (рис 4.92). Характеристики плати наведені у таблиці 4.6, а її распиновка на рисунку 4.92.

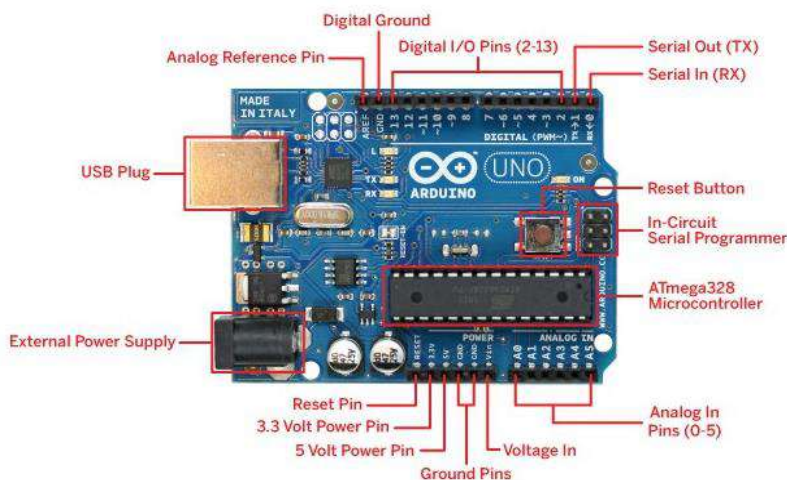


Рисунок 4.92 – Елементи плати Arduino

Таблиця 4.6 – Характеристики Arduino Uno R3

Мікроконтролер	АТmega328
Робоча напруга	5В
Напруга живлення (рекомендована)	7-12В
Напруга живлення (гранична)	6-20В
Цифрові входи / виходи	14 (з них 6 можуть використовуватися в якості ШІМ-виходів)
Аналогові входи	6
Flash-пам'ять	32 КБ (АТmega328) з яких 0.5 КБ використовуються загрузчиком
SRAM	2 КБ (АТmega328)
EEPROM-пам'ять	1 КБ (АТmega328)
Тактова частота	16 МГц

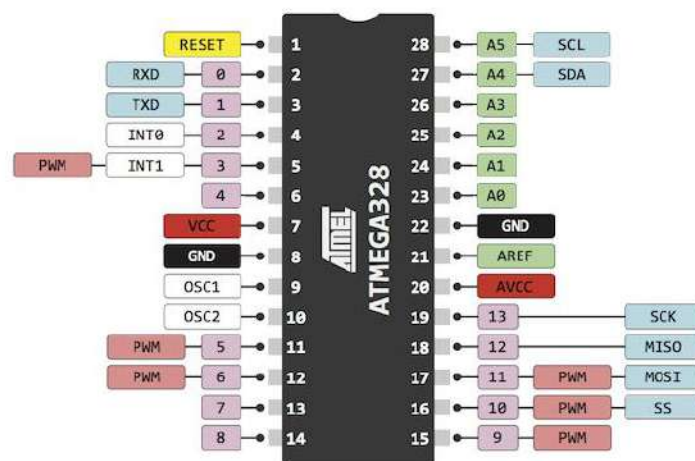


Рисунок 4.93 – Розпіновка Atmega328

4.14.5 Ультразвуковий датчик відстані HC-SR04

Ультразвуковий далекомір HC-SR04 – це приймач і передавач ультразвукового сигналу (рис. 4.94). Передавач генерує сигнал, який, відбитий перешкодою, досягає приймача. Вимірявши час, за який сигнал проходить до об'єкта і назад, можна оцінити відстань. Його технічні параметри наведені у таблиці 4.7.

Таблиця 4.7 – технічні параметри ультразвукового далекоміра

Напруга живлення	5В
Робочий параметр сили струму	15 мА
Сила струму в пасивному стані	< 2 мА
Оглядний кут	15°
Сенсорне розширення	0,3 см
Вимірювальний кут	30°
Ширина імпульсу	10 ⁻⁶ с



Рисунок 4.94 – Датчик відстані HC-SR04

4.14.6 Зовнішній вид розробленої мобільної платформи

У розділі були описані елементи, які використовувалися при виготовленні мобільного робота. На рис. 4.95 зображений зібраний мобільний робот. На рис. 4.96 зображена схема компонентів, з яких складається мобільний робот, та їх підключення.

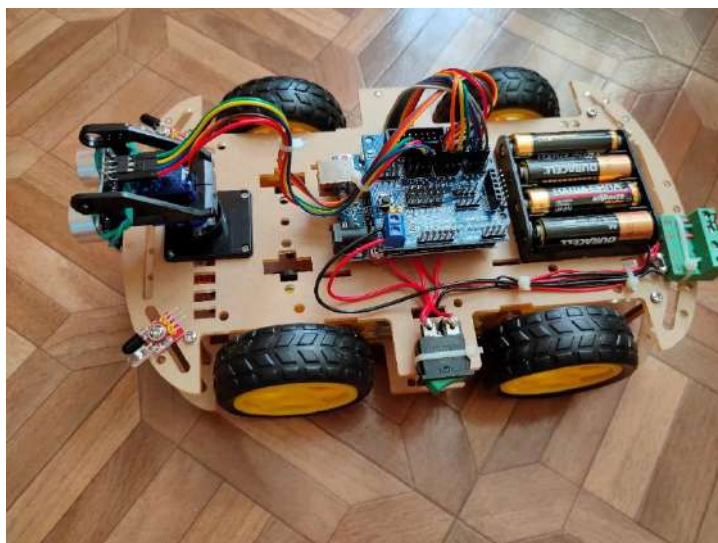


Рисунок 4.95 – Мобільний Arduino-робот

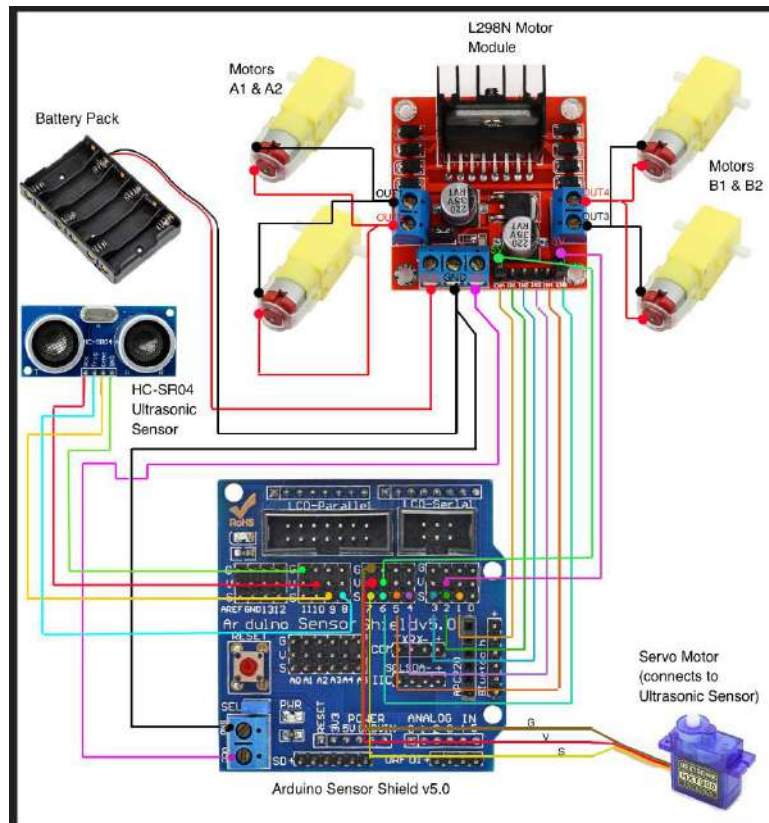


Рисунок 4.96 – схема компонентів мобільної платформи

4.15 Розробка програмного забезпечення для мобільної платформи

4.15.1 Розробка алгоритмічного забезпечення

Система управління роботом призначена для забезпечення спільної роботи всіх підсистем і компонентів робота з метою переміщення робота та виконання руху по необхідній траєкторії. Робот який являє собою складну систему, яка включає в себе механічні конструкції різних складностей, таких як, мікроконтролеру, керуючого пристрою (мікропроцесора), та різних сенсорів та датчиків для вимірювання та поучення інформації. Основним завданням управління роботом полягає у тому, щоб сформувані керуючі впливи для виконавчих приводів на кожному ступені рухливості робота. Всього існує три рівня управління роботом: виконавчий, тактичний та стратегічний рівні управління (рис 4.97).

Програмне забезпечення було розроблено у середі програмування Arduino IDE та призначено для роботи з інтерфейсом мобільного робота.

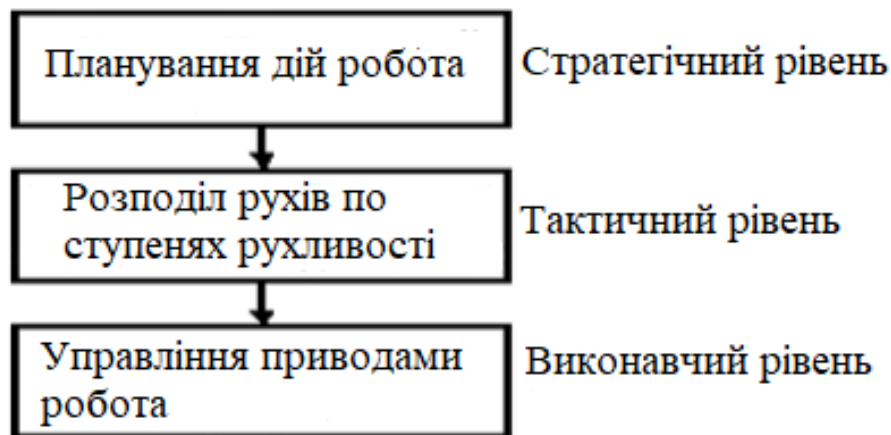


Рисунок 4.97 – Рівні керування роботом

В ході написання програмного забезпечення були розроблені такі функції програми, як:

- функція переміщення мобільного робота по прямій траєкторії до тих пор, поки ультразвуковий датчик відстані не побачить перед собою перешкоду;
- функція управління сервоприводом;
- функція, яка аналізує, в якій стороні від робота більше всього вільного місця;
- функція повороту робота на задану кількість градусів, в сторону, в якій найбільша кількість вільного місця.

Алгоритм переміщення Arduino-робота має наступний вигляд:

При вмиканні робота кнопкою, включається енергопостачання до плати мікроконтролеру Arduino UNO та вона вмикається, починаючи при цьому виконувати вшитий в її пам'ять код програми. Спочатку робот починає рух по прямій траєкторії, поки встановлений в ньому ультразвуковий датчик відстані не помітить перед собою перешкоду. Після такого, як перешкоду буде виявлено, робот почне знижувати швидкість, повільно під'їжджаючи до неї. Під'їхавши на відстань 60 сантиметрів, робот почне обертати сервоприводом на який встановлений датчик відстані, щоб проаналізувати і зрозуміти в якій стороні більше доступного вільного простору.

Проаналізувавши і обробивши ці дані, робот вибере ту сторону, в якому вільного простору буде більше і прийме рішення рухатися саме в цей бік. Даний цикл буде продовжувати своє виконання поки на плату Arduino буде подаватися напруга, або поки робот не відключити вручну. Алгоритм роботи програми представлено на рисунку 4.98.



Рисунок 4.98 – Алгоритм роботи програми

4.15.2 Порядок розробки програмного забезпечення

В даному розділі програми задаються піни для роботи з ультразвуковим датчиком а також відбувається основна настройка плати Arduino UNO. Також тут відбувається оголошення трьох змінних типу long, необхідних для вимірювання відстані до перешкоди по центру, а також відстані до перешкод зліва і справа (рис. 4.99).

```

#define PIN_TRIG 10//пин для работы с уз датчиком
#define PIN_ECHO 11//пин для работы с уз датчиком
#define enA 5// синий
//пины для работы с платой двигателя
#define enB 9//желтый
#define in1 6// зеленый
#define in2 8//коричневый
#define in3 13//оранжевый
#define in4 12//красный

long cmleft, cmright, cmcentr;

void setup()
{
  Serial.begin (19200);
  pinMode(PIN_TRIG, OUTPUT);
  pinMode(PIN_ECHO, INPUT);

  servo.attach(7);

  pinMode(enA, OUTPUT);
  pinMode(enB, OUTPUT);
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
  analogWrite(enA, 255);
  analogWrite(enB, 255);

  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
}

```

Рисунок 4.99 – Основна настройка програми для Arduino UNO

Далі необхідно написати функцію для визначення відстані в сантиметрах до перешкоди. У змінну *a* ми будемо записувати час затримки акустичного сигналу на ехолотаторі. У змінну *b* ми будемо записувати результат перетворення часу в відстань (рис. 4.100).

```

long rastoyanie (void)
{long a;
long b;
digitalWrite (PIN_TRIG, LOW);
delayMicroseconds (5);
digitalWrite (PIN_TRIG, HIGH);

delayMicroseconds (10);
digitalWrite (PIN_TRIG, LOW);

a = pulseIn (PIN_ECHO, HIGH);

b = (a / 2) / 29.1;

delay (250);
return b;}

```

Рисунок 4.100 – Функція визначення відстані до перешкоди

Далі було реалізовано функцію повороту сервоприводом вліво і вправо, для аналізу і визначення відстані до перешкод з різних сторін. В якій стороні перешкода буде знаходитися далі від датчика, туди і поїде робот (рис 4.101).

```

servo.write (140);
cmright=rastoyanie ();
Serial.print ("Расстояние до объекта справа");
Serial.print (cmright);
Serial.println (" см.");

servo.write (40);
cmleft=rastoyanie ();
Serial.print ("Расстояние до объекта по слева ");
Serial.print (cmleft);
Serial.println (" см.");
servo.write (115);

if (cmright >= cmleft)

{analogWrite (enA, 255);
analogWrite (enB, 0);
delay (2000);
analogWrite (enB, 255);

}
else
{analogWrite (enA, 0);
analogWrite (enB, 255);
delay (2000);
analogWrite (enA, 255);}}

```

Рисунок 4.101 – Вибір подальшого напрямку руху

Робот буде продовжувати рух по прямій, поки не побачить перешкоду на відстані 70 сантиметрів або ближче до себе (рис 4.102).

```
if(cmcentr>70)
{
  digitalWrite(enA, 255);
  digitalWrite(enB, 255);

  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
}
```

Рисунок 4.102 – Функція руху робота по прямій траєкторії

4.15.3 Тестування ПЗ

Для проведення коректного тестування роботи мобільної системи потрібно використовувати певну методику тестування. Потрібно ввести деякі обмеження як до об'єктів, розташованих в просторі проведення тестування так і до місця проведення тестування в цілому.

Конструкція даного мобільного робота не призначена для переміщення по пересіченій місцевості, відповідно, в приміщенні, в якому буде проводитися тестування поверхня повинна бути рівною і гладкою, без маленьких ям та інших невеликих перешкод, здатних поміщати роботу робота. Описані вище перешкоди здатні змінити траєкторію руху робота, і нашкодити роботі програми в цілому.

Для правильної роботи ультразвукового датчика відстані необхідно внести наступні обмеження до ймовірних перешкод в просторі:

- об'єкти не повинні бути нижче по висоті, ніж висота, на якій закріплений датчик;
- об'єкти повинні дорівнювати 13 сантиметрам або більше у висоту;
- об'єкти не повинні бути занадто тонкими, тому що датчик може некоректно зафіксувати відстань до такого об'єкта;
- мінімальна ширина об'єкта повинна складати 6 см;
- об'єкти не бути занадто пористими або м'якими, а також не повинні мати структуру яка може поглинути звукову хвилю.

З цього слід, що об'єкт, який буде використовуватися в якості перешкоди, має бути твердим, в висоту мати 13 сантиметрів або більше, а також бути шириною більше ніж 6 сантиметрів.

Встановимо робота в будь-яку частину простору, де немає перешкод, на відстані до 70 сантиметрів або менше. На рисунку 4.103 показано початкове положення робота в деякому просторі з перешкодами.



Рисунок 4.103 – Стартова позиція робота в просторі з перешкодами.

На рисунку 4.104 видно, що робот продовжує свій рух по прямій, поки не наблизитися до найближчої перешкоди на 60 сантиметрів.



Рисунок 4.104 – Робот наблизився до перешкоди на 60 сантиметрів

Наблизившись до перешкоди на відстань 60 сантиметрів, робот ненадовго зупинився і провів аналіз, в який бік йому рухатися далі. Відповідно до алгоритму він обрав сторону, де більше вільного місця, і продовжив свій рух вже в цю сторону. Це показано на рисунку 4.105.



Рисунок 4.105 – Продовження руху згідно з алгоритмом

За результатами тестування можна сказати що мобільна система готова до експлуатації і реалізує всі функціональні можливості при дотриманні всіх встановлених обмежень.

4.16 Типові закони управління та регулятори

Під законом управління розуміють алгоритм, згідно з яким пристрій управління (регулятор) формує вплив, який подається на вхід об'єкта управління.

Законом управління називається функціональна залежність вихідної величини пристрою управління від його вхідної величини, складена без урахування динамічних запізнень елементів пристрою управління.

На рис. 4.106 подано класичну структурну схему системи управління з одиничним негативним зворотним зв'язком.

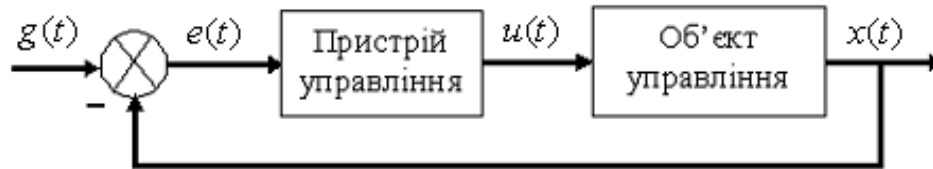


Рисунок 4.106 – Структурна схема САУ

Задача синтезу системи автоматичного управління ставиться таким чином: при заданому об'єкті управління потрібно синтезувати (побудувати) пристрій управління, при якому система задовольняє заданим вимогам до її якості. У такій постановці задачі структура ПУ (регулятора) і всієї системи задана, необхідно, виходячи із заданих вимог до системи, визначити параметри ПУ і якщо треба параметри коригуючої ланки.

Вид передатної функції пристрою управління визначає закон управління. На сьогодні у промисловості розрізняють чотири основні закони керування: пропорційний, інтегральний, пропорційно-інтегральний, пропорційно-диференціальний, пропорційно-інтегро-диференціальний.

Пропорційний закон управління (П закон) визначається наступним рівнянням пристрою управління:

$$u(t) = k \cdot e(t). \quad (4.1)$$

Передавальна функція пристрою управління при П законі управління дорівнює:

$$W_p(s) = k. \quad (4.2)$$

Регулятор з таким законом управління називається пропорційним регулятором або П-регулятором.

При інтегральному законі управління (І-закон) пристрій управління виробляє сигнал, пропорційний інтегралу від помилки:

$$u(t) = k \int_0^t e(t) dt. \quad (4.3)$$

Передавальна функція пристрою управління при І-законі управління дорівнює:

$$W_p(s) = \frac{k}{s}. \quad (4.4)$$

Регулятор з таким законом управління називається інтегральним регулятором або І-регулятором.

В порівнянні з П-законом І-закон управління забезпечує астатизм системи, проте динамічні властивості системи з І-законом управління звичайно гірше, ніж у системи з П-законом. Введення інтеграла в закон управління, як правило, підвищує коливальність системи і у деяких випадках може зробити систему нестійкою, якщо не вживати спеціальних заходів.

При пропорційно-інтегральному законі управління (ПІ-закон) пристрій управління виробляє суму двох сигналів: пропорційного помилці і пропорційного інтегралу від помилки:

$$u(t) = K_1 e(t) + K_2 \int_0^t e(t) dt. \quad (4.5)$$

Передавальна функція пристрою управління при ПІ законі управління дорівнює:

$$W_p(s) = K_1 + \frac{K_2}{s}. \quad (4.6)$$

Регулятор з таким законом управління називається пропорційно-інтегральним регулятором або ПІ-регулятором.

