

# МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

## ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Інститут	Навчально-науковий інститут Комп'ютерних систем і технологій «Індустрія 4.0» ім. П.М. Платонова
Факультет	Автоматизації та робототехніки
Кафедра	Автоматизації технологічних процесів і робототехнічних систем
Ступінь вищої освіти	Бакалавр
Спеціальність	151 - Автоматизація та комп'ютерно-інтегровані технології
Освітня програма	Комп'ютерно-інтегровані робототехнічні системи

### КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

**Тема: « Роботизація і автоматизація технологічного процесу фасування сипучих матеріалів у мішки »**

Розробив	І.Д. Жигунов
Керівник д.т.н., доцент	В.Б. Єгоров
Зав. кафедри АТПіРС д.т.н., професор	В.А. Хобін
«Електронна версія роботи прийнята» Депозитор кафедри АТПіРС	Т.В. Волик

# ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Інститут	Навчально-науковий інститут Комп'ютерних систем і технологій «Індустрія 4.0» ім. П.М. Платонова
Факультет	Автоматизації та робототехніки
Кафедра	Автоматизації технологічних процесів і робототехнічних систем
Ступінь вищої освіти	Бакалавр
Спеціальність	151 - Автоматизація та комп'ютерно-інтегровані технології
Освітня програма	Комп'ютерно-інтегровані робототехнічні системи

## ЗАТВЕРДЖУЮ

Завідувач кафедри АТП і РС

д.т.н., проф. Хобін В.А.

«\_\_» \_\_\_\_\_ 202\_ року

## Завдання на кваліфікаційну роботу бакалавра

1. Студент: Жигунов Іван Дмитрович
2. Тема кваліфікаційної роботи «Роботизація і автоматизація технологічного процесу фасування сипучих матеріалів у мішки»
3. Зміст кваліфікаційної роботи (розділи, які потрібно розробити):
  - Розділ 1. Характеристика технологічного процесу, обґрунтування доцільності та шляхів підвищення ефективності керування процесом.
  - Розділ 2. Конкретизація задачі дотримання регламентів технологічного процесу, розробка і реалізація комплексу його моделей як об'єкта регулювання.
  - Розділ 3. Розробка і параметрична оптимізація алгоритмів регулювання, порівняльний аналіз САР.
  - Розділ 4. Конкретизація задачі і розробка алгоритмів логічного керування технологічним процесом.
  - Розділ 5. Вибір технічних засобів отримання інформації про змінні процесу та реалізації керуючих впливів, розробка технічної структури контролерно-комп'ютерної мережі
  - Розділ 6. Параметризація технічних засобів, контролера і мереж, програмування алгоритмів регулювання та логічного керування РТК
  - Розділ 7. Імітаційне та колізіяційне моделювання РТК
  - Розділ 8. Розробка фрагментів документації технічного забезпечення РТК та заходів з охорони праці
  - Розділ 9. Попереднє обґрунтування економічної доцільності впровадження проекту автоматизації.

Студент

Жигунов І.Д.

Керівник кваліфікаційної роботи

Егоров В.Б.

## ЗМІСТ

<b>ЗМІСТ .....</b>	<b>5</b>
<b>Перелік умовних позначень, символів, одиниць.....</b>	<b>9</b>
<b>ВСТУП</b>	
.....	<b>10</b>
<b>Розділ 1. Характеристика технологічного процесу, розробка концептуаль-ної моделі об'єкту керування.....</b>	<b>12</b>
1.1 Загальний аналіз технологічного процесу, реалізованого агрегатом, цілей і умов його ведення .....	13
1.1.2 Опис конструкції технологічного агрегату та особливостей його експлуатації.....	15
1.1.3 Формулювання (у загальному виді) умов, при яких можливо й доцільно реалізувати розглянутий технологічний процес .....	16
1.1.4. Параметризація схеми технологічного процесу й загальна (на якісному рівні) характеристика параметрів і їх взаємозалежностей .....	20
1.2 Конкретизація регламентів і умов ведення процесу, його формалізоване представлення.....	21
1.2.1. Виявлення нормативів ведення технологічного процесу і роботи технологічного агрегату .....	22
1.2.2. Аналіз наслідків виходу технологічних і експлуатаційних параметрів за регламентні допуски .....	23
1.2.3. Виявлення параметрів, що характеризують умови ведення технологічного процесу та експлуатації технологічного агрегату.....	24
1.2.4 Формалізація параметризованої схеми технологічного процесу і одержання його параметричної схеми .....	25
<b>Розділ 2. Конкретизація задачі дотримання регламентів технологічного процесу, розробка і реалізація комплексу його моделей як об'єкта регулювання.....</b>	<b>26</b>
2.1 Розробка структурної (координатної) схеми об'єкта керування .....	27
2.1.1 Конкретизація мети і завдань керування об'єктом, виявлення регульованих координат.....	28
2.1.2 Вибір управляючих дій об'єкту керування.....	28
2.1.3 Виділення та класифікація збурень об'єкта керування.....	28
2.1.4. Складання структурної (координатної) схеми об'єкту регулювання.....	29
2.2 Ідентифікація (розробка моделей) каналів перетворення об'єкта регулювання.....	29
2.2.1 Априорний аналіз статичних і динамічних властивостей об'єкту керування .....	29
2.2.2. Априорний аналіз і прийняття рішення про допустимість лінеаризації статичних властивостей каналів управління та збурень об'єкту регулювання, виходячи, зокрема, з діапазонів зміни регульованих змінних при роботі САР, для розробки якої ведеться ідентифікація об'єкту .....	30
2.2.3 Ідентифікація лінеаризованих моделей динаміки каналів управління об'єкта регулювання в околиці його робочих режимів .....	31
2.2.4 Аналіз отриманої в ході експерименту інформації, обґрунтування та вибір структури моделей каналів (структурна ідентифікація моделей). .....	32