Динамічні властивості у системи з ПІ-законом управління краще, ніж у системи з І-законом (за інших рівних умов). За своїми властивостями пропорційно-інтегральна система у перехідному режимі наближається до системи із пропорційним управлінням, а у сталому режимі подібна до системи з інтегральним управлінням.

При пропорційно-диференціальному законі управління (ПД закон) пристрій управління виробляє суму двох сигналів: пропорційного помилці і пропорційного похідної від помилки:

$$u(t) = K_1 e(t) + K_2 \frac{de(t)}{dt}. \quad (4.7)$$

Передавальна функція пристрою управління при ПД законі управління дорівнює:

$$W_y(s) = K_1 + K_2 s. \quad (4.8)$$

Регулятор з таким законом управління називається пропорційно-диференціальним регулятором або ПД-регулятором.

Введення у закон управління похідної від похибки збільшує швидкість реакції системи на зміну вхідного впливу, підвищує її швидкодію, при цьому зменшується похибка системи у динамічному режимі, покращуються її динамічні властивості.

При пропорційно-інтегро-диференціальному законі управління (ПІД закон) пристрій управління виробляє сигнал, який дорівнює сумі трьох складових: пропорційного помилці, пропорційного інтегралу від помилки і пропорційного похідної від помилки.

$$u(t) = K_1 e(t) + K_2 \int_0^t e(t) dt + K_3 \frac{de(t)}{dt}. \quad (4.9)$$

Передавальна функція пристрою управління при ПІД законі управління дорівнює:

$$W_y(s) = K_1 + \frac{K_2}{s} + K_3 s. \quad (4.10)$$

Регулятор с таким законом управління називається пропорційно-інтегро-диференціальним регулятором або ПІД-регулятором

Введення в закон управління інтегральної складової робить систему астатичною та покращує якість системи в сталому режимі, але оказує дестабілізуючий вплив (тобто може зробити систему нестійкою) та погіршує якість системи в перехідному режимі.

Введення в закон управління диференціальної складової оказує стабілізуючий вплив (тобто може зробити нестійку систему стійкою) та покращує якість системи в перехідному режимі, не оказуючи впливу на якість системи в сталому режимі.

Визначимо, при яких типових законах управління статична помилка системи (рис. 4.107) буде дорівнювати нулю, якщо задана передатна функція об'єкта:

$$W_0(s) = \frac{1}{s + 1}.$$

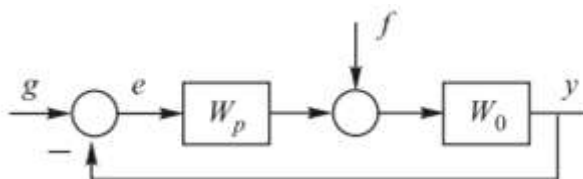


Рисунок 4.107– Структурна схема системи

Статична помилка буде дорівнювати нулю, якщо система буде астатичною щодо задаючого впливу і обурення. А для цього потрібно, щоб регулятор містив інтегруючу ланку. Тому шуканими законами управління будуть пропорційно-інтегральний закон і пропорційно-інтегро-диференціальний закон.

Визначимо, при яких типових законах управління помилка від задаючого впливу системи (рис. 4. 107) буде дорівнювати нулю за умови, якщо

$$W_0(s) = \frac{1}{s^2(s+1)}, g(t) = at \text{ або } f(t) = b.$$

Оскільки помилка від задаючого впливу і обурення мають вигляд:

$$e_{g\infty}(t) = C_{g0}at + C_{g1}a, \quad e_{f\infty} = C_{f0}b,$$

то помилка буде дорівнювати нулю, якщо $C_{g0} = C_{g1} = 0$ та $C_{f0} = 0$. Отже, система повинна бути астатичною з астатизмом 2-го порядку щодо задаючого впливу і з астатизмом 1-го порядку щодо обурення. Так як об'єкт включає дві послідовно з'єднаних інтегруючих ланки, система буде астатичною з астатизмом не менше 2-го порядку щодо задаючого впливу при будь-якому типовому законі управління. Однак вона буде астатичною щодо обурення тільки при ПІ-законі і ПІД-законі. При ПІ-законі передатна функція розімкнутої системи:

$$W(s) = W_p(s)W_0(s) = \frac{k_p p + k_{\Pi}}{s^3(s+1)},$$

а характеристичне рівняння має вигляд:

$$\lambda^4 + \lambda^3 + k_{\Pi}\lambda + k_{\text{и}} = 0.$$

У цьому рівнянні коефіцієнт при λ^2 дорівнює нулю і необхідна умова стійкості не виконується. Тому система при ПІ-законі структурно нестійка.

При ПІД-законі передатна функція розімкнутої системи:

$$W(s) = W_p(s)W_0(s) = \frac{k_{\text{д}}s^2 + k_{\Pi}s + k_{\text{и}}}{s^3(s+1)}$$

та характеристичне рівняння має вигляд:

$$\lambda^4 + \lambda^3 + k_d \lambda^2 + k_p \lambda + k_i = 0.$$

Визначник Гурвіца 3-го порядку:

$$\Delta_3 = \begin{vmatrix} 1 & k_p & 0 \\ 1 & k_d & k_i \\ 0 & 1 & k_p \end{vmatrix} = k_p k_d - k_i - k_p^2.$$

Відповідним вибором параметрів регулятора можна зробити наступне, при ПД-законі система структурно стійка і шуканим законом управління є ПД-закон.

Розрахуємо оптимальні параметри ПД-регулятора з використанням Matlab.

Побудуємо Simulink-модель САУ відповідно до заданої схеми, наприклад як на рис. 4.108, помістивши у вікно моделі блоки Step (з бібліотеки Sources), Sum (з бібліотеки Math Operations), Transfer Fcn (з бібліотеки Continuous), PID Controller (з бібліотеки Simulink Extras/Additional Linear), Signal Constraint (з бібліотеки Simulink Resonse Optimization).

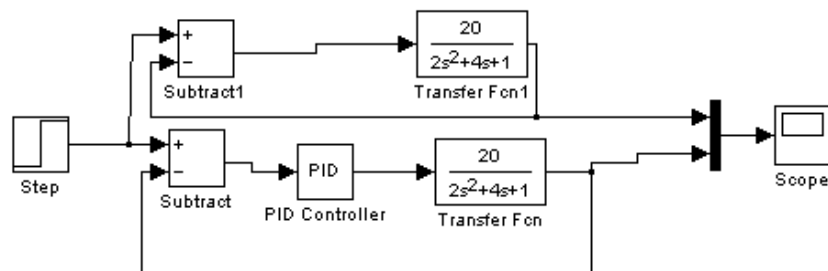


Рисунок 4. 108– Simulink-модель САУ

Змінюючи параметри ПД-регулятора (рис. 4. 109), можна отримати П, ПІ, І та ПД закони управління за перехідною характеристикою системи. Для цього необхідно задавати значення коефіцієнтів підсилення пропорційної, інтегруючої та диференціюючої складової відповідно до варіанта реалізації.

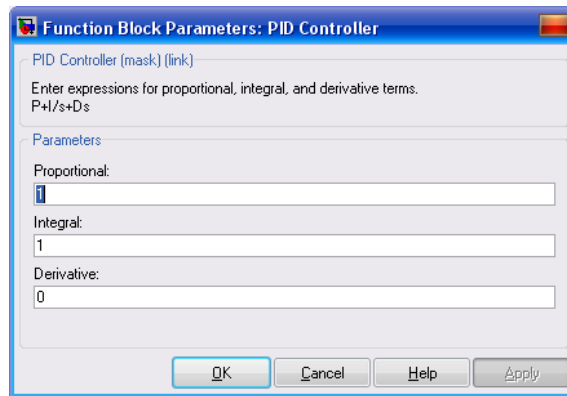


Рисунок 4. 109– Налаштування блоку PID Controller для дослідження законів регулювання

Для прикладу дослідження оптимізації ПІД-регулятора розглянемо схему на рис. 4.110.

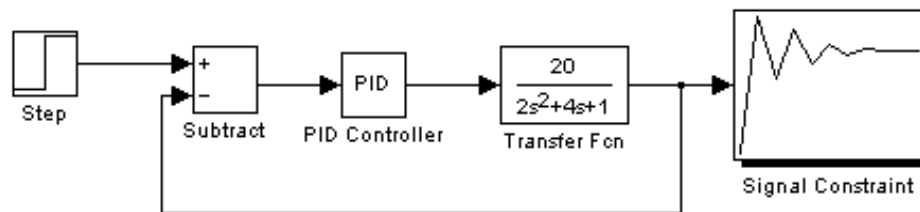


Рисунок 4. 110– Simulink-модель САУ для оптимізації ПІД-регулятора

Для ініціалізації у командному вікні Matlab змінних $K_p=1$; $K_i=1$; $K_d=0$, необхідно перейти до командної строки Matlab та ввести команду:

`>> Kp=1; Ki=0; Kd=1.`

Налаштування параметрів блоку PID Controller (рис. 4. 111), здійснюється вводом у поле параметра Proportional змінної K_p , у полі Integral – K_i , а у полі Derivative – K_d .

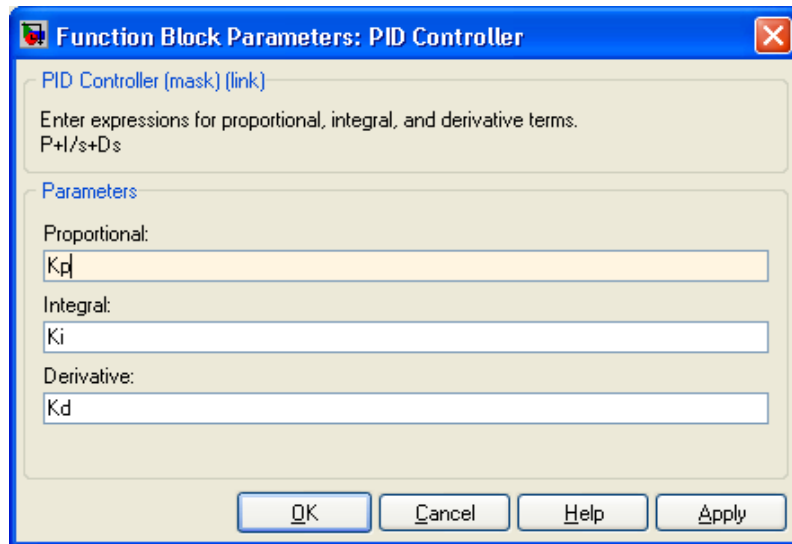


Рисунок 4.111 – Налаштування блоку PID Controller

Після подвійного натискання лівої клавіші миші на блоці Signal Constraint відкривається вікно блоку параметрів (рис. 4.112).

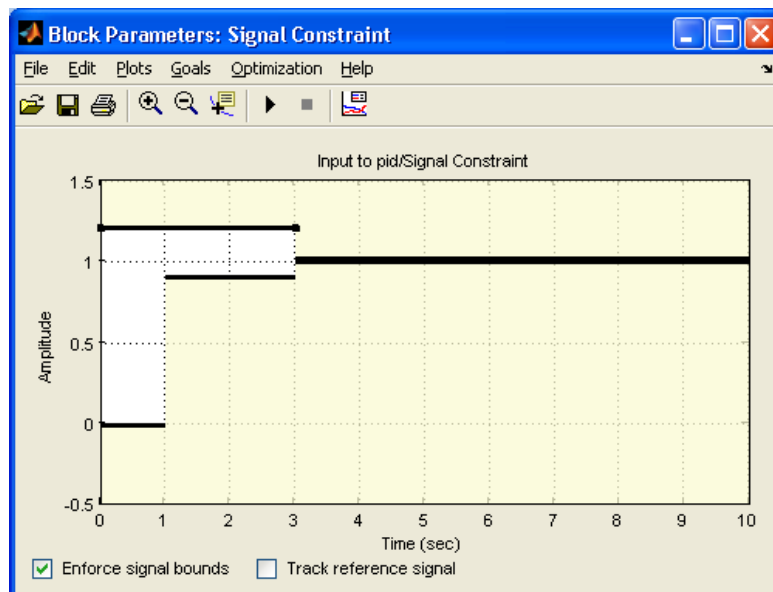


Рисунок 4.112 – Налаштування параметрів системи

У вікні налаштування параметрів можна задати границі для побудови перехідної характеристики досліджуваної системи із заданими прямими показниками якості. Для цього встановлюється коридор, в межах якого має знаходитися вхідний сигнал блоку Signal Constraint відповідно до вимог технічного завдання. Це можна зробити, пересуваючи за допомогою миші чорні лінії. Натиснути кнопку «Add...» (рис. 4.113), у вікні вибору параметрів, що

з'явилося, потрібно виділити відповідну змінну та натиснути кнопку Ок. Таким чином у якості параметрів оптимізації вказати всі змінні ПІД-регулятора.

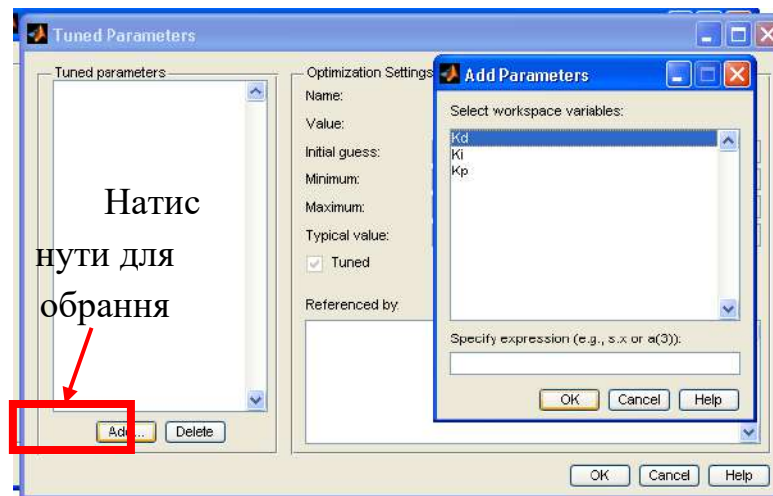


Рисунок 4.113 – Процес вибору параметрів для оптимізації ПІД-регулятора

Для налаштування параметрів розрахунку необхідно обрати пункт «Optimization/Simulation Options...» та у вкладці «Optimization Options» вказати кількість ітерацій для розрахунку рівною 3 (рис. 4.114).

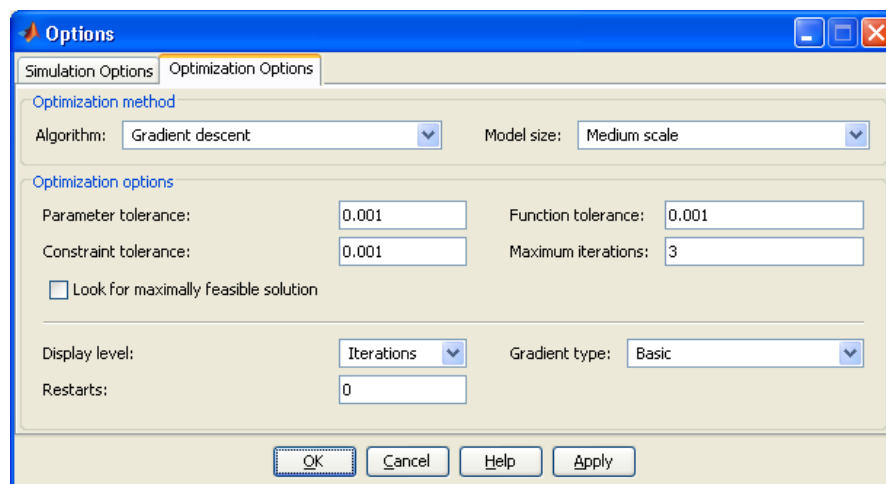


Рисунок 4.114 – Налаштування параметрів розрахунку оптимальних параметрів регулятора

Результатами проведеної оптимізації будуть обчислені оптимальні параметри ПІД-регулятора (рис. 4.115) та графіки розрахованих перехідних характеристик (рис. 4.116).

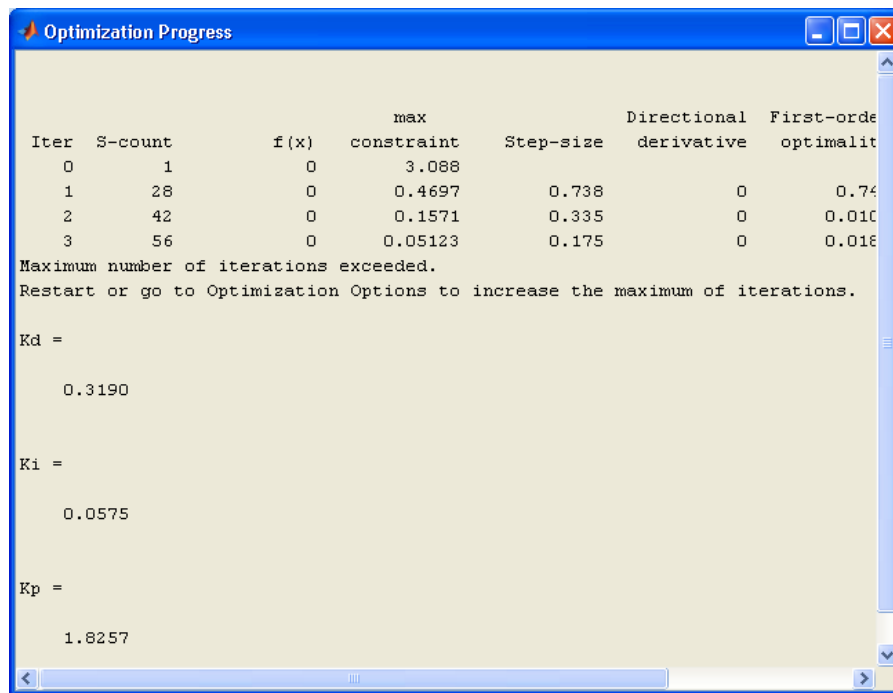


Рисунок 4.115 – Результати оптимізації параметрів ПІД-регулятора

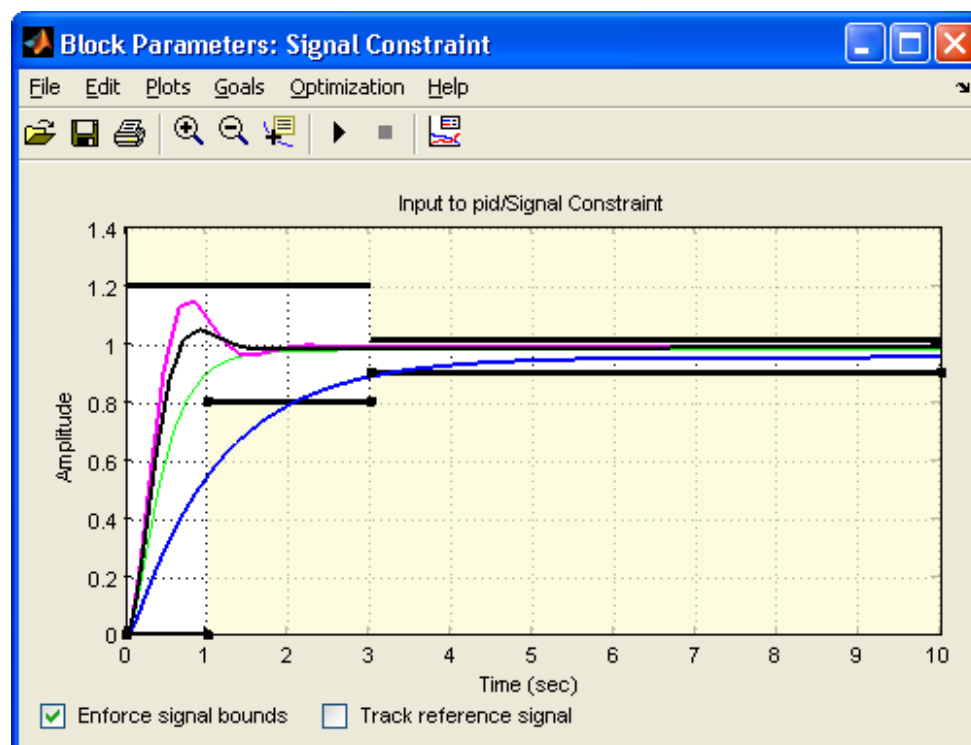


Рисунок 4.116 – Результати ітераційного розрахунку оптимальних параметрів регулятора

При використанні Matlab версії вище 2012 дещо змінився інтерфейс і методика підбору оптимальних параметрів ПІД регулятора, але суть залишилася та ж – необхідно підключити блок в Simulink-моделі (рис. 4.117) послідовно

перед об'єктом управління в пряму ланцюга зворотного зв'язку додати ланку ПІД-регулятора.

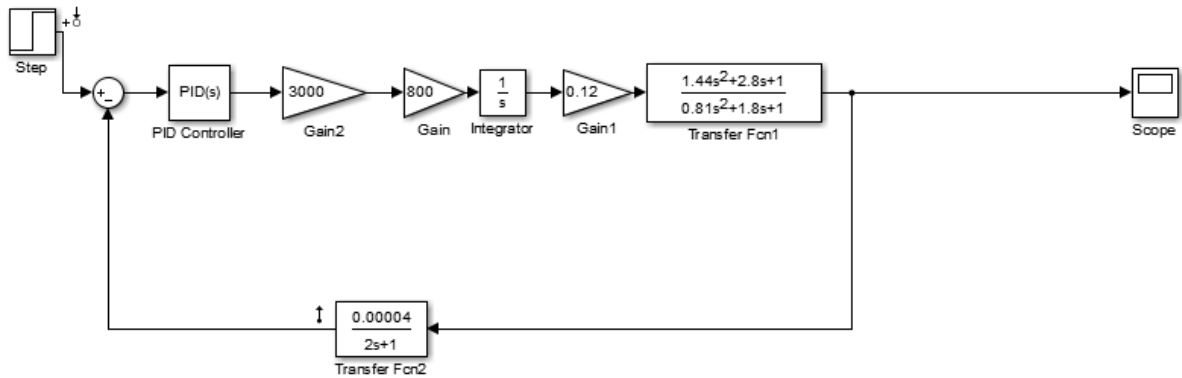


Рисунок 4.117 – Модель з ПІД-регулятором

Далі відкрити подвійним натисканням блок ПІД-регулятора, де задати параметри для моделювання (рис. 4.118) і натиснути кнопку Tune... Результат роботи PID-Tuner подано на рис. 4.119.

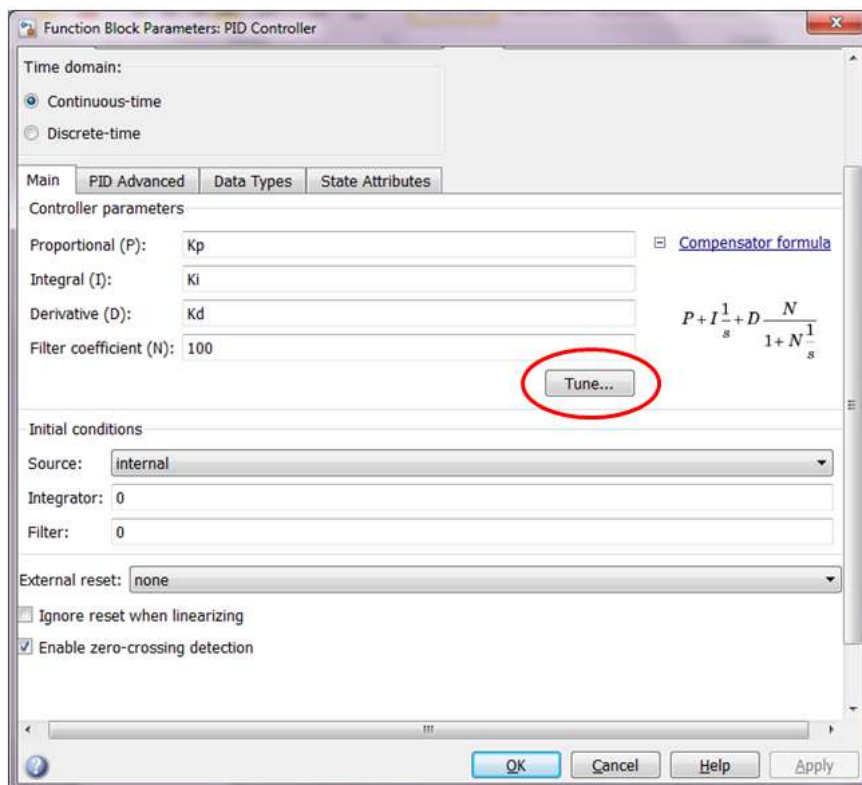


Рисунок 4.118 – Налаштування блоку ПІД-регулятора

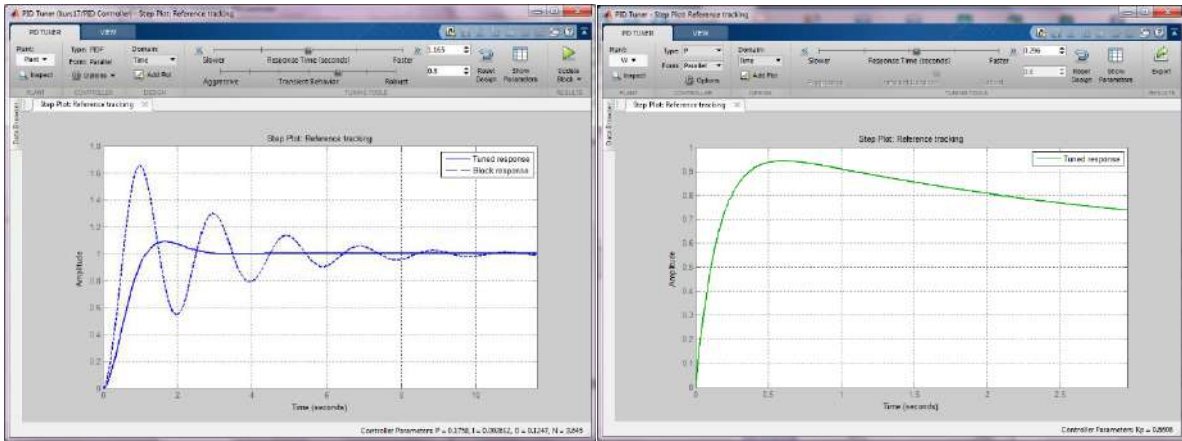


Рисунок 4.119 – Результат роботи PID-Tuner

Також можливо задати передавальну функцію в командному режимі й використовувати засоби PID-Tuner. Таким чином можна проводити ручне налаштування ПІД-регулятора за описаною вище методикою.

4.17 Синтез коригуючих пристроїв за логарифмічними частотними характеристиками

Закон управління вибирається з урахуванням можливості забезпечення стійкості і заданих вимог до якості системи. Зазвичай задаються вимоги до якості окремо в перехідному режимі і в сталому режимі. Проте не завжди вдається задовольнити заданим вимогам тільки вибором параметрів регуляторів, тоді в систему вводять коригуючі ланки. При цьому виникає задача синтезу коригуючої ланки. Виділяють три типу корегуючих ланок: послідовні, паралельні, зворотні.

Послідовні коригуючі ланки вводяться в систему послідовно з іншими ланками. На рис. 4.120 подана структурна схема системи з послідовною корегуючою ланкою.

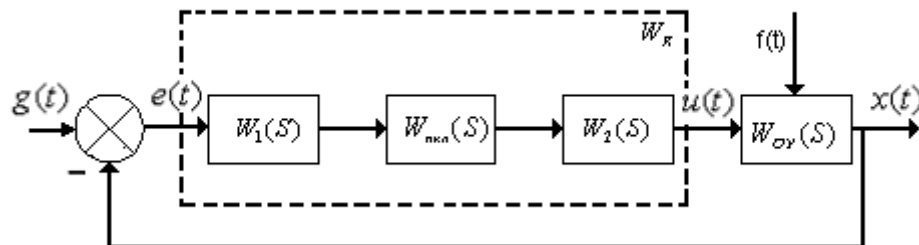


Рисунок 4.120– Структурна схема системи з послідовною корегуючою ланкою

Передавальні функції $W_1(s)$ та $W_2(s)$ представляють собою передавальні функції заданих частин регулятора, $W_{нкл}(s)$ – передавальна функція послідовної коригуючої ланки, $W_{об}(s)$ – передавальна функція об'єкта управління.

Передавальна функція регулятора з послідовною корегуючою ланкою:

$$W_{R1}(s) = W_1(s) \cdot W_2(s) \cdot W_{нкл}(s). \quad (4.11)$$

Спосіб корекції за допомогою послідовних коригуючих ланок не вимагає складних розрахунків і простий в практичній реалізації. Тому він знайшов широке застосування, особливо при корекції систем, в яких використовується електричний сигнал у вигляді напруги постійного струму, величина якого функціонально пов'язана з сигналом неузгодженості. Однак, послідовні коригувальні ланки не послаблюють впливу змін параметрів елементів системи на її показники якості. Тому послідовні коригувальні ланки рекомендується застосовувати в системах, в яких елементи мають досить стабільні параметри.

Паралельні коригуючі ланки вводяться в систему паралельно з іншими ланками. На рис. 4.121 подана структурна схема системи з паралельною корегуючою ланкою.

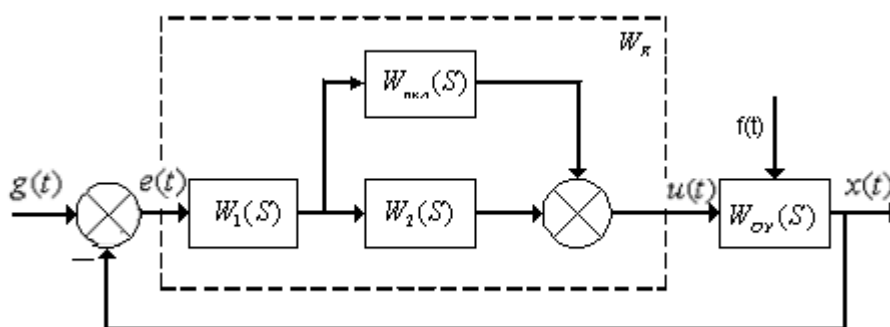


Рисунок 4.121 – Структурна схема системи з паралельною корегуючою ланкою

Передавальна функція регулятора з паралельною корегуючою ланкою:

$$W_{R2}(s) = W_1(s) \cdot (W_2(s) + W_{нкл}(s)). \quad (4.12)$$

Корекція систем управління за допомогою паралельних коригуючих ланок ефективна, коли потрібно формувати складні закони управління з введенням похідних і інтегралів від сигналу помилки. Прикладом цього можуть служити розглянуті раніше типові регулятори.

Зворотні коригуючі ланки вводяться в ланцюг регулятора і охоплюють будь-які його ланки. Зворотні зв'язки можуть бути позитивними і негативними, жорсткими і гнучкими. На рис. 4.122 подана структурна схема системи зі зворотною корегуючою ланкою.

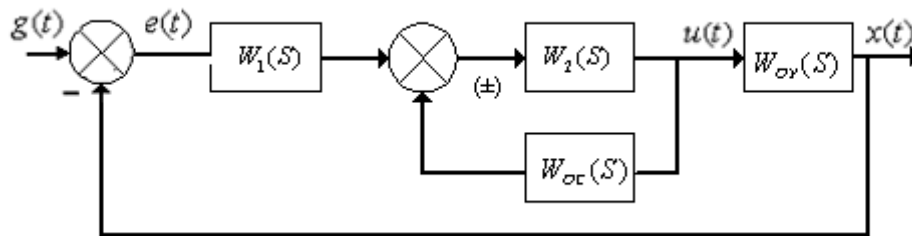


Рисунок 4.122 – Структурна схема системи зі зворотною корегуючою ланкою

Передавальна функція регулятора зі зворотною корегуючою ланкою:

$$W_{R3}(s) = W_1(s) \cdot \frac{W_2(s)}{1 \pm W_2(s)W_{oc}(s)} \quad (4.13)$$

Корекція місцевим зворотним зв'язком використовується в системах автоматичного управління найбільш часто. Коригувальний зворотний зв'язок утворює в системі внутрішній контур крім контуру, утвореного головним зворотним зв'язком. У переважній більшості випадків використовуються негативні коригувальні зворотні зв'язки, однак можуть застосовуватися також і позитивні зворотні зв'язки, наприклад в комбінованих системах з компенсацією динамічних помилок.

Негативний коригувальний зворотний зв'язок дозволяє істотно послаблювати вплив зміни параметрів елементів і нелінійностей, які входять до місцевого контуру. Тому місцевим зворотним зв'язком бажано охоплювати ті елементи корегуємої системи, які в процесі роботи можуть змінювати свої параметри і мають високі значення коефіцієнтів передачі.

Найбільш поширений на практиці є частотний метод синтезу послідовної корегуючої ланки за допомогою логарифмічних частотних характеристик. Логарифмічна амплітудна частотна характеристика розімкненої системи управління однозначно визначається її передавальною функцією і відповідно навпаки, за логарифмічною амплітудною частотною характеристикою можна однозначно скласти передавальну функцію розімкненої системи.

Отже, на основі вимог, що пред'являються до системи можна сформуванати бажаний вид ЛАЧХ, якій буде відповідати необхідна передавальна функція системи і закон управління. На основі цього взаємозв'язку і побудований метод синтезу систем автоматичного управління за логарифмічним частотним характеристикам.

Процес синтезу системи управління включає в себе наступні етапи.

- будується ЛАЧХ заданої (нескорегованої) системи;
- будується бажана ЛАЧХ по заданих показниках якості перехідного процесу;
- будується ЛАЧХ послідовної корегуючої ланки шляхом графічного вирахування ЛАЧХ заданої системи з ЛАЧХ бажаної системи. Передавальна функція послідовної корегуючої ланки записується по виду її ЛАЧХ.

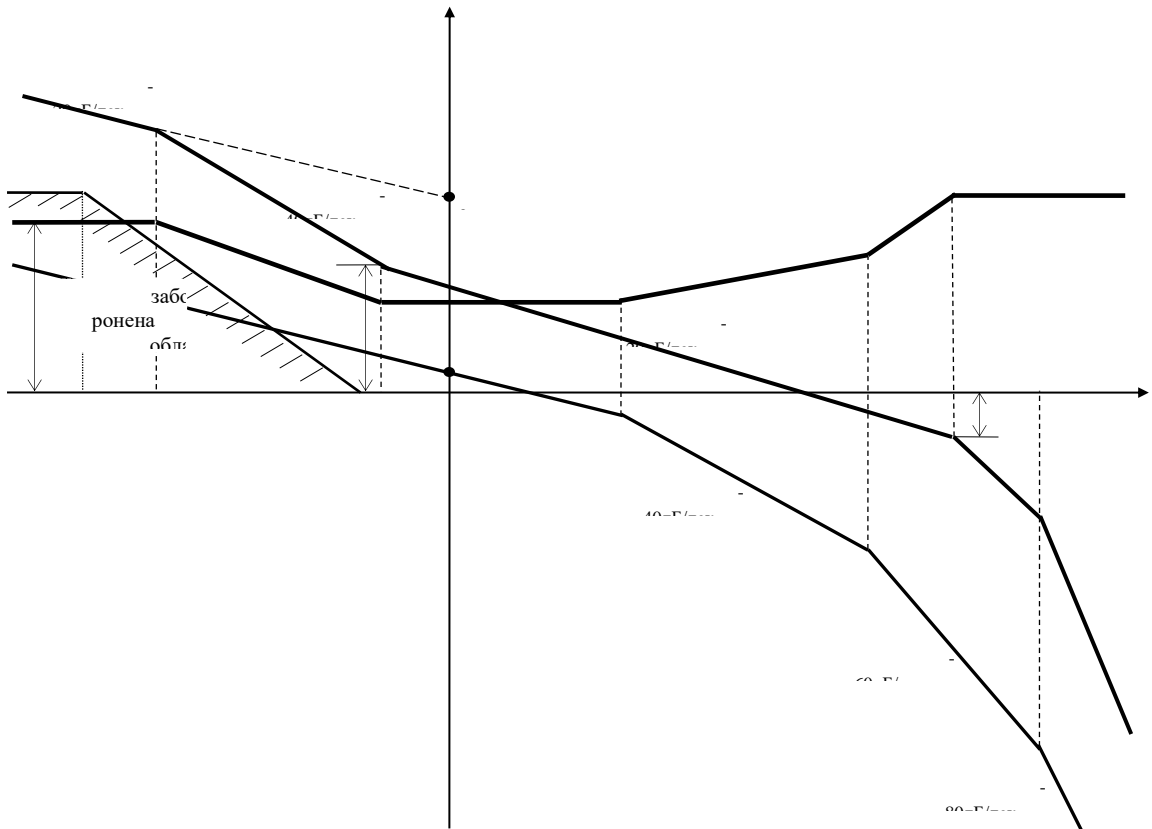
При побудові бажаної ЛАЧХ бажаної САУ виділяють три області: область низьких частот, область середніх частот і область високих частот. Вид ЛАЧХ в кожній з областей по різному впливає на якість системи. У області низьких частот вид ЛАЧХ визначає точність роботи системи в сталих режимах. Область середніх частот визначає динамічні властивості системи (швидкодія, коливальність). Вид ЛАЧХ в області високих частот практично мало впливає на якість системи.

Побудову бажаної ЛАЧХ починають з області середніх частот в такій послідовності. По заданих величинах σ_{max} і t_p визначають частоту зрізу $\omega_{зр}$ за значеннями у табл. 4.8.

Таблиця 4.8– Таблиця параметрів для розрахунку $\omega_{зр}$

$\sigma_{max}, \%$	10	15	20	25	30	35	40
$t_p \cdot \omega_{зр} / \pi$	5,0	4,4	4,0	3,6	3,2	3,0	2,8
$L_1, \text{дБ}$	18,0	15,0	13,5	12,0	11,0	10,5	10,0
$\Delta\varphi, \text{гр}$	85	80	65	55	45	40	35

На вісь абсцис наноситься частота зрізу $\omega_{зр}$ і через неї проводиться пряма лінія з нахилом мінус 20 дБ/дек (рис. 4.123).



б) ЛАЧХ бажаної системи; в) ЛАЧХ послідовної корегуючої ланки.
 Рисунок 4.123 – Частотні характеристики системи: а) ЛАЧХ заданої системи;

Частота ω_2 , що обмежує область середніх частот бажаної ЛАЧХ зліва визначається величиною відрізка L_1 , яку знаходять в залежності від заданої величини σ_{max} по табл. 4.5. Частоту ω_3 , що обмежує область середніх частот праворуч, визначають величиною відрізка L_2 , при цьому $|L_2| \geq L_1$.

У області високих частот бажану ЛАЧХ будують у вигляді прямолінійних відрізків з нахилом, кратним 20 дБ/дек. (тобто 40, 60, 80 і т.д.), таким чином, щоб різниця характеристик бажаної і заданої в межі при $\omega \rightarrow \infty$ складала пряму лінію, паралельну осі частот.

У області низьких частот бажану ЛАЧХ будують таким чином. Визначають величину

$$L_{A2} = 20 \lg K_c, \quad (4.14)$$

де K_c – коефіцієнт посилення бажаної системи:

$$K_c = \frac{\omega_{0max}}{x_{max}}$$

На графік наносять точку A_2 з координатами $(1; L_{A2})$ і через неї проводять пряму лінію з нахилом мінус 20 дБ/дек. Від точки М, що обмежує область середніх частот зліва, проводимо пряму лінію з нахилом мінус 40 дБ/дек до перетину з низькочастотною частиною бажаної ЛАЧХ.

Якщо в завданні на проектування системи вказана максимально допустима помилка X_m при умові, що вхідний сигнал може змінюватися з максимальною кутовою швидкістю ω_{0max} і з максимальним кутовим прискоренням ε_{0max} , то для виконання цих вимог необхідно, щоб бажана ЛАЧХ не попадала в заборонену область.

Заборонена область будується таким чином. Відмічаємо на кресленні точку В ($lg \omega_B; L_B$):

$$\omega_B = \frac{\varepsilon_{0max}}{\omega_{0max}}, \quad L_B = 20 \lg \frac{\omega_{0max}^2}{\varepsilon_{0max} X_{max}}.$$

Від точки В праворуч проводять лінію з нахилом мінус 40 дБ/дек. Якщо бажана ЛАЧХ попадає в заборонену область, то це означає, що при даному коефіцієнті K_c задана точність стеження не може бути забезпечена і потрібно її збільшити, тобто графічно підняти бажану ЛАЧХ так, щоб вона не попадала в заборонену область.