2.3 Ідентифікація моделей відних дій об'єкту керування . Априорний аналіз вхідних впливів.....	33
2.3.1 Аналіз фізичної суті, можливих діапазонів зміни та частотних властивостей вхідних дій і їх наслідків. ....	34
2.3.2 Аналіз можливостей ідентифікації моделей координатних збурень і вибір загальної структури моделей .....	35
2.3.3 Планування пасивного експерименту для збору інформації про координатні збурення, структурна і параметрична ідентифікація неконтрольованих збурень .....	36
2.3.4 Оцінювання імовірнісних характеристик випадкових складових координатних збурень, аналіз оцінок щільностей імовірності, кореляційних функцій, спектральних щільностей і уточнення структури моделі (структурна ідентифікація).....	37
2.4 Реалізація у середовищі імітаційного моделювання моделей каналів перетворення дій і підтвердження їх відповідності експериментальним даним .....	38
2.4.1 Розробка схем моделювання динаміки каналів перетворення дій, отримання перехідних характеристик моделей і їх порівняння з експериментальними перехідними характеристиками.....	39
2.4.2 Представлення моделей випадкових складових дій у виді послідовно включених генератора базового випадкового процесу („білого шуму”) і формуючого фільтра.....	40
2.4.3 Розробка структурних схем моделювання, генерація випадкових складових моделей вхідних дій, параметрична ідентифікація їх імовірнісних характеристик і підтвердження відповідності моделей експериментальним даним.....	43
2.4.4 Відтворення у середовищі імітаційного моделювання моделей статички каналів перетворення дій і підтвердження їх відповідності експериментальним даним .....	43
2.4.5 Реалізація у середовищі імітаційного моделювання повної моделі об'єкту керування і підтвердження її адекватності .....	43
2.5 Формулювання задач керування технологічним агрегатом.....	45
2.5.1 Декомпозиція загальної задачі керування технологічним агрегатом і формулювання приватних завдань .....	45
2.5.2 Обґрунтування необхідності і доцільності автоматизації кожної з приватних задач керування .....	45
2.5.3 Формалізація вимог до вирішення задач регулювання і вибір принципу побудови системи автоматичного регулювання .....	46
2.5.4 Формалізація інтегральних вимог до перехідних і динамічно сталих процесів регулювання у формі інтегрального критерію оптимальності САР .....	46
2.5.5 Вибір, обґрунтування і представлення у вигляді узагальненої структурної схеми принципу побудови системи автоматичного регулювання.....	47
<b>Розділ 3. Розробка і параметрична оптимізація алгоритмів регулювання, порівняльний аналіз САР .....</b>	<b>49</b>
3.1 Конкретизація структури САР і алгоритмів регулювання.....	49
3.1.1 Розробка на основі обраного загального принципу побудови САР, її найпростішої конкретної структури.....	49
3.1.2 Вибір кількох альтернативних варіантів типових алгоритмів регулювання, запис їх рівнянь і передаточних функцій .....	49

3.1.3 Розробка структурної схеми і програми цифрового імітаційного моделювання САР при детермінованих і стохастичних вхідних впливах .....	50
3.2 Параметричний синтез САР з різними варіантами типових алгоритмів регулювання та їх порівняльний аналіз для детермінованих вхідних впливів .....	61
3.2.1 Аналіз особливостей об'єкта регулювання, що знижують динамічну точність і вибір способів її підвищення за рахунок введення в структуру САР додаткових зв'язків. ....	61
3.2.2 Розробка на основі обраних способів підвищення динамічної точності САР її структурної схеми і формулювання (в аналітичній формі) умов, що забезпечують необхідні властивості САР.....	61
3.2.3 Вивід передаточних функцій коригуючих зв'язків, аналіз їх структури з умов фізичної можливості бути реалізованим, приведення до фізично реалізується увазі, подання до формі з'єднання типових динамічних ланок та отримання перехідних характеристик .....	62
3.2.4 Розробка структурної схеми і програми цифрового імітаційного.....	63
3.2.5 Аналіз грубості САР підвищеної динамічної точності до варіацій параметрів об'єкта регулювання .....	63
3.2.6 Вибір параметрів ОК, по який доцільно оцінити грубість САР значень .....	63
3.2.7 Аналіз сталих процесів в САР при стохастичних вхідних впливах .....	63
3.2.8 Вплив неконтрольованих збурень на показники якості функціонування САР в сталих режимах.....	63
3.2.9 Вибір параметрів ОК, по який доцільно оцінити грубість САР значень .....	66
<b>Розділ 4. Конкретизація задачі і розробка алгоритмів логічного керування технологічним процесом .....</b>	<b>69</b>
4.1 Короткий опис технологічного процесу з наведенням детальних процедур технологічного пуску, технологічної зупинки та аварійної зупинки у позаштатних ситуаціях .....	69
4.2 Формалізація регламентів функціонування Робототехнічного комплексу (РТК): за пуском, технологічною та аварійною зупинками технологічного процесу у вигляді блок-схем та їх опис .....	74
4.3 Розробка алгоритмів автоматичного керування: за пуском, технологічною та аварійною зупинкою (циклом роботи) ТОК у вигляді блок-схем та їх опис.....	75
<b>Розділ 5. Вибір технічних засобів отримання інформації про змінні процесу та реалізації керуючих впливів, розробка технічної структури контролерно-комп'ютерної мережі.....</b>	<b>90</b>
5.1 Характеристики середовищ, з якими взаємодіють засоби автоматизації й обслуговуючий персонал, які повинні бути враховані при виборі технічних засобів і місць їх розміщення.....	90
5.2 Характеристики середовищ, з якими взаємодіють засоби автоматизації й обслуговуючий персонал, які повинні бути враховані при виборі технічних засобів і місць	

їх розміщення та вибір технічних засобів реалізації керуючих дій на технологічний процес і управління обладнанням .....	96
5.3 Вибір технічних засобів реалізації керуючих дій на технологічний процес і управління обладнанням .....	102
<b>Розділ 6. Параметризація технічних засобів, контролера і мереж, програмування алгоритмів логічного керування та регулювання САК.....</b>	<b>127</b>
6.1 Ознайомлення з мовами програмування FANUC. Демонстрація декількох способів програмування роботів .....	127
6.2 Вибір технічних засобів реалізації керуючих дій на технологічний процес і управління обладнанням .....	130
6.3 Розробка програм для реалізації алгоритмів регулювання.....	138
<b>Розділ 7. Візуалізація РТК в програмному середовищі ROBO DK .....</b>	<b>197</b>
7.1 Створення цифрового двійника .....	197
7.2 Створення симуляції у ROBO DK.....	204
7.2.1 Розташування створених об'єктів у ROBO DK .....	206
7.2.2 Створення візуалізації роботи ділянки.....	207
<b>Розділ 8. Розробка фрагментів документації технічного забезпечення САК та заходів з охорони праці .....</b>	<b>214</b>
8.1 Коротка характеристика технологічного об'єкту керування, склад документації, що розроблена, та опис основних технічних рішень із побудови САК.....	215
8.2 Розробка схеми автоматизації та специфікацій на технічні.....	215
8.3 Розробка принципової електричної схеми живлення .....	215
8.4 Розробка принципової електричної схеми підключення до блоку аналогових та дискретних входів та виходів ПЛК .....	222
8.5 Охорона праці .....	229
<b>Розділ 9. Попереднє обґрунтування економічної доцільності впровадження проекту автоматизації.....</b>	<b>232</b>
9.1 Розрахунок інвестиційних витрат на розробку та реалізацію робототехнічного комплексу .....	233
9.2 Розрахунок показника економічного ефекту від модернізації системи керування.....	233
<b>Список літератури.....</b>	<b>235</b>

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ

РТК – робототехнічний комплекс;

ОК – об'єкт керування;

САУ – система автоматичного керування;

ОПС – оптимальний параметричний синтез;

ЗНВ – зона незначимих відхилень;

ПЛК – програмований логічний контролер;

ПК – персональний комп'ютер;

ММ – математична модель

ВП – випадковий процес

САР ПДТ – САР підвищеної динамічної точності

$\alpha$  – кут повороту енодера, рад;

$\omega$  – кутова швидкість, рад/с;

і т.д.

## ВСТУП

В сучасному світі технології швидко розвиваються, проникаючи у різні сфери життя людей. Однією з таких областей, яка зазнає значних змін завдяки новаторським розробкам, є промисловість. У цьому контексті, роботехніка, яка включає в себе використання роботів і автоматизованих систем, займає особливе місце. Вона надає великі можливості для оптимізації процесів та підвищення продуктивності.

Цементні заводи є ключовим елементом будівельної промисловості, вони забезпечують виробництво цементу, який використовується в будівельних проектах по всьому світу. У традиційній схемі роботи таких заводів, багато процесів виконується вручну або за допомогою промислового устаткування. Проте, з впровадженням роботехніки цей підхід може зазнати суттєвих змін.

Використання роботів на цементних заводах відкриває широкий спектр можливостей для автоматизації та удосконалення виробничих процесів. Роботи можуть бути використані для автоматичного завантаження сировини, переміщення великих вантажів, моніторингу та регулювання параметрів виробництва, а також для забезпечення безпеки працівників на заводі.

Однією з найбільш вигідних переваг використання роботів у цементній промисловості є підвищення ефективності та точності процесів. Роботи можуть працювати неперервно, без втрати продуктивності, і виконувати складні завдання з високою швидкістю.

Перевагою використання роботів у цементній промисловості є зменшення ризику для працівників. Цементний виробничий процес може включати важкі фізичні роботи, контакт з небезпечними матеріалами та ризик виникнення травм. Встановлення роботів для виконання таких завдань дозволяє знизити вплив на здоров'я людей та забезпечити безпеку працівників.

Крім того, роботехніка сприяє зниженню витрат на виробництво цементу. Роботи можуть бути програмовані для ефективного використання сировини, уникнення втрат та забезпечення оптимального рівня виробництва. Це дозволяє зменшити витрати на матеріали та збільшити загальну ефективність процесу.

Застосування роботів у цементній промисловості також сприяє зниженню впливу на довкілля. Роботи можуть бути оснащені сучасними системами керування енергією, що дозволяє ефективно використовувати ресурси та зменшити викиди шкідливих речовин у процесі виробництва цементу.

Враховуючи зростаючі потреби у будівельній галузі та постійне прагнення до покращення продуктивності та якості, використання роботів у цементній промисловості стає необхідним. Вони допомагають підвищити ефективність, точність та безпеку процесів, знизити витрати та негативний вплив на довкілля.

Роботехніка стає ключовим фактором успіху для сучасних цементних заводів у досягненні конкурентних переваг і досягненні високих стандартів якості виробництва цементу.

## РОЗДІЛ 1. ХАРАКТЕРИСТИКА ТЕХНОЛОГІЧНОГО ПРОЦЕСУ, РОЗРОБКА КОНЦЕПТУАЛЬНОЇ МОДЕЛІ ОБ'ЄКТУ КЕРУВАННЯ

### 1.1 Загальний аналіз технологічного процесу, реалізованого агрегатом, цілей і умов його ведення

Технологічний процес - це послідовність дій, які необхідно виконати для виробництва продукції. Технологічний процес може бути складним і включати в себе багато різних операцій. Тому, перед початком виробництва, потрібно провести технологічний пуск. .