ЛАЧХ послідовної корегуючої ланки будується шляхом графічного вирашування з ЛАЧХ бажаної системи ЛАЧХ заданої системи.

По виду ЛАЧХ записуються передавальні функції бажаної системи та послідовної корегуючої ланки. Коефіцієнт посилення корегуючої ланки $K_{кр}$ визначають з співвідношення:

$$20 \lg K_{кр} = L_3.$$

Далі проводиться моделювання системи і визначення показників якості. Моделювання є останньою й однією з найважливіших задач синтезу систем автоматичного управління, що дозволяє імітувати поведінку реальної системи в різних умовах експлуатації, передбачити аварійні ситуації або підвищення навантаження на елементи системи без ризику їхнього руйнування. Моделювання перехідних процесів є обов'язковим етапом дослідження умовно стійких і нестійких систем. Воно заміняє експерименти з реальними дорогими об'єктами, які в робочих умовах завжди повинні функціонувати стійко, надійно й безпечно. По отриманим характеристикам розраховують прямі показники

якості системи в перехідному режимі й коефіцієнти помилок системи в сталому режимі та переконуються, що отримані показники якості задовольняють вимогам.

Якщо передавальна функція послідовної коригуючої ланки визначена викладеним вище способом та задана система має передавальну функцію W_3 і складається з двох ланок з передавальними функціями W_{31} і W_{32} і паралельна коректуюча ланка включається паралельно першій ланці. Передавальна функція розімкненої системи з паралельною коригуючою ланкою має вигляд:

$$W = (W_{31} + W_{кз пар}) \cdot W_{32}. \quad (4.15)$$

З умови рівності цієї передавальної функції бажаної передавальної функції $W_{жс} = W_{кз нас} \cdot W_3$ для передавальної функції корегуючої ланки одержуємо:

$$W_{кз пар} = W_{31}(W_{кз нос} - 1). \quad (4.16)$$

Якщо коригуюча ланка включена в зворотний зв'язок, то передавальна функція розімкненої системи:

$$W = \frac{W_{31}W_{32}}{1+W_{31}W_{кз обр}}. \quad (4.17)$$

З рівності цієї передавальної функції бажаною для передавальної функції зворотної коригуючої ланки одержуємо:

$$W_{кз обр} = \frac{1 - W_{кз нос}}{W_{31}W_{32}} \quad (4.18)$$

Розглянемо вирішення завдання синтезувати послідовну коригуючу ланку так, щоб замкнута система з одиничним від'ємним зворотнім зв'язком задовольняла прямим показникам якості: час регулювання $t_p = 0,2$ с, перерегулювання $\sigma_{max} = 20\%$. Передатна функція її розімкненої системи дорівнює:

$$W(s) = \frac{500}{s(0,033s + 1)(0,0125s + 1)}.$$

Для цього побудуємо ЛАЧХ заданої розімкненої системи (рис. 4.124). Для цього відмітимо точку $\omega = 1 \text{ с}^{-1}$, $L(\omega) = 20 \lg k = 20 \lg 500 \approx 54 \text{ дБ}$. Оскільки в знаменнику змінна s у першому ступені, то нахил першої асимптоти $-\nu \cdot 20 \frac{\text{дБ}}{\text{дек}} = -20 \text{ дБ/дек}$.

Знайдемо частоти спряження асимптотичної ЛАЧХ:

$$\omega_1 = \frac{1}{T_1} = \frac{1}{0,033} = 30 \text{ с}^{-1};$$

$$\omega_2 = \frac{1}{T_2} = \frac{1}{0,0125} = 80 \text{ с}^{-1}.$$

На першій частоті спряження ω_1 нахил асимптоти зміниться на $(-20) \text{ дБ/дек}$ і дорівнюватиме $(-40) \text{ дБ/дек}$. На другій частоті спряження ω_2 нахил асимптоти знову зміниться на $(-20) \text{ дБ/дек}$ і дорівнюватиме $(-60) \text{ дБ/дек}$.

Асимптотична ЛАЧХ заданої розімкненої системи подана на рис. 4.58.

Побудуємо бажану ЛАЧХ (на рис. 4.124 – $L_{\text{баж}}$).

З табл. 4.7 для σ_{max} визначаємо, що $\frac{t_p \cdot \omega_c}{\pi} = 4,0$, звідки $\omega_c = \frac{4,0 \cdot \pi}{t_p} \approx 63$.

Нанесемо на вісь абсцис частоту зрізу ω_c і проведемо через неї пряму лінію з нахилом $(-20) \text{ дБ/дек}$ (рис. 4.124, $L_{\text{баз}}$).

Знайдемо ω_2 та ω_3 . За таблицею 4.8 знайдемо, що $L_1 = 13,5 \text{ дБ}$. Частота, для якої ордината дорівнює $13,5 \text{ дБ}$, і буде $\omega_2 = 9,2 \text{ с}^{-1}$. Оберемо $L_2 = 15 \text{ дБ}$, тоді $\omega_3 = 400 \text{ с}^{-1}$.

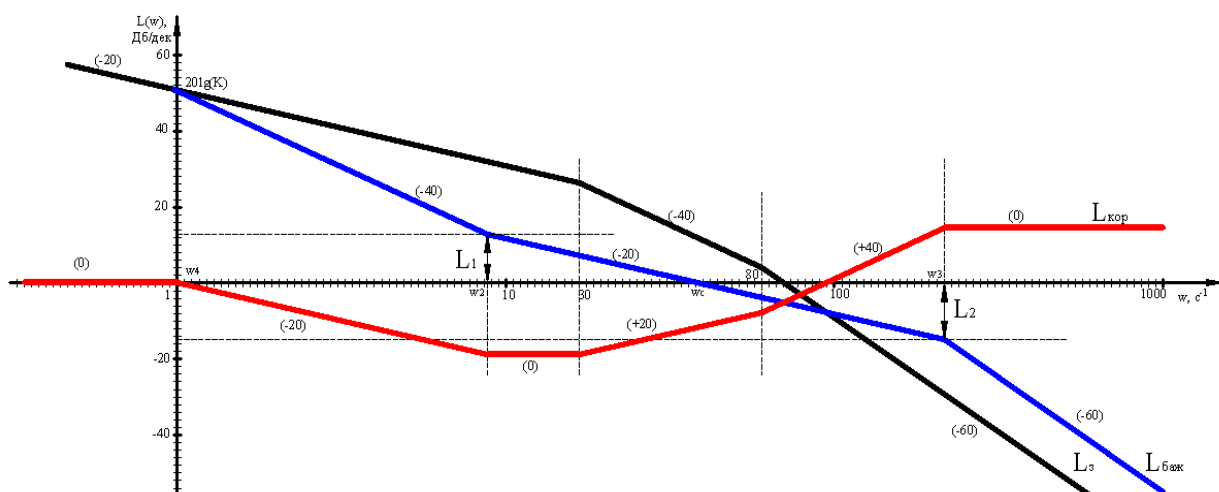


Рисунок 4.124 – Асимптотичні ЛАЧХ системи

З'єднаємо низькочастотну асимптоту з асимптотою у середньочастотній області лінією з нахилом (-40) дБ/дек.

Високочастотну асимптоту будуємо паралельною високочастотній асимптоті заданої ЛАЧХ, таким чином з нахилом (-60) дБ/дек.

За видом бажаної ЛАЧХ можна записати передатну функцію:

$$W_{\text{баж}}(s) = \frac{500 \cdot (0,109s + 1)}{s(1,124s + 1)(0,025s + 1)^2}.$$

ЛАЧХ послідовної коригуючої ланки будуємо шляхом графічного вирахування ЛАЧХ заданої системи з ЛАЧХ бажаної (на рис. 4.124 $L_{\text{кор}}$), тобто

$$L_{\text{кор}} = L_{\text{баж}} - L_{\text{з}}.$$

За видом ЛАЧХ коригуючої ланки запишемо передавальну функцію:

$$W_{\text{кор}}(s) = \frac{(0,109s + 1)(0,033s + 1)(0,0123s + 1)}{(1,124s + 1)(0,025s + 1)^2}.$$

Проводимо моделювання отриманої системи з послідовною коригуючою ланкою з метою одержання прямих показників якості. Отримані показники якості відповідають заданим.

4.18 Синтез лінійної неперервної системи автоматичного управління

Синтез системи управління це спрямований розрахунок системи, що має кінцевою метою, по-перше, відшукування раціональної структури системи і, по-друге, визначення оптимальних значень параметрів її окремих ланок за умови забезпечення ряду вимог, які впливають з призначення системи і забезпечення її характеристик .

Таким чином, синтез можна трактувати як задачу оптимізації і розглядати таку побудову системи управління, при якому для заданих умов роботи забезпечується оптимум обраного критерію якості роботи системи. Якщо характеристики задаючих впливів і впливів обурення відомі, то систему можна спроектувати як оптимальну, забезпечивши мінімальне значення сумарної середньоквадратичної помилки.

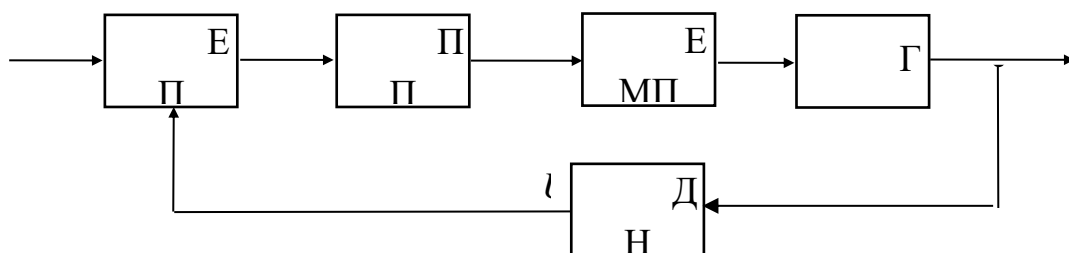
Синтез можна трактувати як інженерну задачу, яка зводиться до побудови системи автоматичного управління, яка забезпечує виконання технічних вимог до неї. Один з можливих способів опису вимог до системи, яка проектується – це завдання показників якості роботи системи. Це може бути зроблено, якщо відомі характеристики задаючих впливів і впливів обурення. Детерміновані впливи повинні бути задані як функції часу або їх похідні. Для випадкових впливів повинні бути відомі їх кореляційні функції або спектральні щільності. При такій постановці синтез системи зводиться до вибору структурної схеми, за допомогою якої можна забезпечити показники якості роботи системи не гірше заданих.

Іноді в поняття інженерного синтезу вкладається ще більш вузький зміст і розглядається синтез, який має на меті визначення виду і параметрів коригувальних засобів, які необхідно додати до постійної частини системи - об'єкта управління з регулятором, щоб забезпечити необхідні динамічні якості. Забезпечення необхідної якості управління досягається формуванням певного закону управління $u(t)$. Для цього необхідно, щоб при відомій передавальній функції об'єкта управління $W_{oy}(s)$ регулятор мав певну передавальну функцію $W_R(s)$ і, відповідно, передавальна функція розімкненої системи має бути:

$$W(s) = W_R(s) \cdot W_{oy}(s).$$

При інженерному синтезі системи управління необхідно забезпечити, по-перше, необхідну точність, по-друге, прийнятний характер перехідних процесів.

Наприклад, лінійна неперервна система автоматичного управління регулювання напруги (рис. 4.125) містить у собі елемент порівняння (ЕП), попередній підсилювач (ПП), електромагнітний підсилювач (ЕМП), генератор (Г) і дільник напруги (ДН) у ланцюзі негативного зворотного зв'язка.



U_0 – вхідне значення напруги, $U_{ЕП}$ – напруга на виході ЕП,
 $U_{ПП}$ – напруга на виході попереднього підсилювача,
 $U_{ЕМП}$ – напруга на виході електромагнітного підсилювача,
 $U_{Г}$ – напруга на виході генератора, $U_{ДН}$ – напруга на виході ДН

Рисунок 4.125 – Функціональна схема САУ

Елементи САУ описуються наступними рівняннями:

– елемент порівняння:

$$U_{EP} = U_0 - U_{DH},$$

– попередній підсилювач:

$$T_{ПП} \frac{dU_{ПП}}{dt} + U_{ПП} = K_{ПП} U_{EP},$$

– електромашинний підсилювач:

$$T_2^2 \frac{d^2 U_{EMП}}{dt^2} + T_1 \frac{dU_{EMП}}{dt} + U_{EMП} = K_{EMП} U_{ПП},$$

– генератор:

$$T_G \frac{dU_G}{dt} + U_G = K_G U_{EMП},$$

– дільник напруги:

$$U_{DH} = K_{DH} U_G,$$

Значення параметрів САУ подано у табл. 4.9.

Таблиця 4.9 – Значення параметрів САУ

$T_{ПП}$	$K_{EMП}$	T_1	T_2	K_G	T_G	K_{DH}	σ	$t_{пер}$
0,14	5,5	0,6	0,015	5	0,25	1/20	20	2,5

Для заданої системи необхідно: обчислити передавальні функції всіх елементів системи; побудувати структурну схему нескорегованої системи; визначити передавальні функції розімкнутої й замкнутої системи регулювання, а також їхні характеристичні поліноми; побудувати область стійкості замкнутої системи методом D-розбивки по невідомому коефіцієнту підсилення; оцінити стійкість розімкнутої системи по кореням характеристичного рівняння, стійкість замкнутої системи, використовуючи критерій Гурвіца, Михайлова й Найквіста, визначити запаси стійкості по амплітуді й фазі; побудувати ЛАЧХ і ЛФЧХ вихідної розімкнутої системи, сформулювати бажану ЛАЧХ із умови забезпечення необхідних значень швидкодії, перерегулювання й точності; побудувати ЛАЧХ коригуючої ланки, знайти її передавальну функцію; побудувати структурну схему системи з коригуючою ланкою; знайти позиційну, швидкісну помилку та помилку по прискоренню скорегованої системи, запаси стійкості по амплітуді й

фазі; зрівняти їх з аналогічними характеристиками вихідної системи; розрахувати перехідну характеристику, змодельовати САУ і порівняти результати; оцінити якість управління в скорегованій системі.

Необхідно скласти структурну схему досліджуваної системи та розрахувати передатні функції елементів. Елемент порівняння на структурній схемі зображується у вигляді круга, розділеного на сектор. Сектор, на який подається величина із зворотним (негативним) знаком, затемнюють або перед відповідним входом ставлять знак мінус.

Передатна функція попереднього підсилювача дорівнює:

$$W_{\text{ПП}}(p) = \frac{K_{\text{ПП}}}{0,14p + 1}$$

Передатна функція електромашинного підсилювача дорівнює:

$$W_{\text{ЕМП}}(p) = \frac{5.5}{0,000225p^2 + 0,6p + 1}$$

Передатна функція генератора дорівнює:

$$W_{\text{Г}}(p) = \frac{5}{0,25p + 1}$$

Передатна функція дільника напруги дорівнює:

$$W_{\text{ДН}}(p) = \frac{1}{20}$$

Структурна схема заданої системи зображена на рис. 4.126.

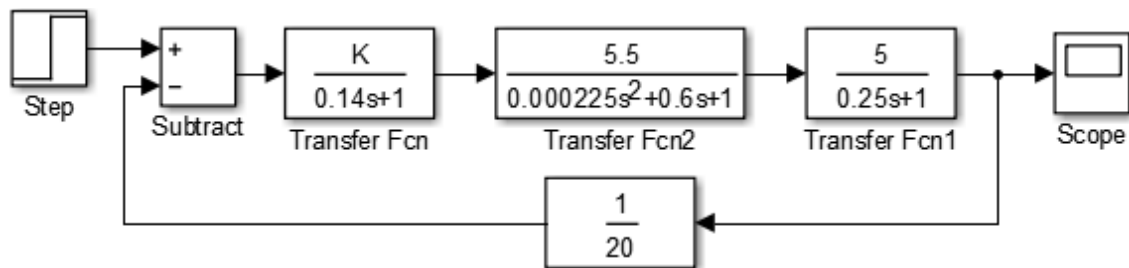


Рисунок 4.126 – Структурна схема заданої системи

За допомогою структурної схеми можемо розрахувати передатні функції замкненої та розімкненої систем. На рис. 4.127 подано фрагмент програмного коду пакету прикладних програм MatLab для визначення передатних функцій замкнутої та розімкнутої систем.

```
>> W1=tf([4],[0.14 1])
W2=tf([5.5],[0.000225 0.6 1])
W3=tf([5],[0.25 1])
W4=tf([1],[20])
Wraz=series(W1,W2)
Wraz=series(Wraz,W3)
Wraz=series(Wraz,W4)
Wzamk=W1*W2*W3
Wzamk=feedback(Wzamk,W4)
```

Рисунок 4.127 – Фрагмент програмного коду пакету прикладних програм MatLab для визначення передатних функцій замкнутої та розімкнутої систем

Передатна функція розімкнутої системи:

$$W_{\text{роз}}(s) = \frac{27,5K_{\Pi}}{0,0001575s^4 + 0,421755s^3 + 5,3845s^2 + 19,8s + 20}$$

Передатна функція замкнутої системи:

$$W_{\text{замк}}(s) = \frac{550K_{\Pi}}{0,0001575s^4 + 0,421755s^3 + 5,3845s^2 + 19,8s + 20 + 27,5K_{\Pi}}$$

В розрахованих передатних функціях є невідомий параметр K_{Π} . Розрахуємо його за допомогою методу D-розбивки.

Характеристичний поліном замкнутої системи визначаємо за передавальною функцією замкнутої системи:

$$D_{\text{замк}}(\lambda) = 0,0001575\lambda^4 + 0,421755\lambda^3 + 5,3845\lambda^2 + 19,8\lambda + 20 + 27,5K_{\Pi};$$

$$R(\lambda) = 0,0001575\lambda^4 + 0,421755\lambda^3 + 5,3845\lambda^2 + 19,8\lambda + 200;$$

$$Q(\lambda) = 27,5.$$

Заміняємо λ на $j\omega$ і отримуємо вираз:

$$K_{\Pi}(i\omega) = \frac{-R(i\omega)}{Q(i\omega)} = \frac{-(0,0001575\omega^4 - 0,421755i\omega^3 - 5,3845\omega^2 + 19,8i\omega + 20)}{27,5}$$

Звідси знаходимо дійсну $U(\omega)$ та уявну функції $V(\omega)$. Задаючи значення ω від -1000 до 1000 будуємо область стійкості. По осі абсцис будуємо дійсну складову $U(\omega)$, а по осі ординат - уявну $V(\omega)$.

На рис. 4.126 подано фрагмент програмного коду пакету прикладних програм MatLab для побудови кривої D розбивки Графік області стійкості зображений на рис. 4.127.