Технологічний процес наповнення мішка здійснюється шляхом подачі цементу з верхнього накопичувального бункеру, куди він потрапляє під дією шнекового завантажувача з вагону-хоперу. На низу бункера іноді встановлюють шиберну засувку, яку використовують в якості регулюючого органу. Виконавчий механізм відкриває засувку спочатку на більшу степінь, а потім для більш точного визначення дози – на меншу степінь. Однак через властивість цементу до злежування та утворення склепінь можуть виникати раптові зсуви цементу, які приводять до значного перевантаження мішка з цементом. Тому частіше в якості регулюючого органу використовують шлюзові затвори або шнеки, які під дією електродвигунового виконавчого механізму спочатку обертаються швидко, а під кінець завантаження мішка – з меншою швидкістю, щоб забезпечити необхідну точність дозування.

Визначення необхідної дози здійснюється шляхом зважування мішка з цементом на вагах. В якості датчика ваги найчастіше використовуються тензометричний датчик. Тензометричний датчик (від латан. *tensus* - напружений) - це різновид датчика, що перетворює прикладену до нього фізичну силу на електронний сигнал. Його ще називають тензорезистивним чи просто тензодатчиком. Вимірювальним елементом тензодатчика є тензорезистор - резистор, у якого опір змінюється залежно від його деформації. У найбільш поширеному випадку, тензорезистор є невеликою пластиною-основною, на яку приклеєна металева пластина-фольга або зигзагоподібний

провідник. Зверху провідник ламінується тонкою плівкою. Основа зазвичай робиться з тканини, пластмаси, полімерної плівки чи паперу. Крім металеві фольги, тіло чутливого елемента може бути зроблено з напівпровідника - германію або кремнію - і напилюватися на основу тонким шаром. За способом деформації пружного елемента розрізняють датчики, що працюють на:

- стиснення (тензодатчики колонного типу);
- розтягування (S-подібні тензометричні датчики);
- скручування (торсійні тензодатчики);
- вигин (тензометричні датчики балкового типу);
- зсув (балки зсуву).

Спосіб деформації не сильно впливає на точність і характеристики обладнання, тому вибір, які використовувати тензодатчики, робиться виходячи з простоти і зручності їх монтажу в обладнанні.

За типом сигналу, що видається, тензодатчики діляться на аналогові і цифрові. На якість вимірів це не впливає, основна різниця – цифрові датчики простіше замінювати та обслуговувати.

Залежно від точності, тензометричні датчики поділяються на 4 класи. Найбільш поширеними є тензодатчики класу С3, де С - це клас, а число 3 означає кількість тисяч перевірочних поділів (тобто 3000). D – найнижчий рівень точності, А, відповідно, найвищий. Комбінована похибка класу точності С3 становить 0,02%. Це означає, що в різних умовах похибка буде змінюватися, а слово "комбінована" можна розуміти як аналог середнього арифметичного. Чим більше перевірочних поділів, тим вища точність тензодатчика і датчик класу С5 точніше датчика класу С3.

Клас точності визначає величину похибки. Якщо тензодатчики мають однакову кількість перевірочних поділів, але різний клас, похибка буде різною. У тензодатчика D1 похибка на максимальних навантаженнях буде вищою за похибку датчика С1 в 1,5 рази. Клас точності та кількість перевірочних поділів тензометричних датчиків регламентується державним стандартом ДСТУ ОІМЛ

R-1:2019. У маркуванні тензометричного датчика вказується клас точності, кількість перевірочних поділів та найбільша межа зважування (НМЗ).

Корпус тензодатчиків зазвичай виготовляється з легованої чи нержавіючої сталі. Цей факт може відобразитися у найменуванні. Наприклад, тензометричні датчики ZSFY компанії Keli мають у назві закінчення -A, якщо вони зроблені з легованої сталі або -SS, якщо з нержавіючої. Приклад - ZSFY-A20t - це тензодатчик із легованої сталі з НМЗ 20 тонн.

За кількістю діапазонів вимірювання тензодатчики поділяються на одноінтервальні, двоінтервальні та багатоінтервальні, які на різних навантаженнях ваги видають результат із різною дискретою. Робиться це для підвищення точності зважування на малих навантаженнях. Наприклад, одноінтервальні ваги з НМЗ 100 кг мають дискрету 20 г на всьому діапазоні зважування, а двоінтервальні ваги в діапазоні до 30 кг мають дискрету 10 г.

Компенсований діапазон температури - це діапазон, де тензодатчик зберігає свої метрологічні характеристики. Стандартним компенсованим діапазоном для тензодатчиків вважається температура від -10 до +40. У деяких моделей його розширено.

Ваги, сертифіковані за III-середнім класом точності, мають взаємозв'язок найбільшої межі зважування (НМЗ), дискрети (ціни розподілу шкали, D) та найменшої межі зважування (НмМЗ).

Табл. 1.1 Метрологічні характеристики ваг III класу точності

НМЗ	Дискретність	НмМЗ	Характерні види ваг
1 кг	0,5 г	10 г	• Настільні торгові ваги
2 кг	1 г	20 г	• Кранові ваги на 30 кг • • Ветеринарні ваги
3 кг	1 г	20 г	• Фасувальні дозатори
5 кг	2 г або 1 г	40 г або 20 г	• підлогові складські
6 кг	2 г або 1 г	40 г або 20 г	
10 кг	5 г або 2 г	100 г або 40 г	
15 кг	5 г або 2 г	100 г або 40 г	

30 кг   **10 г** або 5 г     **200 г** або 100 г

60 кг   **20 г** або 10 г     **400 г** або 200 г

1.1.2 Складання схеми та опис суті технологічного процесу, реалізованого технологічним агрегатом, як цілеспрямованого перетворення матеріальних і енергетичних потоків

Суть технологічного процесу – це цілеспрямоване перетворення матеріальних і енергетичних потоків у спеціальному технологічному устаткуванні.