За графіком бачимо, що область стійкості розташована в межах від 0 до 8. Обираємо коефіцієнт підсилення $K_{\Pi} = 4$.

```
>> symw w;  
Kp_jw = -(0.0001575*w^4-0.421755*i*w^3-5.3845*w^2+19.8*i*w+20)/27.5;  
Re = []; Im = [];  
for w=-1000:0.1:1000  
Kp_jw = -(0.0001575*w^4-0.421755*i*w^3-5.3845*w^2+19.8*i*w+20)/27.5;  
Re1 = real(Kp_jw);  
Im1 = imag(Kp_jw);  
Re = [Re,Re1];  
Im = [Im,Im1];  
end  
figure, plot(Re,Im)  
grid on  
xlabel('Re (Kp) '); ylabel('Im (Kp) ');
```

Рисунок 4.126 – Фрагмент програмного коду пакету прикладних програм MatLab для побудови АФЧХ

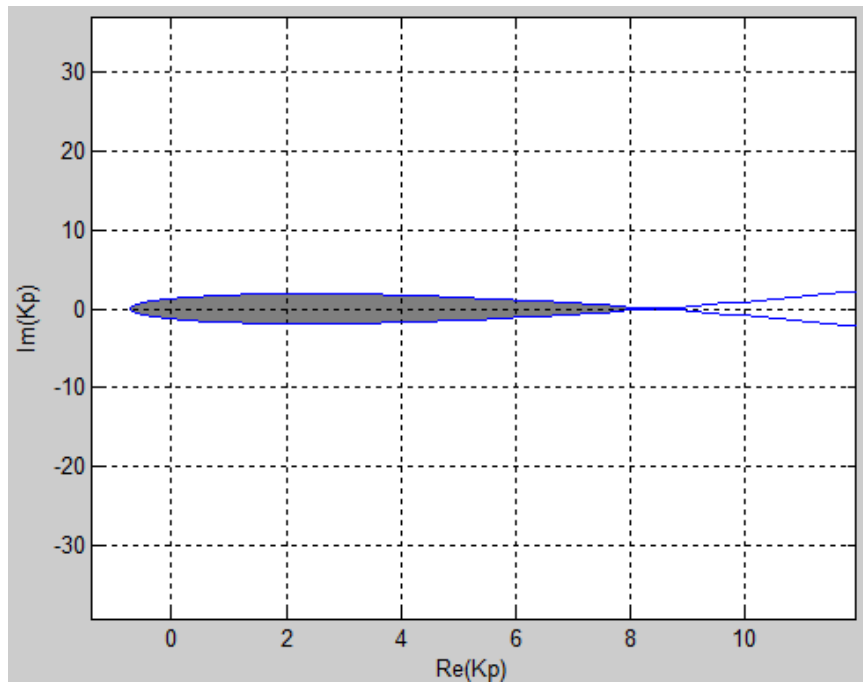


Рисунок 4.127 – Зображення замкненої поверхні D

Таким чином, передатні функції розімкнутої та замкнутої САУ приймуть наступний вигляд:

$$W_{\text{роз}}(s) = \frac{110}{0,0001575s^4 + 0,421755s^3 + 5,3845s^2 + 19,8s + 20};$$

$$W_{\text{замк}}(s) = \frac{2200}{0,0001575s^4 + 0,421755s^3 + 5,3845s^2 + 19,8s + 130}$$

Проведемо оцінку стійкості заданої САУ наступними методами: оцінка розімкнутої системи за коренями характеристичного рівняння; оцінка замкнутої системи, використовуючи критерій Гурвіца, Михайлова та Найквіста. Також потрібно визначимо запаси стійкості за амплітудою і фазою.

Для оцінка стійкості розімкнутої САУ за коренями характеристичного рівняння складемо характеристичне рівняння розімкнутої САУ:

$$0,0001575\lambda^4 + 0,421755\lambda^3 + 5,3845\lambda^2 + 19,8\lambda + 20 = 0.$$

Знайдемо корені характеристичного рівняння:

$$\lambda_1 = -2665; \lambda_2 = -7,14; \lambda_3 = -4; \lambda_4 = -1,66.$$

У методі оцінки стійкості розімкнутої САУ за коренями характеристичного рівняння необхідна та достатня умова полягає у тому, щоб всі корені характеристичного рівняння були від'ємні, тобто знаходилися у лівій півплощині. Так як усі корені нашого характеристичного рівняння від'ємні, то можна зробити висновок, що система стійка.

Проведемо аналіз стійкості замкненої системи за критерієм Гурвіца. Необхідна умова стійкості: для того, щоб система була стійкою необхідно та достатньо, щоб усі її коефіцієнти характеристичного рівняння були одного знаку.

Характеристичне рівняння замкненої системи:

$$0,0001575\lambda^4 + 0,421755\lambda^3 + 5,3845\lambda^2 + 19,8\lambda + 130 = 0;$$

$$a_0 = 0,0001575; a_1 = 0,421755; a_2 = 5,3845; a_3 = 19,8; a_4 = 130.$$

Алгебраїчний критерій стійкості Гурвіца формулюється так: для того, щоб система автоматичного управління була стійкою, необхідно та достатньо, щоб усі визначники Гурвіца були додатними $\Delta_i > 0$, $i = \overline{1, n}$ при $a_0 > 0$.

$$\Delta_4 = \begin{vmatrix} a_1 & a_3 & 0 & 0 \\ a_0 & a_2 & a_4 & 0 \\ 0 & a_1 & a_3 & 0 \\ 0 & a_0 & a_2 & a_4 \end{vmatrix} = \begin{vmatrix} 0,421755 & 19,8 & 0 & 0 \\ 0,0001575 & 5,3845 & 130 & 0 \\ 0 & 0,421755 & 19,8 & 0 \\ 0 & 0,0001575 & 5,3845 & 130 \end{vmatrix} = 2831;$$

$$\Delta_3 = \begin{vmatrix} a_1 & a_3 & 0 \\ a_0 & a_2 & a_4 \\ 0 & a_1 & a_3 \end{vmatrix} = \begin{vmatrix} 0,421755 & 19,8 & 0 \\ 0,0001575 & 5,3845 & 130 \\ 0 & 0,421755 & 19,8 \end{vmatrix} = 21,77;$$

$$\Delta_2 = \begin{vmatrix} a_1 & a_3 \\ a_0 & a_2 \end{vmatrix} = \begin{vmatrix} 0,421755 & 19,8 \\ 0,0001575 & 5,3845 \end{vmatrix} = 2,26;$$

$$\Delta_1 = a_1 = 0,421755.$$

Згідно з необхідною та достатньою умовою стійкості по критерію Гурвіца $a_0 > 0, \Delta_1 > 0, \Delta_2 > 0, \Delta_3 > 0, \Delta_4 > 0$ випливає, що замкнута система стійка.

Проведемо аналіз стійкості замкненої системи за критерієм Михайлова. Частотний критерій стійкості Михайлова формулюється так: для того, щоб система автоматичного управління була стійкою, необхідно та достатньо, щоб крива Михайлова при зміні частоти ω від 0 до $+\infty$, починаючись при $a_0 > 0$ на

дійсній додатній півосі, обходила послідовно n квадрантів (чвертей) координатної площини проти руху годинникової стрілки, не потрапляючи у початок координат.

Для оцінки стійкості треба записати характеристичне рівняння замкнутої САУ та замінити $\lambda = i\omega$ для знаходження комплексного виразу характеристичного рівняння системи:

$$D_{\text{замк}}(i\omega) = 0,0001575\omega^4 - 0,421755i\omega^3 - 5,3845\omega^2 + 19,8i\omega + 130.$$

Далі знаходимо дійсну $U(\omega)$ та уявну $V(\omega)$ з отриманого виразу (рис. 4.128) та будуємо криву Михайлова, яка зображена на рис. 4.129.

```
>> syms w;
Djw = 0.0001575*w^4-0.421755*i*w^3-5.3845*w^2+19.8*i*w+130;
Re = []; Im = [];
for w = 0:0.1:1000
Djw = 0.0001575*w^4-0.421755*i*w^3-5.3845*w^2+19.8*i*w+130;
Re1 = real(Djw);
Im1 = imag(Djw);
Re = [Re,Re1];
Im = [Im,Im1];
end
figure, plot(Re,Im)
grid on
xlabel('Re(Djw)'); ylabel('Im(Djw)');
```

Рисунок 4.128 – Фрагмент програмного коду для побудування годографа Михайлова

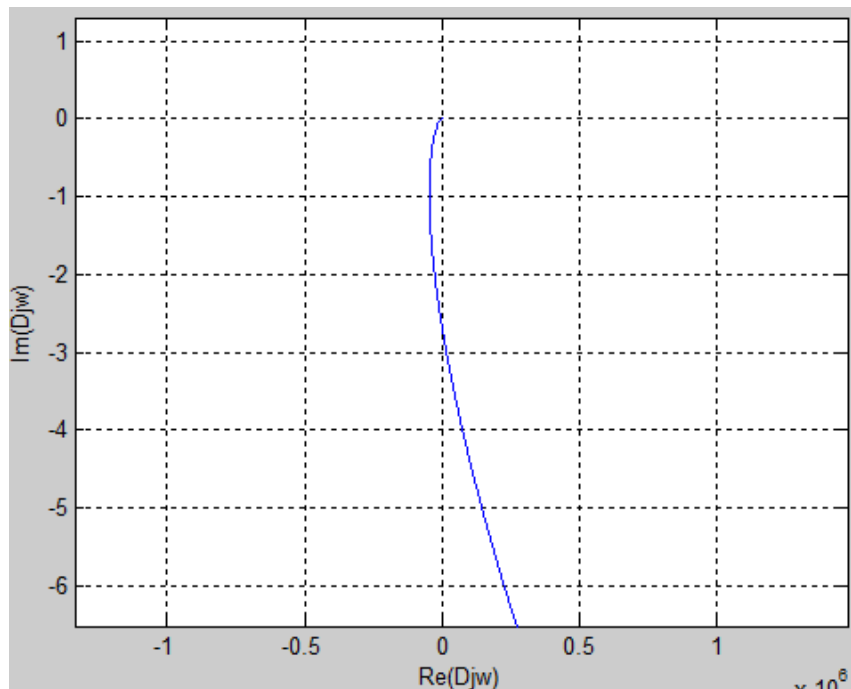


Рисунок 4.129 – Зображення годографа Михайлова

Щоб переконатися в тому, що годограф обходить n квадрантів, подивимося на збільшене зображення годографа Михайлова (рис. 4.130).

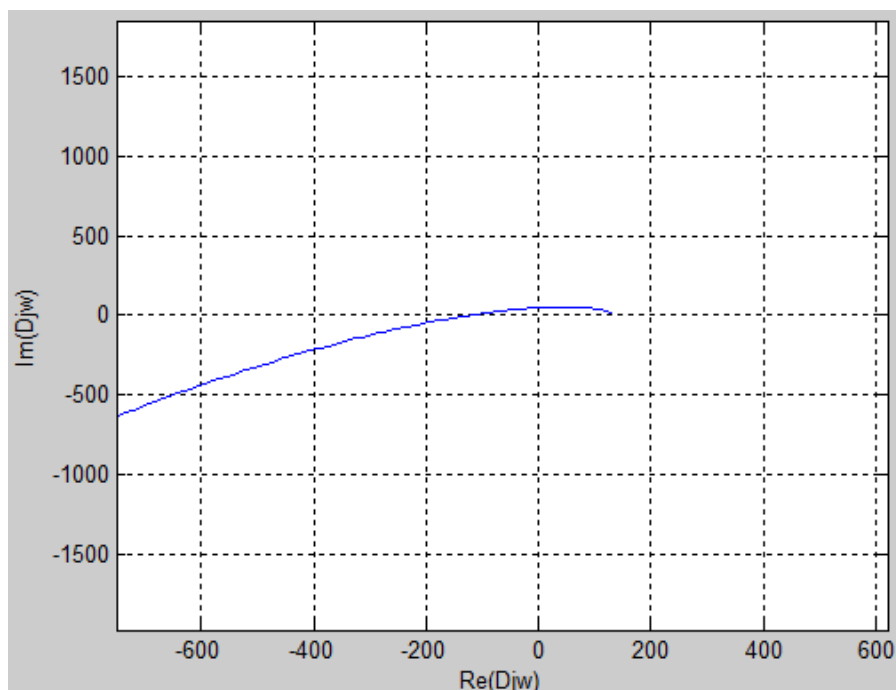


Рисунок 4.130 – Збільшене зображення годографа Михайлова

Проведемо аналіз стійкості замкненої системи за критерієм Найквіста. Критерій Найквіста: для того, щоб замкнута система була стійкою необхідно і достатньо, щоб АФЧХ, її розімкнутої системи зі зростанням частоти від 0 до ∞ охоплювало точку $(-1; j_0)$ в позитивному напрямку (проти напрямку часової стрілки) $\frac{l}{2}$ раз, де l – кількість правих коренів характеристичного рівняння розімкнутої системи.

Для того, щоб оцінити стійкість САУ за критерієм Найквіста запишемо характеристичний поліном розімкнутої системи та знайдемо його корні:

$$D_{\text{роз}}(\lambda) = 0.0001575\lambda^4 + 0.421755\lambda^3 + 5.3845\lambda^2 + 19.8\lambda + 20;$$

$$\lambda_1 = -2665; \lambda_2 = -7.14; \lambda_3 = -4; \lambda = -1.66.$$

Побудуємо годограф Найквіста заданої розімкнутої системи. Даний графік зображений на рис. 4.130.

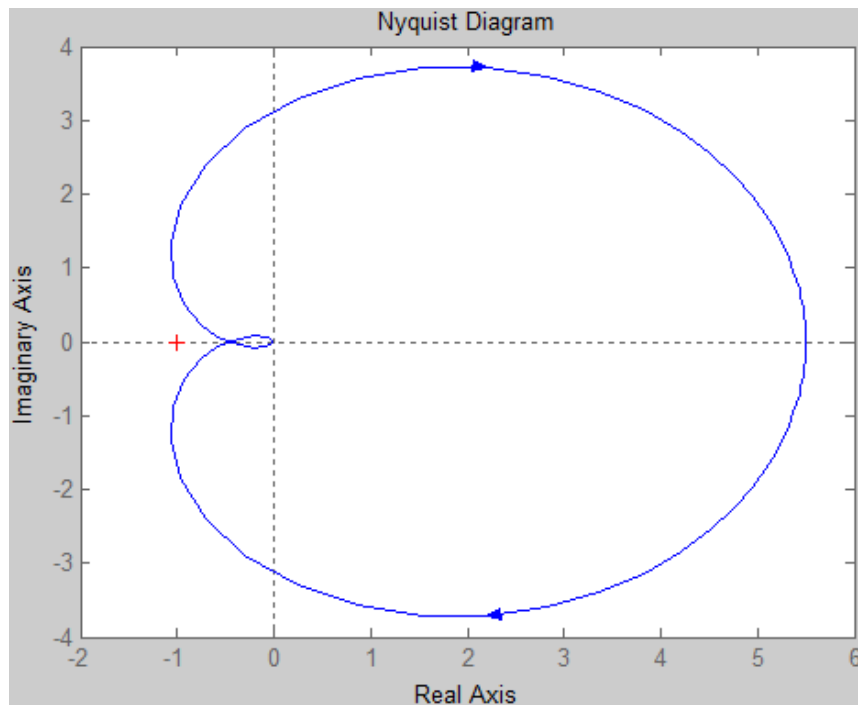


Рисунок 4.130 – Годограф Найквіста

Оскільки всі корені розімкненої системи від’ємні, то АФЧХ не охоплює точку $(-1; j_0)$, що можна побачити на годографі Найквіста. Тому можна зробити висновок, що замкнута САУ стійка за критерієм Найквіста.

Для визначення запасу стійкості за амплітудою та фазою побудуємо ЛАЧХ та ЛФЧХ заданої нескорегованої САУ за допомогою програмного засобу Matlab. Результат зображено на рис. 4.131.

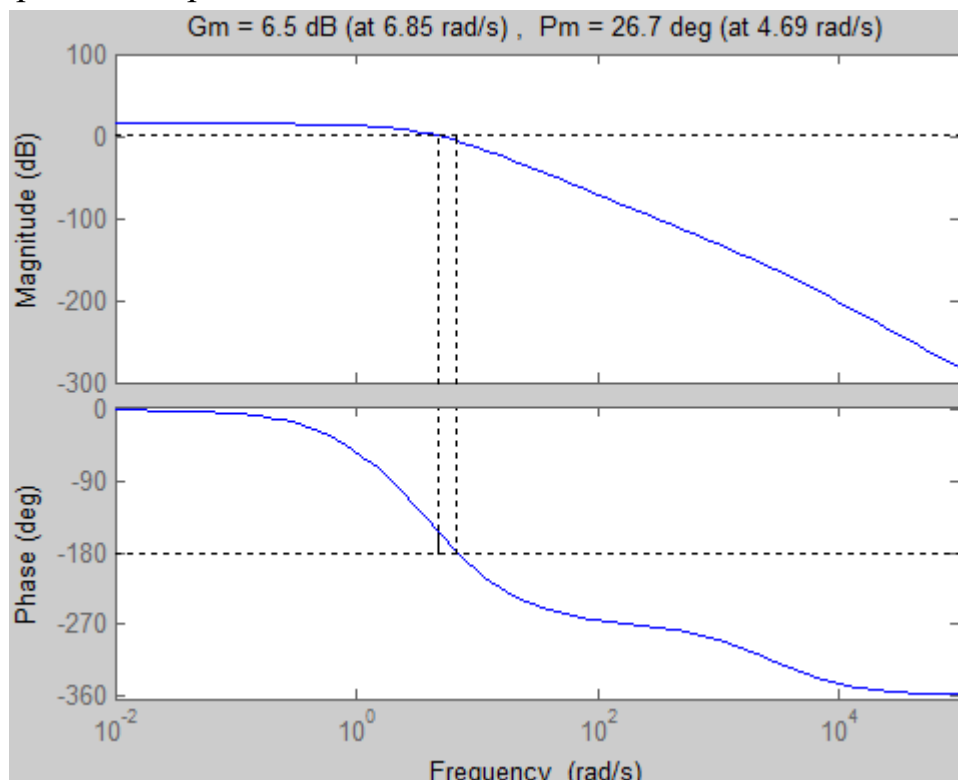


Рисунок 4.131 – ЛАЧХ і ЛФЧХ нескорегованої САУ

З отриманих графіків визначимо запаси стійкості досліджуваної за амплітудою ΔL та фазою $\Delta\varphi$:

$$\begin{aligned}\Delta L &= 6,5 \text{ дБ}; \\ \Delta\varphi &= 26,7 \text{ deg}.\end{aligned}$$

Синтезуємо послідовну коригуючу ланку за допомогою частотного методу синтезу. Передавальна функція послідовної коригуючої ланки записується по виду її ЛАЧХ. Побудова ЛАЧХ послідовної коригуючої ланки проводиться в такій послідовності:

- будується ЛАЧХ заданої (нескорегованої) системи;
- будується бажана ЛАЧХ по заданих показниках якості перехідного процесу;
- будується ЛАЧХ послідовної коригуючої ланки шляхом графічного вирахування з ЛАЧХ бажаної системи ЛАЧХ заданої системи.

Для побудови ЛАЧХ заданої САУ візьмемо передавальну функцію розімкненої системи:

$$W_{\text{роз}}(s) = \frac{110}{0,0001575s^4 + 0,421755s^3 + 5,3845s^2 + 19,8s + 20}$$

та приведемо її до зручного вигляду:

$$W_{\text{роз}}(s) = \frac{5,5}{(0,14s + 1)(0,000225s^2 + 0,6s + 1)(0,25s + 1)}$$

Далі знаходимо точку перетину ЛАЧХ заданої системи з віссю ординат:

$$20 \lg(K) = 20 \lg(5,5) = 14,8.$$

Отже точка перетину $A_1(0; 14,8)$.

З нахилом:

$$-9 \cdot 20 \frac{\text{дБ}}{\text{дек}} = 0 \cdot 20 \frac{\text{дБ}}{\text{дек}} = 0 \frac{\text{дБ}}{\text{дек}}.$$

Далі визначаємо частоти, що сполучають ланки системи (табл. 4.10):

Таблиця 4.10– Частоти, які сполучають ланки

Ланки	T	$\omega = \frac{1}{T}$	$lg\omega$	Нахил, дБ/дек
$\frac{1}{(0.25s + 1)}$	0.25	4	0.6	$-20 \frac{дБ}{дек}$
$\frac{1}{(0.14s + 1)}$	0.14	7.14	0.85	$-20 \frac{дБ}{дек}$
$\frac{1}{(0.000225s^2 + 0.6s + 1)}$	0.015	66	1.82	$-40 \frac{дБ}{дек}$

Початковий нахил ЛАЧХ заданої системи проходить через точку A_1 з початковим нахилом 0 дБ/дек. Далі шукаємо частоти сполучення і від першого нахилу будуємо далі наступні (рис. 4.132).

При побудові ЛАЧХ бажаної системи виділяють три області: область низьких частот, область середніх частот і область високих частот. Вид ЛАЧХ в кожній з областей по різному впливає на якість системи. У області низьких частот вид ЛАЧХ визначає точність роботи системи в сталих режимах. Область середніх частот визначає динамічні властивості системи (швидкодія, коливальність). Вид ЛАЧХ в області високих частот практично мало впливає на якість системи.

Побудову ЛАЧХ починають з області середніх частот. Спочатку знаходять по заданих величинах σ_{max} і t_{per} частоту зрізу ω_z по табл. 4.11.