Цемент горизонтальним шнеком (гвинтовим конвеєром), який приводиться у рух приводом М1, подається до похилого шнека, що приводиться у рух приводом М2, і завантажується у приймальний бункер. Рівень цементу в бункері визначається за сигналами датчиків нижнього ДНР та верхнього ДВР рівнів, які використовують для керування пуском та зупинкою приводів М1 і М2 (рис. 1.1).

Рис. 1.1 Схема лінії розвантаження цементу з вагону-хопера та фасування у мішки

З бункера цемент потрапляє до шнека напівавтоматичного фасувальника, який приводиться у рух приводом М3, і далі висипається у мішок. Мішок на

вагову платформу фасувальника встановлюється оператором. Частота обертання приводу МЗ поступово зменшується частотним перетворювачем ЧП для забезпечення точності дозування. Поточна вага мішка з цементом визначається датчиком ваги ДГ, до складу якого входить тензорезистор і нормуючий перетворювач. Після входу сигналу ваги мішка у регламентну зону, привід МЗ зупиняється і наповнений мішок пневмоприводом Р1 скидається з вагової платформи.

1.1.3 Опис конструкції технологічного агрегату та особливостей його експлуатації.

Комплект лінії на базі розвантажувача КРС-30Ц та напівавтоматичного фасувальника в клапанні мішки МФЦ-10 випускається підприємством «Еквіптех», м. Дніпро, Україна. Ця лінія призначена для механізації технологічного процесу фасування цементу, сухих сумішей, тонкодисперсних матеріалів у закриті (клапанні) мішки (за ГОСТ 2226-75) за вагою від 10 до 50 кг.

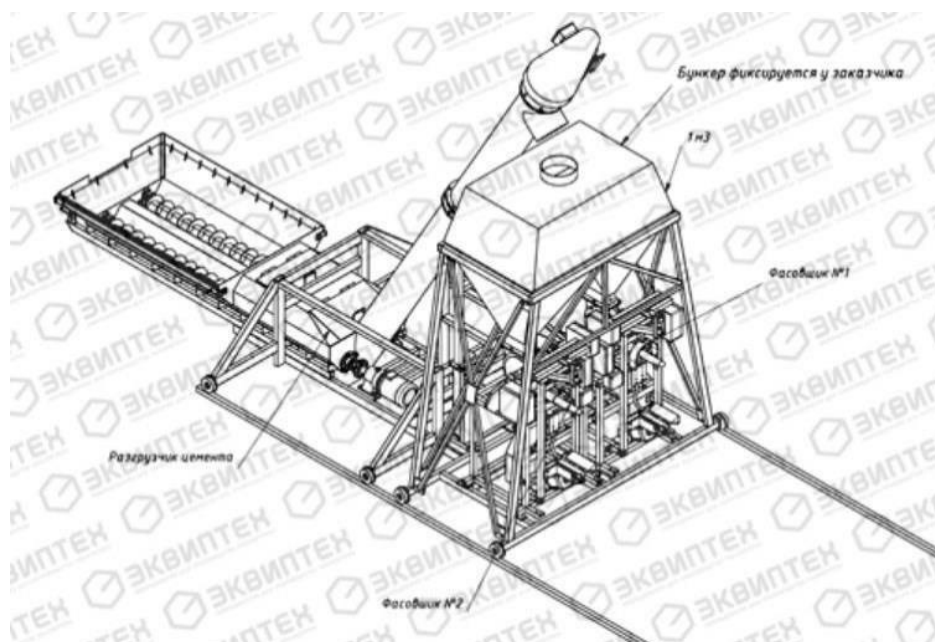


Рис. 1.2 Загальний вид лінії розвантаження цементу з вагону – хопера і фасування у мішки

Технічні характеристики лінії:

1 Продуктивність, т/год

12 - 20

2 Похибка дозування матеріалу в мішок,	% $\pm 2$
3 Встановлена потужність, кВт	17,5
4 Напруга,	380
5 витрата повітря, л/хв	300
6 Маса, т	2
7 Тип вагового контролера	мікропроцесорний
8 Мінімальний розмір мішка 25 кг см	48*37*10

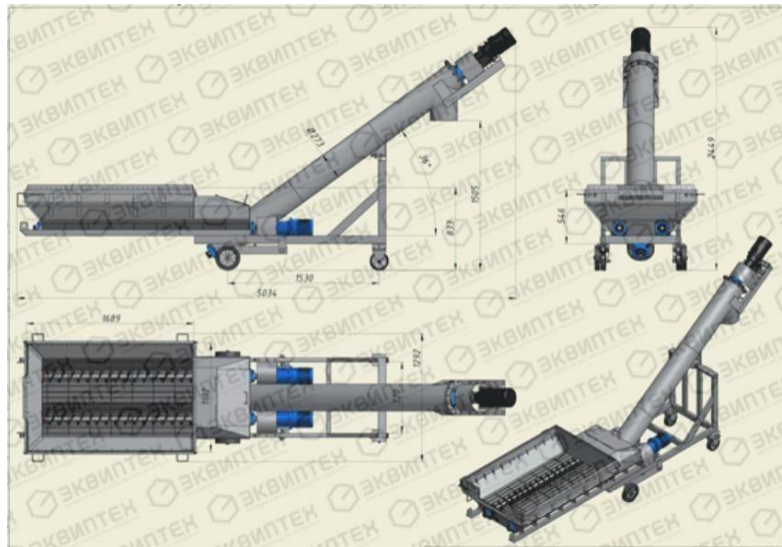


Рис. 1.3 Розвантажувач КРС-30Ц

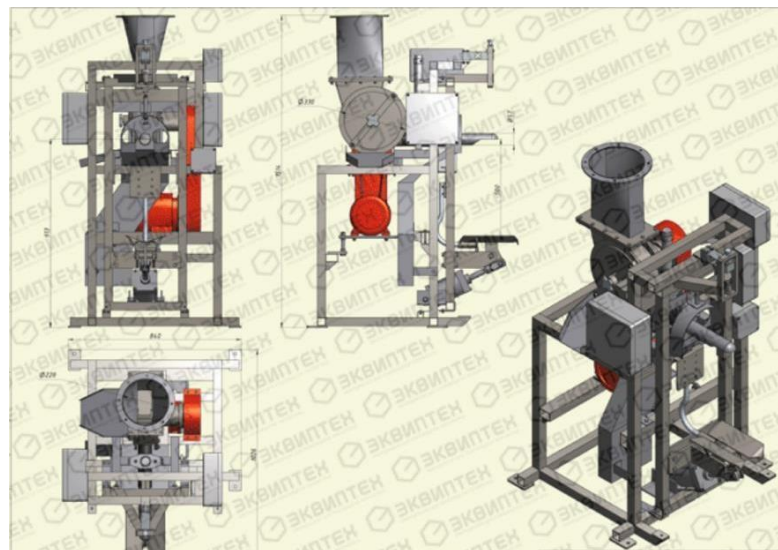


Рис. 1.4 Напівавтоматичний фасувальник в клапанні мішки МФЦ-10

Принцип роботи заснований на механічній подачі продукту, що фасується (цементу) РОТОРНИМ МЕТАЛЮ – без використання стисненого повітря, що суттєво знижує енерговитрати та пилоутворення, виключає розриви мішків під тиском повітря. Фасування цементу та інших сипких сумішей у клапанні мішки здійснюється за рахунок застосування відцентрової сили, що надається робочим колесом. Робоче колесо також виконує функції подрібнювача у випадках попадання цементу, що злежався, у фасувальну машину.

Алгоритм роботи фасовщика цементу в мішки:

- оператор на пульті керуванні (контролері) задає необхідну фасувальну дозу (наприклад 50 кг), "хвости" - інерційність матеріалу враховуються;
- клапанний мішок вручну надівається на патрубок; - оператор натискає кнопку "Start";
- відбувається пневматичне фіксування мішка, враховується вага мішка, відкривається заслінка, що подає, включається роторний металник і фасований продукт (цемент) надходить через патрубок в мішок;
- мішок починає наповнюватися - змінюється його вага, після досягнення запрограмованої межі відбувається зниження швидкості обертання ротора з наступною зупинкою, закривається заслінка. Вага прописується в пам'ять контролера. Мішок зафасований;
- пневмо-притискання звільняє мішок і поворотне сидло скидає його. Машина готова до наступної фасування;
- цикл фасування починається наново.

Суттєвим недоліком розглянутої лінії є використання людини-оператора, якій доводиться виконувати одноманітну фізичну роботу в умовах запиленого повітря. Повністю автоматизувати вказаний технологічний процес можна за рахунок використання робототехнічного комплексу. Можливий варіант додаткових технічних засобів вказаний на рис. 1.5, де:

1. Система заповнення мішка цементом.
2. Конвеєр.
3. Стіл для розташування заготовок (порожніх мішків).
4. Робот Fanuc M-10iA.
5. Кастомний пневматичний захват.
6. Пневмоциліндр FESTO.
7. Лінійний привід FESTO.
8. Вакуумний пневмозахват.

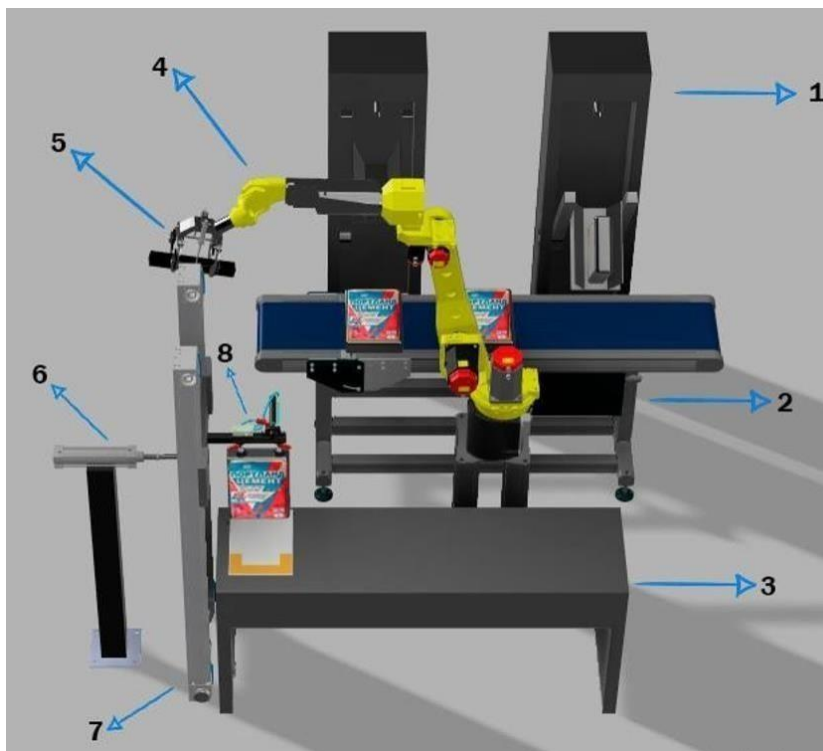


Рис. 1.5 Загальний вид робототехнічного комплексу

Алгоритм дій робототехнічного комплексу:

1. Підняття проміжного мішку завдяки пневматичного захвату.
2. Позиціонування мішка пневматичним циліндром для того, щоб робот зміг взяти його.
3. Взяття мішка за допомогою маніпулятора та перенесення на систему заповнення мішка.
4. Після заповнення сипучим матеріалом, система отримає сигнал від датчика ваги, що мішок заповнений та опускає його на конвеєр.