Таблиця 4.11 – Таблиця заданих значень

$\sigma_{max}, \%$	20
$t_p \cdot \omega_{зр} / \pi$	4
$L_1, дБ$	13,5
$\Delta\varphi, гр$	65

Для заданої САУ максимальне перерегулювання дорівнює $\sigma_{max} = 20\%$, а час регулювання $t_{per} = 2,5$ с. Тоді частота зрізу розраховується за формулою:

$$\omega_{зр} = \frac{4 * \pi}{t_{per}} = \frac{4 \cdot 3,14}{2,5} = 5,02$$

;

$$\lg(\omega_{зр}) = 0,7.$$

Далі на вісь абсцис наноситься частота зрізу $\omega_{зр}$ і через неї проводиться пряма лінія з нахилом -20 дБ/дек. Потім з точки перетину лінії нахилом мінус 20 дБ/дек і L_1 (згідно табл. 4.5) опускаємо перпендикуляр і знаходимо

$$\omega_2 = 0,031.$$

В області низьких частот ЛАЧХ бажаної системи будується наступним чином. На графік наноситься точка $A_2(0; 2 * L_1)$ і через неї проводиться пряма з нахилом -20 дБ/дек. Від точки, яку знайшли шляхом перетину лінії нахилу мінус 20 дБ/дек і L_1 , проводимо пряму лінію з нахилом -40 дБ/дек до перетину з низькочастотною прямою (яка проходить через точку A_2) ЛАЧХ бажаної системи. У точці перетину опускаємо перпендикуляр на вісь абсцис і знаходимо

$$\omega_1 = -0,756.$$

Частоту ω_5 , що обмежує область середніх частот праворуч, визначають величиною відрізка L_2 , при цьому $|L_2| \geq L_1$. Беремо $L_2 = 22,5$, що дає нам результат $\omega_5 = 1,82$.

У області високих частот бажану ЛАЧХ будують у вигляді прямолінійних відрізків з нахилом, кратним 20 дБ/дек. (тобто $40, 60, 80$ і т.д.), таким чином, щоб різниця характеристик бажаної і заданої в межі при $\omega \rightarrow \infty$ складала пряму лінію, паралельну осі частот.

В області високих частот ЛАЧХ бажаної системи ω_5 , тобто частоти яка йде перша від заданої системи і від неї проводимо пряму з нахилом -80 дБ/дек (рис. 4.132).

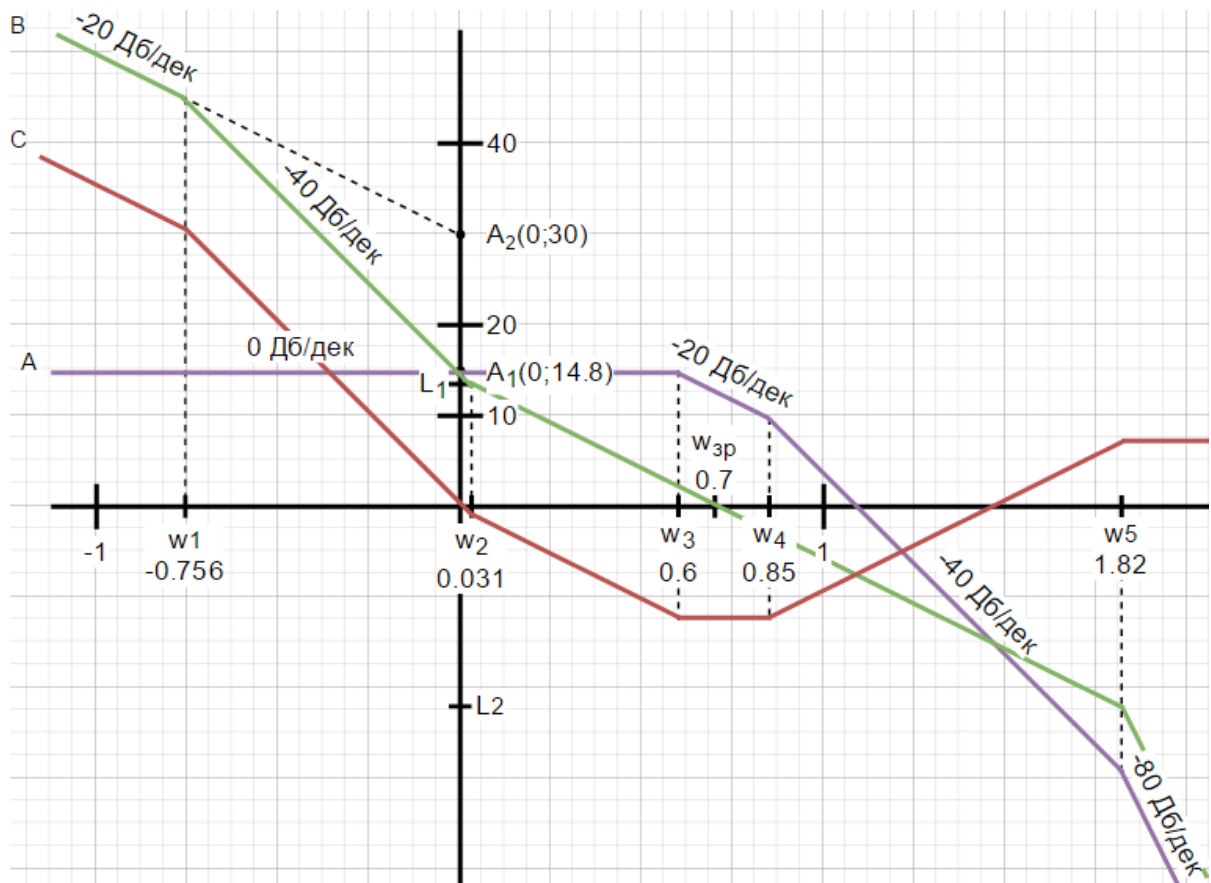


Рисунок 4.132 – Асимптотичні ЛАЧХ: А – заданої системи; В – бажаної системи; С – послідовної корегуючої ланки.

ЛАЧХ послідовної корегуючої ланки будується шляхом графічного вирахування з ЛАЧХ бажаної системи ЛАЧХ заданої системи (рис. 4.69).

Отримуємо:

$$\begin{aligned}
 -20 \frac{\text{дБ}}{\text{дек}} - 0 \frac{\text{дБ}}{\text{дек}} &= -20 \frac{\text{дБ}}{\text{дек}}; \\
 -40 \frac{\text{дБ}}{\text{дек}} - 0 \frac{\text{дБ}}{\text{дек}} &= -40 \frac{\text{дБ}}{\text{дек}}; \\
 -20 \frac{\text{дБ}}{\text{дек}} - 0 \frac{\text{дБ}}{\text{дек}} &= -20 \frac{\text{дБ}}{\text{дек}}; \\
 -20 \frac{\text{дБ}}{\text{дек}} - \left(-20 \frac{\text{дБ}}{\text{дек}}\right) &= 0 \frac{\text{дБ}}{\text{дек}}; \\
 -20 \frac{\text{дБ}}{\text{дек}} - \left(-40 \frac{\text{дБ}}{\text{дек}}\right) &= +20 \frac{\text{дБ}}{\text{дек}}; \\
 -80 \frac{\text{дБ}}{\text{дек}} - \left(-80 \frac{\text{дБ}}{\text{дек}}\right) &= 0 \frac{\text{дБ}}{\text{дек}}.
 \end{aligned}$$

Для початку визначається коефіцієнт підсилення корегувальної ланки:

$$20 \lg K = A_3, \text{ де } A_3 = A_2 - A_1 = 30 - 14.8 = 15.2;$$

$$20 \lg K = 15.2 \rightarrow K = 10^{\frac{15.2}{20}} = 5.75.$$

Далі визначасмо нахили прямих корегуючої ЛАЧХ та по ним знаходимо передатні функції (табл. 4.9).

Таблиця 4.12 – Передатні функції ЛАЧХ скорегованої системи

$\lg \omega$	$\omega = 10^{\lg \omega}$	$T = \frac{1}{\omega}$	$\frac{\text{дБ}}{\text{дек}}$	Ланка
-0,756	0,175	5,7	-20	$\frac{1}{(5,7s + 1)}$
0,031	1,07	0,93	+20	$0,93s + 1$
0,6	3,98	0,25	+20	$0,25s + 1$
0,85	7,07	0,14	+20	$0,14s + 1$
1,80	66,06	0,0015	-20	$\frac{1}{(0,015s + 1)}$

Отже, згідно з коефіцієнтом підсилення скорегованої системи та табл. 4.9 складемо передатну функцію корегуючої ланки:

$$W_{\text{кор}}(s) = \frac{5,75(0,93s + 1)(0,25s + 1)(0,14s + 1)}{s(5,7s + 1)(0,015s + 1)} =$$

$$= \frac{0,1872s^3 + 2,287s^2 + 7,59s + 5,75}{0,2727s^3 + 8,002s^2 + 8,59s + 5,75}$$

Після знаходження передатної функції послідовної корегуючої ланки, побудуємо структурну схему скорегованої системи (рис. 4.133)

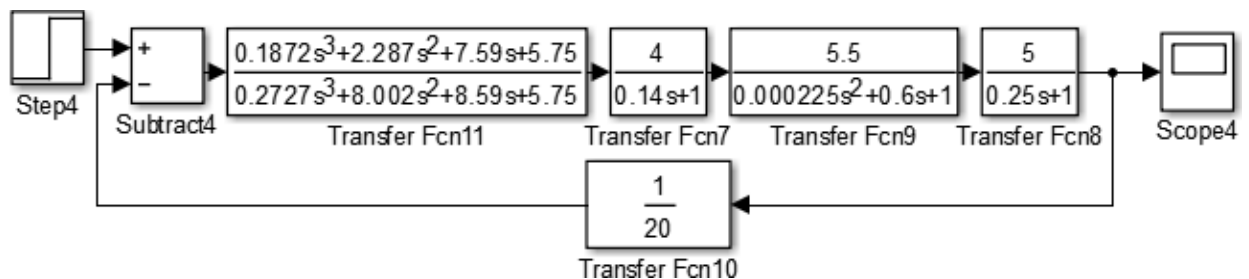


Рисунок 4.133 – Структурна схема скоригованої САУ

Враховуючи проведені розрахунки отримаємо наступні передатні функції:

$$W_{\text{роз}}(s) = \frac{110}{0.0001575s^4 + 0.421755s^3 + 5.3845s^2 + 19.8s + 20};$$

$$W_{\text{кор}}(s) = \frac{0.1872s^3 + 2.287s^2 + 7.59s + 5.75}{0.2727s^3 + 8.002s^2 + 8.59s + 5.75};$$

$$\begin{aligned} W_{\text{кор_роз}}(s) &= W_{\text{кор}} W_{\text{роз}} = \\ &= \frac{20.59s^3 + 251.5s^2 + 834.9s + 632.5}{10^{-5} * 4.294s^7 + 0.1163s^6 + 4.844s^5 + \dots + 361.1s^2 + 285.7s + 115} \end{aligned}$$

$$\begin{aligned} W_{\text{кор_замк}}(s) &= \frac{W_{\text{кор}} W_1 W_2 W_3}{1 + W_{\text{кор}} W_1 W_2 W_3 W_4} = \\ &= \frac{411.8s^3 + 5031s^2 + 16700s + 12650}{10^{-5} * 4.294s^7 + 0.1163s^6 + 4.844s^5 + \dots + 612.6s^2 + 1121s + 747.5}. \end{aligned}$$

Передатна функція замкнутої скорегованої системи за помилкою розраховується за формулою:

$$\begin{aligned} W_{eg}(s) &= \frac{1}{1 + W_{\text{кор_роз}}} = \\ &= \frac{10^{-5} * 4.294s^7 + 0.1163s^6 + 4.844s^5 + \dots + 361.1s^2 + 285.7s + 115}{10^{-5} * 4.294s^7 + 0.1163s^6 + 4.844s^5 + \dots + 612.6s^2 + 1121s + 747.5} \end{aligned}$$

Коефіцієнти помилок є числовими характеристиками якості системи в сталому режимі. Коефіцієнт C_{g0} називають коефіцієнтом позиційної помилки, коефіцієнт C_{g1} називають коефіцієнтом швидкісної помилки, коефіцієнт C_{g2} називають коефіцієнтом помилки по прискоренню. Розрахунки наведено нижче:

$$C_{g0} = W_{eg}(0) = 0.1538;$$

$$C_{g1} = \left. \frac{dW_{eg}(s)}{ds} \right|_{s=0} = 0.1288;$$

$$C_{g2} = \left. \frac{d^2 W_{eg}(s)}{ds^2} \right|_{s=0} = 0.03.$$

Для порівняння скоригованої системи с заданою, розрахуємо коефіцієнти помилок для заданої системи. Знайдемо задану замкнену скориговану систему за помилкою:

$$W_{eg}(s) = \frac{1}{1 + W_{\text{роз}}(s)} = \frac{0,0001575s^4 + 0,4218s^3 + 5,385s^2 + 19,8s + 20}{0,0001575s^4 + 0,4218s^3 + 5,385s^2 + 19,8s + 130};$$

та розрахуємо коефіцієнти помилок:

$$C_{g0} = W_{eg}(0) = 0,1538;$$

$$C_{g1} = \left. \frac{dW_{eg}(s)}{ds} \right|_{s=0} = 0,1514;$$

$$C_{g2} = \left. \frac{d^2 W_{eg}(s)}{ds^2} \right|_{s=0} = 0,25.$$

Коефіцієнти помилок заданої системи є дещо більшими.

Моделювання є останньою й однією з найважливіших задач аналізу систем автоматичного управління, що дозволяє імітувати поведінку реальної системи в різних умовах експлуатації, передбачити аварійні ситуації або підвищення навантаження на елементи системи без ризику їхнього руйнування. Моделювання перехідних процесів є обов'язковим етапом дослідження умовно стійких і нестійких систем. Воно замінює експерименти з реальними дорогими об'єктами, які в робочих умовах завжди повинні функціонувати стійко, надійно й безпечно.

Перехідні характеристики заданої та скорегованої системи зображені на рис. 4.134 та рис. 4.135 відповідно.

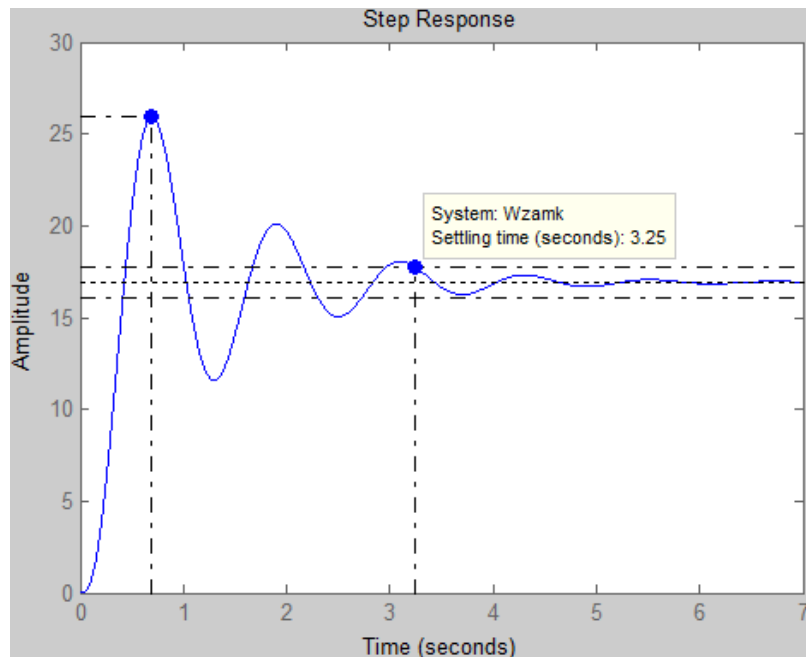


Рисунок 4.134 – Перехідна характеристика заданої системи

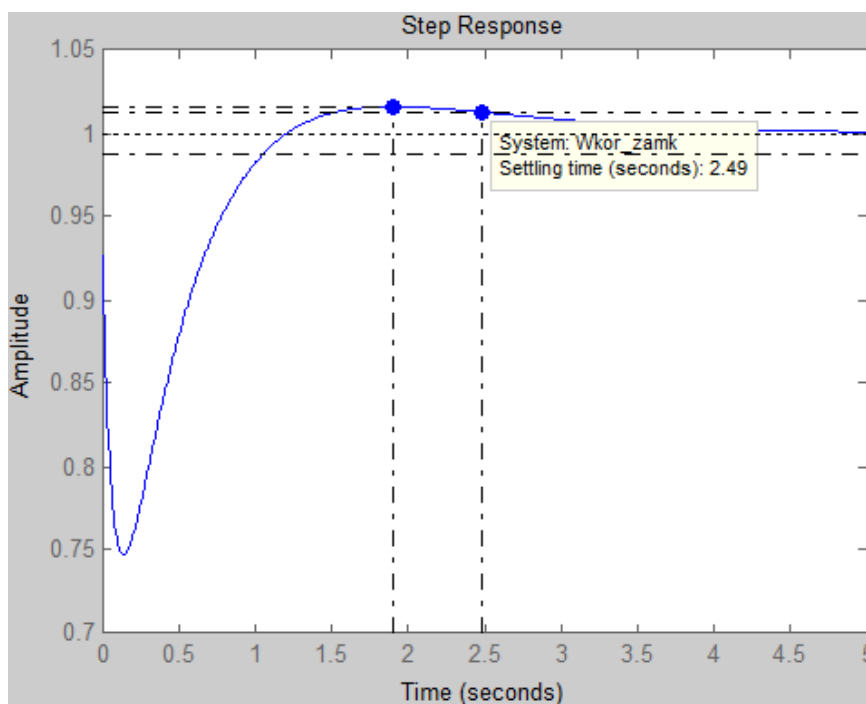


Рисунок 4.135 – Перехідна характеристика скорегованої системи

За видом перехідної характеристики визначимо прямі показники якості: час регулювання та перерегулювання та непрямі показники якості: запас стійкості по амплітуді та фазі, які належать до частотних показників якості.

Час регулювання t_p – це мінімальний час, по закінченні якого відхилення вихідної величини від сталого значення не перевищує певної заданої величини $\Delta = (0,05 \div 0,1)$, де h_∞ – стале значення перехідної характеристики. Для досліджуваної САУ задана величина $\Delta = 0.05$. Час регулювання визначається як

останній перетин перехідної характеристики з заданою величиною Δ . Час регулювання для перехідної характеристики скорегованої САУ дорівнює:

$$t_p = 2,49 \text{ с.}$$

Перерегулювання σ – максимальне відхилення перехідної характеристики від сталого значення. Перерегулювання знаходиться за формулою:

$$\sigma = \frac{h_{max} - h_{\infty}}{h_{\infty}} \cdot 100\%.$$

Перерегулювання для перехідної характеристики скорегованої САУ дорівнює:

$$\sigma = \frac{1.02 - 1.0}{1.0} \cdot 100\% = 2\%$$

Час регулювання та показник перерегулювання задовольняють заданим показникам.

Запас стійкості по амплітуді та фазі знаходяться за допомогою ЛАЧХ та ЛФЧХ. Запас стійкості за амплітудою визначається як відстань від ЛАЧХ до осі частот на частоті, на якій ЛФЧХ перетинає пряму $\varphi(\omega) = -\pi$. Запас стійкості за фазою визначається як відстань від ЛФЧХ до прямої $\varphi(\omega) = -\pi$ на частоті зрізу. Частота зрізу – це частота, на якій ЛАЧХ перетинає вісь частот.

На рис. 4.136 зображені ЛАЧХ та ЛФЧХ скорегованої розімкнутої САУ.