5. Конвеєр відправляє мішок на наступну ділянку.

1.1.4 Формулювання (у загальному виді) умов, при яких можливо й доцільно реалізувати розглянутий технологічний процес.

Мета ведення процесу – отримання заданих продуктів із заданими властивостями (показниками якості). При промисловому виробництві досягнення мети може бути виправдано тільки тоді, коли технологічний процес має сенс реалізувати, коли виконується цілий ряд вимог до технологічного процесу. Найчастіше зустрічаються такі з них:

- кількість виробленого в одиницю часу продукту не повинно бути меншим встановленої межі;
- втрати сировини та (або) продукту не повинні перевищувати нормативних значень;
- питомі, тобто на одиницю продукції, витрати енергії на процес не повинні перевищувати нормативних значень;
- режими роботи технологічного обладнання не повинні виходити за гранично-припустимі;
- шкідливий вплив технологічного процесу на людину та навколишнє середовище не повинен перевищувати нормативних значень;
- порушення режимів ведення технологічного процесу та роботи обладнання, що приводять до аварійних ситуацій, переростання яких в аварію приводить до істотних економічних втрат або навіть до катастрофічних наслідків, повинні бути досить рідкісними подіями та повинні бути передбачені заходи щодо запобігання переростання аварійних ситуацій в аварію.

Технологічний процес фасування цементу у мішки доцільно реалізовувати, якщо забезпечуються:

- напруга живлення в електромережі  $380\text{ В} \pm 10\%$ ;
- компресор забезпечує тиск стисненого повітря 5...6 бар.

1.1.5. Параметризація схеми технологічного процесу й загальна (на якісному рівні) характеристика параметрів і їх взаємозалежностей.

Технологічна схема є графічним відображенням руху і перетворення матеріальних та енергетичних потоків. Кожен потік і процес перетворення його механічних, фізичних, хімічних і біологічних властивостей характеризується набором параметрів, які відображають їх властивості та умови перетворення потоків. Їх нанесення на схему технологічного процесу і є її параметризацією.

Проведемо параметризацію технологічної схеми фасування цементу у мішки на лінії розвантажувач КРС-30Ц - напівавтоматичний МФЦ-10. Результати параметризації наведені на рис. 1.6.

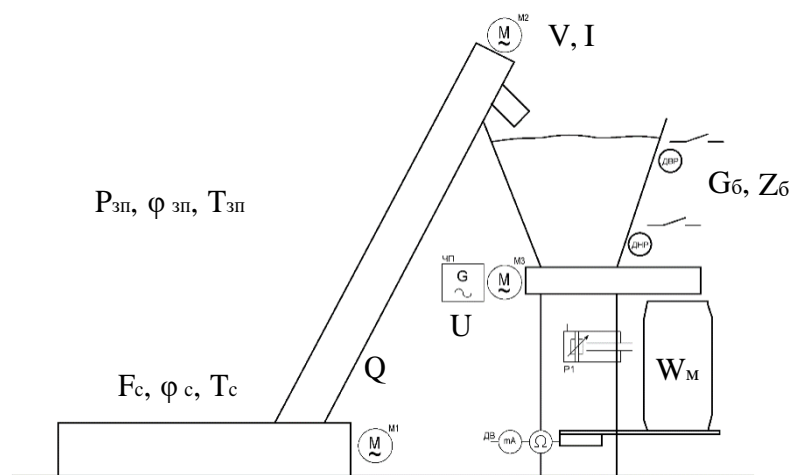


Рис. 1.6 Параметризована схема процесу фасування цементу

Позначення на параметризованій технологічній схемі:

$F_c, \varphi_c, T_c$  – витрати, вологість, температура цементу на вході;  $G_6, Z_6$  – висота стовпа та степінь злежуваності цементу в бункері;

$P_{zp}, \varphi_{zp}, T_{zp}$  – тиск, відносна вологість, температура зовнішнього повітря;  $W_m$  – вага цементу в мішку;

$U$  – керуюча дія на зміну частоти обертання приводу  $M_3$  фасувальника;  $Q$  – знос механічних частин лінії та відкладання цементу на стінках;

$V, I$  – напруга і струм живлення двигунів електроприводів.

1.2 Конкретизація регламентів і умов ведення процесу, його формалізоване представлення

### 1.2.1. Виявлення нормативів ведення технологічного процесу і роботи технологічного агрегату

Основу нормативів складають три основні регламенти:

- Технологічний регламент - визначає умови, при яких в результаті технологічного процесу виходить продукт із заданими властивостями. Він являє собою набір номінальних значень і допусків на відхилення від номіналу технологічних параметрів. Для технологічного процесу фасування цементу таким параметром є:  $W_M$  – вага цементу в мішку.
- Експлуатаційна регламент - визначає умови нормальної безаварійної роботи технологічного обладнання. Він являє собою набір номінальних значень і допусків на відхилення експлуатаційних параметрів цього технологічного обладнання. Для технологічного процесу фасування цементу таким параметром є:  $I$  – струм живлення двигунів електроприводів.
- Техніко-економічний та екологічний регламент визначає умови ефективності ведення процесу та його екологічності. Він являє собою набір номінальних або гранично-допустимих значень техніко-економічних і екологічних параметрів. До цієї групи параметрів можна віднести:  $I/W_M$  – відношення сили струму до ваги цементу в мішку.