З отриманих графіків визначимо запаси стійкості досліджуваної за амплітудою ΔL та фазою $\Delta\varphi$:

$$\begin{aligned} \Delta L &= 17,8\text{дБ}; \\ \Delta\varphi &= 33,1^\circ. \end{aligned}$$

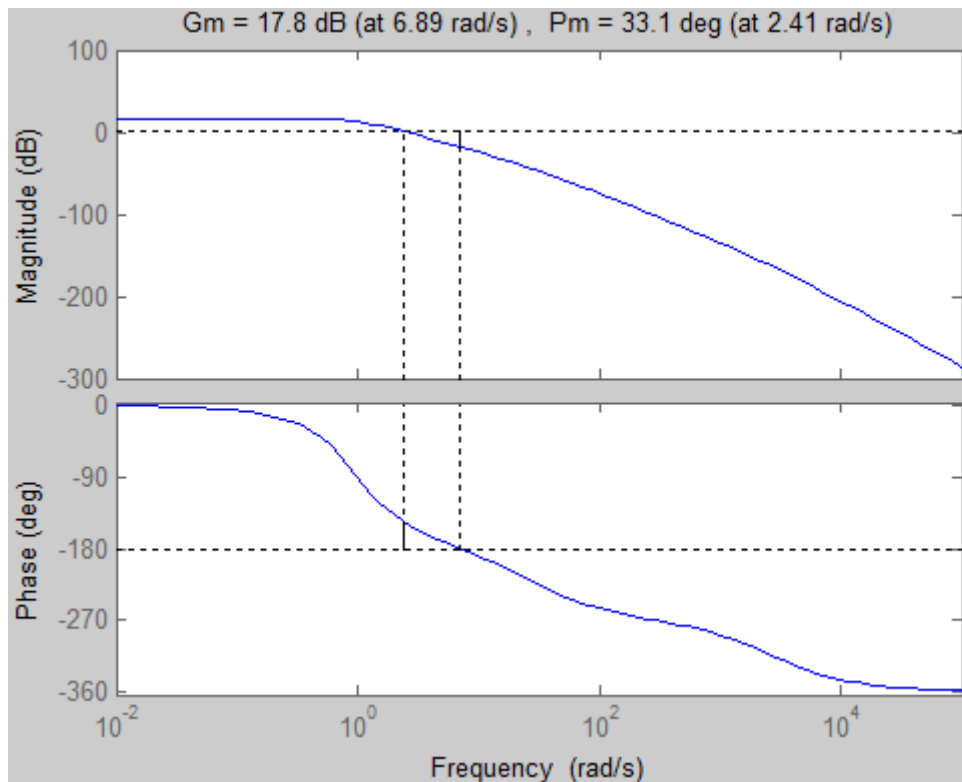


Рисунок 4.136 – ЛАЧХ та ЛФЧХ скоригованої системи

Запас стійкості по амплітуді та запас стійкості по фазі скоригованої системи збільшився від заданого. Тому можна зробити висновок, що дана система задовольняє заданим вимогам.

4.19 Розрахунки надійності розробленого модуля

Надійність розробленого модулю – це його властивість виконувати задані функції в певних умовах експлуатації при збереженні значень основних параметрів в заданих межах.

Надійність характеризується низкою розрахункових показників, найбільш важливими з яких є:

- інтенсивність відмов;
- середнє напрацювання на відмову;
- імовірність безвідмовної роботи.

Імовірність безвідмовної роботи вказує на те, яка частина виробів буде працювати безвідмовно протягом заданого часу t_p . Для більшості радіоелектронних пристроїв ймовірність безвідмовної роботи залежить як від фізичних властивостей, так і від часу t_p , протягом якого пристрій повинен працювати безвідмовно:

$$P(t) = e^{-\lambda \times t_p}, \quad (4.18)$$

де λ – інтенсивністю відмов, 10^{-6} 1/год;

t_p – часу, протягом якого пристрій повинен працювати безвідмовно, год.

Інтенсивністю відмов називають кількість відмов за одиницю часу, що припадає на один виріб, який продовжує працювати в даний момент часу:

$$\lambda = \frac{n}{N \cdot t_p} \quad (4.19)$$

Інтенсивність відмов апарата що, складається з N різних елементів, визначають за формулою:

$$\lambda = \lambda_1 + \lambda_2 + \lambda_3 + \dots + \lambda_N = \sum_{i=1}^N \lambda_i. \quad (4.20)$$

Розрахунок надійності проводимо в наступній послідовності.

Складаємо таблицю вихідних даних для розрахунку, визначаємо конструктивну характеристику компонентів, кількість компонентів по групах, розраховуємо інтенсивність відмов λ_i для кожної з груп компонентів.

Вихідні дані для розрахунку надійності зводимо в таблицю 4.13.

Для врахування умов експлуатації знаходимо поправочні коефіцієнти K_e , K_m , K_a і розраховуємо поправочний коефіцієнт.

Приймаємо $K_e = 1,0$, $K_m = 1,0$, $K_a = 1,0$.

$$K_\lambda = K_e \cdot K_m \cdot K_a, \quad (4.21)$$

$$K_\lambda = 1,0 \cdot 1,0 \cdot 1,0 = 1,0$$

Розрахунок інтенсивності відмов проводимо по формулі:

$$\lambda = (\lambda_1 + \lambda_2 + \lambda_3 + \dots + \lambda_m) \cdot K_\lambda = K_\lambda \cdot \sum_{i=1}^m \lambda_i, \quad (4.22)$$

$$\lambda = 0,436 \cdot 10^{-6} \text{ 1/год.}$$

Таблиця 4.13 – Вихідні дані розрахунку надійності

Типи елементів	Кількість елементів	Допустимі значення				Інтенсивність відмов, 10^{-6} 1/год
		Т, °С	Відносна вологість %	Механічні взаємодії		
				Вібрація, Гц	Удари, g	
Arduino nano	1	-10...+50	90	1-4000	40	0,019
TFT IPS 240x240	1	0...+50	90	1-2000	20	0,19
RS-485	1	0...+50	85	1-4000	40	0,023
Кнопки	3	0...+50	85	1-2000	20	0,16
Резистори	11	-20...+60	90	1-4000	40	0,044
Всього	17					0,436

Середнє напрацювання на відмову розраховуємо за формулою:

$$T_{cp.} = \frac{1}{\lambda}, \quad (4.23)$$

$$T_{cp} = 1 / 0,436 \cdot 10^{-6} = 2293578 \text{ год.}$$

Проводимо розрахунок імовірності безвідмовної роботи пристрою за формулою:

$$P(t) = e^{-\lambda \cdot t_p}, \quad (4.24)$$

де e – основа натурального логарифма;

λ – інтенсивність відмов;

t_p – час випробувань.

Результати розрахунків імовірності безвідмовної роботи пристрою записуємо в таблицю 4.14.

Таблиця 4.14 – Значення щільності розподілу напрацювання між відмовами та показників надійності в різних точках часу напрацювання

Час напрацювання t, год	Ймовірність безвідмовної роботи P(t)	Ймовірність відмови Q(t)
10	0,99985947	0,00014053
100	0,998595587	0,001404413
1000	0,986044296	0,013955704
10000	0,868888909	0,131111091
100000	0,245268929	0,754731071
1000000	7,87817E-07	0,999999212

4.20 Зміст підрозділу з охорони праці

Основна мета підрозділу «Охорона праці» (ОП) в кваліфікаційних роботах бакалавра – виявлення чинників, що спричиняють можливість виробничого травматизму, професійних захворювань, отруєнь, пожеж, вибухів, забруднення навколишнього середовища промисловими викидами під час експлуатації, виготовлення і застосування радіоелектронного устаткування на робочих місцях і в робочих приміщеннях, та розробка заходів, щодо техніки безпеки.

Зміст підрозділу має відповідати темі кваліфікаційної роботи бакалавра і передбачати розробку одного або кількох конкретних питань з техніки безпеки, промислової санітарії, пожежної профілактики, захисту навколишнього середовища. Загальний обсяг підрозділу ОП має складати 2...10 аркушів.

4.20.1 Виробнича санітарія в лабораторії

Робота з розробки модуля відображення інформації та програмного забезпечення для нього, проводитиметься у приміщенні лабораторії, робота в якій згідно до ДСН 3.3.6.042-99 за енерговитратами організму людини «легка 1б» – це роботи, які виконуються сидячи та не вимагають постійного фізичного напруження і переміщення важких речей. Енерговитрати для цієї категорії складають 121...150 ккал/год.

Для забезпечення нормативних показників кліматичних параметрів у приміщенні лабораторії в холодну пору року застосовують опалення, а у теплу – кондиціонування приміщення.

Роботи в лабораторії відносяться до III розряду високої точності, категорія

«г» згідно до СНиП-4-79. Мінімальне допустиме значення освітленості для даних робіт становить 200-400 лк.

Рівень загального штучного освітлення приміщення лабораторії можна перевірити за допомогою методу питомої потужності, розрахунки згідно якого виконуються як розрахунок питомої потужності за формулою

$$W = \frac{W_{\Sigma}}{S}, \quad (4.25)$$

де S – площа приміщення, м^2 ;

W_{Σ} – загальна потужність світлової установки, Вт.

$$W_{\Sigma} = W_{cv} \cdot n_{cv}, \quad (4.26)$$

де W_{cv} – потужність одного світильника, Вт;

n_{cv} – кількість світильників у приміщенні.

В лабораторії, де плануються проводитись роботи, розташовано 10 світильників номінальною потужністю 375 Вт кожний. Загальна площа приміщення складає 84 м^2 . Виходячи з цього отримаємо:

$$W_{\Sigma} = 375 \cdot 10 = 3750 \text{ Вт}, \quad (4.27)$$

$$W = \frac{W_{\Sigma}}{S} = \frac{3750}{84} = 44,6 \text{ Вт/м}. \quad (4.28)$$

Нормативне значення номінальної потужності для заданих умов складає 42 Вт/м^2 . Отримане значення є задовільним для рівня загального штучного освітлення приміщення лабораторії.

4.20.2 Пожежна безпека в приміщенні лабораторії

Згідно ДБН В.1.1.7-2002, будівля, в якій знаходиться лабораторія, має II ступінь вогнестійкості, так як дане приміщення виконано з цегли із застосуванням твердих матеріалів, що можуть бути спалені. За пожежонебезпекою приміщення, згідно СНиП 2.09.02-85, відноситься до категорії В. Приміщення має пожежонебезпечні зони, отже відноситься до класу П-Па згідно ПУЕ-2011.

До основного обладнання відносить сам макет, 6 комп'ютерів та макет промислового маніпулятора. Для зменшення небезпеки займання від кабелів живлення їх покривають ізоляційним вогнезахисними покриттями.

Захист проти пожежі досягається застосуванням первинних засобів пожежогасіння. Згідно ДНАОП 0.00-1.31-99 у приміщенні лабораторії розміщується два вуглекислотні вогнегасники ВВКОВ-2 із розрахунку один вогнегасник на три комп'ютери, телефон, встановлений в легкодоступному місці, два пожежних димових оптичних сповіщувача із розрахунку один на 100 м², але не менше двох на одне приміщення, автоматична пожежна сигналізація, яка реагує на появу диму, протипожежне покривало та ящик з піском з об'ємом 0,3 м³.

Відповідно до ГОСТ 12.1.004-91 ССБТ для запобігання пожежі передбачено:

- не поєднувати системи кондиціонування лабораторії та інші приміщення;
- застосовувати загальні або місцеві протипожежні перешкоди;
- навчати групу студентів і обслуговуючий персонал лабораторії протипожежним правилам.

Організаційні заходи включають в себе проведення інструктажів з пожежної безпеки, призначення відповідального за пожежну безпеку, нагляд за засобами пожежогасіння, видання необхідних інструкцій, планів евакуації, наявність індивідуальних засобів захисту.

4.20.3 Охорона праці користувачів персонального комп'ютера

Робота користувача із програмним модулем управління безпроводним обладнанням має на увазі систематичну і тривалу роботу з програмним забезпеченням. При цьому мається на увазі перегляд виведеної на екран інформації і її зміна стандартними пристроями введення (миша, клавіатура). Виведення інформації проводиться на екран монітора в текстовій і графічній формах. На підставі сказаного можна зробити висновок про те, що в процесі користування сайтом, користувач проводить весь час за ПК. При цьому на його органи зору впливає випромінювання монітора, використання клавіатури і миші впливає на суглоби рук, шуми монітора підвищують дратівливість людини і впливають на його центральну нервову систему. В результаті можна зробити наступний висновок: користувач сайту піддається всім шкідливим впливам, типовим для стандартного користувача ПК.

Користувач ПК піддається шкідливому впливу, як з боку техніки, так і через неправильно організованого робочого місця і відчуває на собі такі небезпечні фактори:

- вплив електромагнітного і електростатичних полів (від монітора, системного блоку, пристроїв введення/виведення) на внутрішні органи і на деформацію живих клітин;
- випромінювання (рентгенівський, ультрафіолетовий та інфрачервоний спектри);
- вплив шумів і вібрацій на організм;
- вплив ергономіки робочого місця і дизайну зовнішніх пристроїв комп'ютера (клавіатура, миша) на опорно-рухову систему і м'язовий тонус;
- напруга зору (умови освітлення, контраст монітора);
- небезпека ураження електричним струмом;
- нервово-емоційні перевантаження.

Вплив зазначених несприятливих факторів призводить до порушень центральної нервової, серцево-судинної та ендокринної систем, нейротрофічних порушень і патологічних змін, включаючи зміни складу крові, дратівливості, головних болів, погіршення слуху, запаморочення, зниження пам'яті, підвищеної стомлюваності. У користувачів знижується концентрація уваги, швидко настає втома в зв'язку з підвищеними енергетичними витратами і нервово-психічним напруженням. Все це веде до зниження працездатності, продуктивності, якості та безпеки праці.

Тривале перебування людини в зоні комбінованого впливу різних несприятливих факторів може привести до професійного захворювання.

Дотримання вимог до робочого обладнання, аксесуарів і характеру роботи, допоможе усунути шкідливі фактори, що впливають на користувача:

Оскільки існує кілька видів шкідливих впливів, існує і кілька шляхів їх запобігання.

Для захисту від електромагнітних і електростатичних полів допускається застосування екранних фільтрів, спеціальних екранів та інших засобів індивідуального захисту, що пройшли випробування і мають відповідний гігієнічний сертифікат.

При використанні захисних фільтрів, одягаються або вбудовуються в корпус монітора, вони обов'язково повинні бути заземлені.

З метою попередження захворювань користувачів, викликаних впливом випромінювання при роботі на ПК, рекомендується застосовувати монітори зі зниженим рівнем випромінювання, що відповідають міжнародним стандартам

MPR-II, TCO'95, TCO'99, і контролювати дотримання працюючими регламентованих режимів праці та відпочинку.

Для зменшення шуму в приміщенні з ПК, як правило, застосовують метод акустичної обробки приміщень, використовуючи для облицювання огорожувальних поверхонь звукопоглинальні матеріали з максимальними коефіцієнтами звукопоглинання (α) в інтервалі частот 63 Гц – 8000 Гц. Із цією метою на стелях і стінах розміщують перфоровані панелі з звукопоглинальним наповнювачем. Панелі зміцнюють безпосередньо на поверхні огорожі або з віднесенням від нього на відстань 20 см. В останньому випадку застосування звукопоглинальної облицювання більш ефективно.

Додатковим звукопоглинанням можуть служити однотонні завіси з щільної тканини, що гармоніюють з забарвленням стін і підвішені в складку на відстані 15...20 см від віконного скла. Ширина завіси повинна бути в два рази більше ширини вікна. Знизити рівень шуму можна також, використовуючи для друку малошумні лазерні принтери.

Заходи, що ведуть до усунення нервово-емоційних перевантажень:

- використання зручного і вдосконаленого інтерфейсу. Не допущення використання яскравих контрастних зображень і різку їх зміну, тобто використання в міру можливості набір півтонів;

- виключення присутності на екрані несистематизованої інформації у великій кількості;

- користувачам необхідно забезпечити обчислювальні засоби достатньої швидкодії. Час реакції на запит не повинен перевищувати в середньому 5...10 секунд.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Про вищу освіту [Електронний ресурс] : Закон України від 01.07.2014 р. № 1556-VII // Верховна Рада України : офіційний веб-портал. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/1556-18#Text>. – Станом на 10.05.2023. – Назва з екрану.

2. Положення про організацію освітнього процесу в ХНУРЕ [Електронний ресурс] : наказ ХНУРЕ 19.04.2023 р. № 74. // Нормативно-правова база ХНУРЕ : офіційний веб-портал. – Режим доступу: https://nure.ua/wp-content/uploads/Main_Docs_NURE/polozhennja-pro-organizaciju-osvitnogo-procesu-v-hnure-2023.pdf. – Станом на 10.05.2023. – Назва з екрану.

3. Положення про протидію академічному плагіату в Харківському національному університеті радіоелектроніки [Електронний ресурс] : наказ ректора ХНУРЕ від 28.04.2017 р. № 290 // Нормативно-правова база ХНУРЕ : офіційний веб-портал. – Режим доступу: <https://nure.ua/universytet/normativno-pravova-baza#id13>. – Станом на 18.09.2021. – Назва з екрану.

4. Положення про порядок створення та організацію роботи екзаменаційних комісій з атестації здобувачів вищої освіти ступенів бакалавр, магістр (спеціаліст) [Електронний ресурс] : наказ ректора ХНУРЕ від 09.02.2015 р. № 40 // Нормативно-правова база ХНУРЕ : офіційний веб-портал. – Режим доступу: <https://nure.ua/universytet/normativno-pravova-baza#id13>. – Станом на 18.09.2021. – Назва з екрану.

5. Інформація та документація. Звіти у сфері науки і техніки. Структура і правила оформлювання. – [На заміну ДСТУ 3008-95 ; чинний від 2017-07-01]. – К. : ДП «УкрНДНЦ», 2016. – 31 с.

6. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання (ГОСТ 7.1-2003, IDT) : ДСТУ ГОСТ 7.1:2006. – [Чинний від 2007-01-01]. – К. : Держспоживстандарт України, 2007. – 57 с.

7. Стандарт вищої освіти за спеціальністю 151 «Автоматизація та комп'ютерно-інтегровані технології» галузі знань 15 «Автоматизація та приладобудування» для першого (бакалаврського) рівня вищої освіти [Електронний ресурс] : наказ Міністерства освіти і науки України від 04.10.2018 р. № 1071 // Міністерство освіти і науки України : офіційний веб-портал. – Станом на 18.09.2021. – Режим доступу: <https://mon.gov.ua>.

8. Методичні вказівки з підготовки кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти денної і заочної форми навчання

спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Системна інженерія» / Упоряд.: І.Ш. Невлюдов, О.М. Цимбал, О.В. Токарева, А.І. Бронніков. – Харків: ХНУРЕ, 2023. – 64 с.

9. Дипломне проектування для студентів усіх форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»: навч. посібник / І. Ш. Невлюдов, А. О. Андрусевич, О. В. Токарева, Г. В. Пономарьова. – Київ-58, пр. Космонавта Комарова, 1, 2016. – 245 с.

10. Методичні вказівки до виконання дипломних проектів (робіт) для студентів напряму підготовки 6.051003 «Приладобудування», професійного спрямування «Прилади і системи орієнтації та навігації»; 151 – Автоматизація та комп'ютерно-інтегровані технології, спеціалізації «Комп'ютерно-інтегровані технології і системи навігації та керування»; освітньо-кваліфікаційного рівня бакалавр, денної форми навчання, електронне видання / Укл.: П. М. Бондар, В. В. Мелешко. – К.: НТУУ «КПІ», 2016. – 36 с.

11. Виробничі процеси та обладнання об'єктів автоматизації. Збірник задач : навч. посіб. / І. Ш. Невлюдов, А. О. Андрусевич, Г. В. Пономарьова, А. О. Функендорф ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Кривий Ріг : КК НАУ, 2018. – 332 с.

12. Цимбал О.М.Бронніков А.І. Системи адаптації роботів і технологія OpenCV: Навчальний посібник – Харків: ХНУРЕ, 2019. – 148 с.: іл. – ISBN 978-966-659-268-5.

13. Оформлення технічної документації : навч. посіб. / В. В. Семенець, І. Ш. Невлюдов, А. М. Сінотін, С. В. Сотник ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2021. – 148 с. – ISBN 978-966-659-302-6 ; DOI: 10.30837/978-966-659-302-6.

14. Models and methods of robots adaptive visual control / I. Nevliudov, A. Bronnikov, O. Tsymbal, A. Andrusevitch, V. Gopejenko – Riga, ISMA, 2022. – 164 p.

15. Ковальов Ю.А., Кошель С.О., Манойленко О.П. Проектування промислових роботів та маніпуляторів. «Центр учбової літератури», 2020. – 256 с.

16. Ткачук В. Основи мехатроніки. Конспект лекцій. «Магнолія 2006», 2022. – 266 с.

17. Дмитрів В.Т., Ланець О.С. Динаміка і точність роботів. Львівська політехніка, 2021. – 200 с.

18. Васильківський І.С., Фединець В.О. Виконавчі пристрої систем автоматизації. Львівська політехніка, 2020. – 220 с.

19. Автоматизація виробничих процесів : підруч. / І. В. Ельперін, О. М. Пупена, В. М. Сідлецький, С. М. Швед ; М-во освіти і науки України, Нац.

ун-т харчових технологій. – 2-е вид., випр. – Київ : Ліра-К, 2016. – 378 с. : іл. – ISBN 978-966-2609-81-3.

20. Невлюдов, І.Ш. Теорія автоматичного управління (збірник задач) [Текст]: навчальний посібник / І.Ш. Невлюдов, О.В.Токарева. – Харків: ХНУРЕ, 2020. – 240 с.

21. Невлюдов І.Ш. Пневматичні пристрої та засоби автоматизації мехатронних систем: Навчальний посібник / І.Ш. Невлюдов, Л.О. Кривопляс-Володіна, С.П. Новоселов, О.В. Сичова. – Харків: ФОП Панов А.М., 2020 . – 256 с. DOI: 10.30837/978-617-7859-58-0. ISBN 978-617-7859-58-0.

22. Невлюдов І.Ш. Електропневмоавтоматичні приводи в автоматизованих системах керування: Навчальний посібник / І.Ш. Невлюдов, Л.О. Кривопляс-Володіна, С.П. Новоселов, О.В. Сичова. – Харків: ХНУРЕ, 2021 . – 292 с. DOI: 10.30837/978-966-659-332-3. ISBN 978-966-659-332-3.

23. Невлюдов І. Ш. Механізми технічних засобів автоматизації. Довідкові матеріали з курсового і дипломного проектування : навч. посібник / І. Ш. Невлюдов, В. І. Роменський, І. О. Яшков ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2021. – 292 с. – ISBN ISBN 978-966-659-301-9.

24. Невлюдов І. Ш. Управління якістю виробів : навч. посіб. для студ. вищих навч. закладів / І. Ш. Невлюдов, Д. А. Янушкевич, Л. С. Іванов ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2021. – 364 с. – ISBN 978-966-659-333-0.

25. Нефьодов Л. І. CALS-технології і системи : навч. посіб. / Л. І. Нефьодов, І. Ш. Невлюдов, В. В. Безкоровайний ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2021. – 272 с. – ISBN 978-966-659-326-2.

26. Конспект лекцій з дисципліни "Технічні засоби автоматизації" для студентів усіх форм навчання спеціальності 151 - Автоматизація та комп'ютерно-інтегровані технології" спеціалізації "Автоматизація та комп'ютерно-інтегровані технології"спеціалізації. Ч. 1 / упоряд.: В. І. Роменський, Н. П. Демська; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2018. – 100 с. : іл.

27. Конспект лекцій з дисципліни «Технічні засоби автоматизації» Ч.2: «Електронні компоненти засобів автоматизації» для студентів усіх форм навчання спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології, освітньо-проф [Електронний ресурс] / Упоряд. : В. Ю. Грицюк ; М-во освіти і науки України, ХНУРЕ. – Електрон. вид. – Харків : ХНУРЕ, 2021.

28. Невлюдов І. Ш. Автоматичне управління технологічними об'єктами [Електронний ресурс] : Підручник / І. Ш. Невлюдов, О. В. Токарева ; М-во освіти і науки України, ХНУРЕ. – Харків : ХНУРЕ, 2018. – 119 с.

29. Невлюдов І.Ш. Автоматизована система керування технологічними процесами в SCADA системі TRACE MODE 6: Навчальний посібник / І.Ш. Невлюдов, А.О. Андрусевич, В.В. Євсєєв, С.С. Максимова, М.Г. Стародубцев, В.В.Невлюдова. Кривий Ріг: Криворізький коледж НАУ, 2018. 320 с.

30. Моделі та методи кіберфізичних виробничих систем в концепції Industry 4.0 : монографія / І. Ш. Невлюдов, В. В. Євсєєв, А. О. Андрусевич, С. С. Максимова ; – Oktan Print – Prague. 2023. – 321 с.

31. Невлюдов І.Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Частина 1: Підручник Харків: ФОП Панов А.М., 2021., 604 с. ISBN 978-617-7947-67-6.

32. Невлюдов І.Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Частина 2: Підручник Кривий Ріг: видавець Чернявський Д.О., 2022 – 424 с.з іл. ISBN 978-617-8045-44-9.

33. Ладанюк А.П., Луцька Н.М., Власенко Л.О. , Кишенько В.Д., Іва-щук В.В. Методи сучасної теорії управління. Підручник. Ліра К, 2018. – 368 с.

34. Булгакова О.С., Зосімов В.В., Поздєєв В.О. Методи та системи штучного інтелекту: теорія та практика. Олді-плюс, 2020. – 356 с.

35. Nevliudov, I., Tsymbal, O., Bronnikov, A. (2022). Fuzzy Decision-Making for Intelligent Robotic System. In: Sergiyenko, O. (eds) Optoelectronic Devices in Robotic Systems. Springer, Cham. https://doi.org/10.1007/978-3-031-09791-1_9

ДОДАТОК А

Зразок бланка завдання на кваліфікаційну роботу бакалавра

Харківський національний університет радіоелектроніки

Факультет _____

Кафедра _____

Рівень вищої освіти _____ перший (бакалаврський) _____

Спеціальність _____

(код і повна назва)

Тип програми _____ освітньо-професійна _____

Освітня програма _____

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _____

(прізвище, ім'я, по батькові)

1. Тема роботи _____

затверджена наказом університету від _____ 20 ____ р. № _____

2. Термін подання студентом роботи до екзаменаційної комісії _____ 20 ____ р.

3. Вихідні дані до роботи _____

4. Перелік питань, що потрібно опрацювати в роботі _____

ДОДАТОК Б

Форма заяви щодо самостійності виконання письмової роботи

ЗАЯВА

щодо самостійності виконання письмової роботи

Я, _____
(прізвище, ім'я, по батькові)

посада _____

кафедра _____

заявляю: моя письмова робота на тему _____

(назва роботи)

представлена у _____

(спеціалізовану вчену раду, екзаменаційну комісію тощо)

для публічного захисту, виконана самостійно і в ній не міститься елементів плагіату. Всі запозичення з друкованих та електронних джерел, а також із раніше виконаних дослідницьких робіт та захищених кандидатських і докторських дисертацій мають відповідні посилання. Я ознайомлений (а) з діючим положенням «Про протидію плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску письмової роботи до захисту та застосування дисциплінарних заходів.

Дата

Підпис

ДОДАТОК В

Форма експертного висновку результатів перевірки на унікальність
тексту

Експертний висновок
результатів перевірки на унікальність тексту
в мережі Інтернет (базі ХНУРЕ)

Відповідно до даних програми _____
(назва електронної антиплагіатної програми)

письмової роботи _____

_____ ,
(вид та назва роботи)

автора (ів): _____

містить _____ % авторського тексту.

Експерт _____
(підпис)

(Прізвище, ініціали)

Дата

ДОДАТОК Г

Форма відгуку керівника роботи

Відгук

керівника кваліфікаційної роботи бакалавра

Студента _____
(прізвище, ім'я, по батькові, група)

(номер та назва спеціальності, освітньої програми)

Тема випускної роботи бакалавра _____

Відгук складається у довільній формі і має містити дані, які наведені в даних методичних вказівках.

Керівник кваліфікаційної роботи бакалавра

(прізвище, ініціали)

(місце роботи, посада)

(підпис)

« ___ » _____ 20__ р.

ДОДАТОК Д

Форма рецензії на роботу

Рецензія

На роботу студента гр. _____
(шифр групи)

_____ (прізвище, ім'я, по батькові)

Спеціальність, освітня програма _____
(номер, назва)

Тема роботи _____

Структура роботи:

- пояснювальна записка _____ сторінок;
- графічна частина _____ аркушів.

Рецензія складається у довільній формі з відображенням вимог до рецензії на роботу, які наведені в даних методичних вказівках.

Рецензент _____
(прізвище, ініціали, посада, місце роботи)

« ___ » _____ 20__ р. _____
(підпис)

ДОДАТОК Е

Критерії оцінювання результатів виконання і захисту кваліфікаційної роботи

№	Критерії	Макс. кількість балів	Зміст критеріїв оцінювання	Оцінка в балах
1	Актуальність теми, її відповідність сучасним вимогам	5	відповідає повністю	5
			відповідає неповністю	3
			відповідає недостатньо	1
			відповідність відсутня	0
2	Повнота, науковий рівень обґрунтування розробок та запропонованих рішень відповідно до завдання	25	повно та обґрунтовано	26-16
			недостатньо	15-6
			неповно та недостатньо	5
3	Практична цінність розробок та запропонованих рішень	10	висока практична цінність	10-6
			практична цінність часткова	5-4
			окремі елементи мають практичну цінність	3-2
			не має практичної цінності	0
4	Відповідність роботи та її оформлення нормативним актам України, національним стандартам	10	достатньо повна, висока якість	10-6
			недостатньо повна, прийнятна якість	5-4
			достатньо повна, висока якість	3-2
			недостатньо повна, невисока якість	0
5	Змістовність доповіді та відповідей на запитання членів ЕК під час захисту	50	повні, послідовні, логічні	50-31
			недостатньо повні, послідовні, логічні	30-11
			непослідовно та нелогічно побудовані доповідь, неповні відповіді на запитання	10-1
			відсутні або незадовільні відповіді на питання	0
Разом		100		

ДОДАТОК Ж

Зразок бланка титульного аркуша пояснювальної записки кваліфікаційної роботи бакалавра

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____
(повна назва)

Кафедра _____
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти _____ перший (бакалаврський) _____

(тема)

Виконав:
студент _____ курсу, групи _____

(прізвище, ініціали)

Спеціальність _____

(код і повна назва спеціальності)

Тип програми _____ освітньо-професійна

Освітня програма _____

(повна назва освітньої програми)

Керівник _____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ (підпис) _____ (прізвище, ініціали)

20__ р.

ДОДАТОК И

Приклад оформлення реферату

РЕФЕРАТ

Пояснювальна записка містить: 74 с., 12 табл., 41 рис., 2 дод., 16 джерел.

NODE MCU V3 LOLIN, СЕНСОР ВОЛОГОСТІ І ТЕМПЕРАТУРИ ПОВІТРЯ, СЕНСОР ВОЛОГОСТІ ҐРУНТУ, МОДУЛЬ ФОТОРЕЗИСТОРА, РОЗУМНА ТЕПЛИЦЯ.

Мета роботи – розробити і реалізувати автоматизований модуль, який може автоматично підтримувати середовище теплиці в заданому кліматичному режимі; реалізувати програмний засіб для відображення стану системи в реальному часі у мережі WiFi через смартфон.

Об'єкт дослідження – процес керування мікрокліматом для макету теплиці.

Предмет дослідження – тепличний контролер управління мікрокліматом на базі платформи Node MCU v3 Lolin.

У роботі проведено аналіз існуючих рішень автоматизації виробничого комплексу тепличного господарства.

Розроблено програмний засіб віддаленого контролю мікрокліматом теплиці, який дозволяє здійснювати управління основними факторами, що сприяють зростанню рослин, зокрема, такими як освітлення і вологість ґрунту, і його температура. Також детально досліджена платформа Node MCU, її технічні можливості і особливості програмування вбудованого мікроконтролера Для розробки системи, був обраний контролер Node MCU v3 Lolin збудований на ESP (8266) 12E.

За результатами роботи опубліковано статтю у науково-технічному журналі та тези доповіді у збірнику міжнародної конференції.

ДОДАТОК К

Приклади записів у переліку посилань згідно з ДСТУ ГОСТ 7.1:2006

Книга одного автора:

Азаренкова Г. М. Аналіз моделювання і управління ризиком (в схемах та прикладах) : навч. посіб. / Г. М. Азаренкова ; Нац. банк України, Ун-т банківської справи, Харків. ін-т банківської справи. – Львів : Новий Світ-2000, 2017. – 240 с. : іл. – ISBN 978-966-418-132-4.

Книга двох авторів:

Бичков О. С. Об'єктно-орієнтоване програмування мовою C# : підруч. / О. С. Бичков, Є. В. Іванов ; М-во освіти і науки України, Київ. нац. ун-т ім. Т. Шевченка. – Київ : Київ. ун-т, 2018. – 207 с. – ISBN 978-966-439-965-1.

Книга трьох авторів:

Невлюдов І. Ш. Технологія програмування промислових контролерів в інтегрованому середовищі CODESYS : навч. посіб. / І. Ш. Невлюдов, С. П. Новоселов, О. В. Сичова ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2019. – 264 с. : іл. – ISBN 978-966-659-265-4.

Книга чотирьох авторів:

Основи наукових досліджень : [навч. посіб.] / І. Ш. Невлюдов, Ю. М. Олександров, А. О. Андрусевич, О. О. Чала ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Кривий Ріг : КК НАУ, 2017. – 344 с.

Книга п'яти авторів і більше:

Промисловий Ethernet для MODBUS. Модуль 2 : навчальні матеріали для курсу Промислові мережі / В. Воропаєва, О. Вовна, В. Тарасюк и др. ; М-во освіти та науки України, Донец. нац. техн. ун-т. – Одеса : ФОП Побута М. І., 2017. – 68 с.

Перекладне видання:

Мартін Роберт. Чистий Agile: назад до основ / пер. з англ. В. Луненко. — Харків : Вид-во «Ранок» : Фабула, 2021. — 224 с. — ISBN 978-617-09-6760-2.

Книги із назвою:

Басюк Т. М. Основи інформаційних технологій : навч. посіб. / Т. М. Басюк, Н. О. Думанський, О. В. Пасічник ; за наук. ред. В. В. Пасічника ; М-во освіти і науки України. – Львів : Новий Світ-2000, 2017. – 390 с. : іл. – (Комп'ютинг). – ISBN 978-966-418-121-8.

Статі з журналів:

Филипенко, О.І. Моделювання впливу структури фотонно-кристалічних волокон на розподіл модового поля та втрати оптичного сигналу в їх з'єднаннях / О.І. Филипенко, О.В. Сичова // Радіотехніка. – Харків: ХНУРЕ. – 2012. – Вип. 171. – С. 327-331.

Збірники наукових праць:

Сучасні проблеми розвитку права та економіки в інноваційному суспільстві: зб. наук. праць за матеріалами Міжнародної науково-практичної конференції, 20 березня 2020 р. / за ред. М. Петрової, С. Глібка. – Велико-Тирново, Болгарія : ACCESS PRESS. – 2020. – 278 с.

Тези конференцій:

Підвищення надійності машин і обладнання : зб. тез доп. VII Всеукр. наук.-практ. конф. студентів та аспірантів – Кіровоград: КНТУ, 2013. – 215 с.

Матеріали конференцій:

Радіоелектроніка та молодь у XXI столітті : матеріали XXIV Міжнар. молодіж. форуму, 7-9 квіт. 2020 р. Т. 2. Конференція "Автоматизовані системи та комп'ютеризовані технології радіоелектронного приладобудування" / М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2020. – 188 с.

Стандарти, техніко-економічні та технічні документи:

ДСТУ 4826 : 2007. Інформація та документація. Видання інформаційні. Загальні вимоги (ISO 8:1977, NEQ ; ISO 1086:1991, NEQ) / Національний стандарт України. – уведено вперше 2009-01-01 (зі скасуванням в Україні ГОСТ 7.23-80. – Київ : Держспоживстандарт України, 2008. – 10 с.

Авторські свідоцтва, патенти:

МЕМС-інтерфейс багатоточкових автоматичних контролюючих комплексів [Електронний ресурс] : пат. 98539 Україна / Жарікова І. В., Костенко З. І., Невлюдов І. Ш., Палагін В. А., Разумов-Фризюк Є. А. ; ХНУРЕ, 2006. – № а 2010 11390 ; заявл. 24.09.2010 ; опубл. 25.02.2011, Бюл.№ 4. – 6 с.

Звіти про НДР:

Теоретичні основи створення перспективних компонентів та нових технологій їх виробництва для широкого класу волоконно - оптичних систем : звіт про НДР (заключ.) / МОН України, ХНУРЕ ; наук. рук. І.Ш. Невлюдов ; викон. О.І. Филипенко та ін. – Харків, 2008. – 275 с.

Електронні ресурси:

Віддаленого доступу:

Про вищу освіту [Електронний ресурс] : Закон України від 01.07.2014 р. № 1556-VII // Верховна Рада України : офіційний веб-портал. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/1556-18#Text>. – Станом на 18.09.2021. – Назва з екрану.

Локального доступу:

Романчиков В. І. Основи наукових досліджень [Електронний ресурс] / В. І. Романчиков. – Київ : ЦУЛ, 2012. – 1 електрон. цифровий диск (CD – ROM). – Назва з етикетки диска.

ДОДАТОК Л

Коди виробів і програмних документів

Коди виробів:

- 306558 – модулі;
- 406110 – пристрої числового програмного керування верстатами у машинобудуванні;
- 406210 – пристрої числового програмного керування роботами у машинобудуванні;
- 411710 – системи інформаційно-вимірювальні;
- 411720 – пристрої для вимірювання електричних і магнітних величин;
- 412150 – пристрої комбіновані;
- 416616 – пристрої керування;
- 416633 – пристрої для реєстрації;
- 418240 – вузли детекторів;
- 418250 – пристрої детектування;
- 418455 – блоки електроні, електричні;
- 464430 – модуль програмного керування роботами;
- 468340 – модуль комутації;
- 758800 – плати друковані на гнучкій основі;
- 758713 – плати друковані на жорсткій основі.

Коди програмних документів:

- 501600 – програмні засоби автоматизації технології програмування;
- 502100 – системи керування базами даних;
- 502810 – експертні системи та засоби їх створення;
- 502900 – програмні засоби загального призначення інші;
- 503300 – програмні засоби для автоматизації наукових досліджень;
- 503320 – програмні засоби для управління складними приладами та системами;
- 504300 – програмні засоби для САПР;
- 504390 – програмні засоби для систем автоматизованого проектування;
- 504400 – програмні засоби для систем автоматизації технологічної підготовки виробництва;
- 505200 – програмні засоби для АСУ ТП;
- 505300 – програмні засоби для управління гнучкими виробничими системами;
- 505400 – програмні засоби для систем управління рухомими об'єктами;
- 505500 – програмні засоби для автоматизованих робочих місць;
- 505900 – програмні засоби для управління технічними об'єктами і технологічними процесами інші;
- 565100 – програмні засоби для локальних мікропроцесорних систем контролю, регулювання управління технологічними процесами.

Навчальне видання

НЕВЛЮДОВ Ігор Шакирович
ТОКАРЄВА Олена Віталіївна
ЦИМБАЛ Олександр Михайлович
БРОННІКОВ Артем Ігорович

НАВЧАЛЬНИЙ ПОСІБНИК З ПІДГОТОВКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА

для здобувачів вищої освіти денної і заочної форми навчання
спеціальностей 151 «Автоматизація та комп'ютерно-інтегровані
технології» та 174 «Автоматизація, комп'ютерно-інтегровані
технології та робототехніка»
освітньої програми «Системна інженерія»

Формат 60x84 1/16. Папір офсет. Друк цифровий
Ум. друк. арк. 12,7. Наклад 200 прим. Зам 23-01.

Видавництво та друк
ФОП Іванченко І.С.

пр. Тракторобудівників, 89-а/62, м. Харків, 61135
тел.: +38(050/093) 40-243-50

Свідоцтво про внесення суб'єкта видавничої справи до державного реєстру видавців,
виготовників та розповсюджувачів видавничої продукції ДК № 4388 від 15.08.2012 р.

www.monograf.com.ua



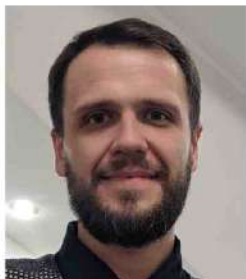
Невлюдов І.Ш.



Токарева О.В.



Цимбал О.М.



Бронінков А.І.



9 786178 332167