### 1.2.2. Аналіз наслідків виходу технологічних і експлуатаційних параметрів за регламентні допуски

Відхилення параметрів технологічного процесу від регламентних значень, які викликаються зміною параметрів, що характеризують умови ведення процесу навіть при досить ефективному управлінні процесом, можуть перевищувати встановлені допуски. Залежно від наслідків, можна виділити два характерних типи таких порушень:

- а) технологічний процес не припиняється, вихід за допуски ліквідується з плином часу, наприклад, за рахунок управління процесом;
- б) технологічний процес припиняється повністю або частково, для його поновлення часто необхідна підготовка обладнання до повторного запуску

Допуски на відхилення поточних (режимних) значень регламентованих параметрів від номінальних прийнято поділяти і задавати у вигляді допусків на тривалі і короткочасні відхилення.

Тривало допустимі відхилення визначають зону незначущих для даного процесу відхилень (ЗНВ) розглянутого параметра. Усередині цієї зони значення параметра можна вважати приблизно рівним нормативному, тому відхилення, що не виходять з ЗНВ, можна не усувати.

Короткочасні допустимі відхилення параметрів перевищують їх ЗНО та призводять до зниження ефективності процесу, тому вони допустимі лише обмежений час, який задається і входить до регламенту. Вихід параметрів за зону короткочасно допустимих відхилень свідчить про порушення в технологічному процесі або в експлуатації обладнання, які можуть призвести до аварії.

Аналіз нормативів ведення технологічного процесу та експлуатації обладнання оформимо у вигляді таблиці регламентів.

Таблиця 1.2 Таблиця регламентів

№	Найменування параметрів	Позн.	Одиниця виміру	Номінальне значення	Допустимі відхилення від номіналу		
					Тривалі	Короткочасні	Час, с
1	Вага цементу в мішку	$W_m$	кг	50	$\pm 0,2$	-0,5	20

1.2.3. Виявлення параметрів, що характеризують умови ведення технологічного процесу та експлуатації технологічного агрегату.

Поточні значення нормативних параметрів завжди відрізняються від номінальних внаслідок зміни умов, в яких ведеться процес. Ці умови також характеризуються певними параметрами і можуть бути згруповані за деякими ознаками.

Сировинні параметри характеризують властивості продукту, що йде на переробку. До цієї групи параметрів віднесемо:  $F_c$ ,  $\varphi_c$ ,  $T_c$  – витрати, залишкову вологість, температуру цементу на вході в лінію.

Енергетичні параметри характеризують енергію, яка підводиться до технологічного устаткування ззовні і витрачається на зміну властивостей продуктів. До цієї групи параметрів віднесемо:  $V$  – напругу живлення двигунів електроприводів;  $F_{зп}$ ,  $\varphi_{зп}$ ,  $T_{зп}$  – витрати, відносна вологість, температура зовнішнього повітря.

Механічні параметри технологічного обладнання характеризують стан його робочих органів. До цієї групи параметрів віднесемо:  $Q$  – знос механічних частин лінії та відкладання цементу на стінках;  $U$  – керуюча дія на зміну частоти обертання приводу  $M_3$  фасувальника.

1.2.4 Формалізація параметризованої схеми технологічного процесу і одержання його параметричної схеми.

Параметризована схема ТП має змістовний характер. Формалізація опису взаємозв'язків між параметрами ТП досягається за рахунок переходу до формалізованої параметричної схеми ТП. При цьому ті параметри, які відображають мету функціонування об'єкту моделювання та додаткових вимог до нього, тобто регламентовані параметри будуть вихідними, а параметри, що відображають умови функціонування об'єкту - вхідними.

Зв'язок між вхідними і вихідними параметрами – причинно-наслідковий. У рамках даної схеми ці зв'язки задаються тільки на якісному рівні, згодом деякими зв'язками можна буде знехтувати, деякі представити опосередковано. Параметрична схема процесу фасування цементу наведена на рис. 1.7.

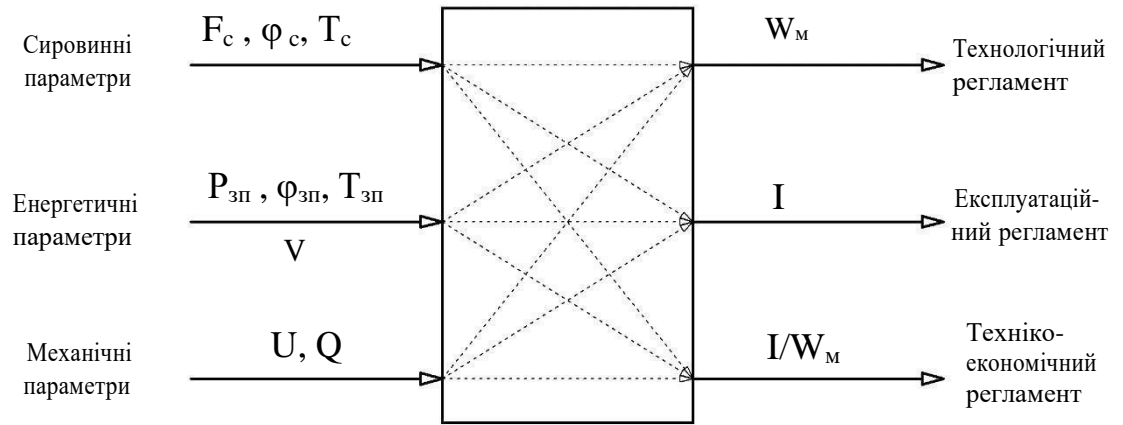


Рис.1.7 – Формалізована параметрична схема процесу процесу фасування цементу

## РОЗДІЛ 2. КОНКРЕТИЗАЦІЯ ЗАДАЧІ ДОТРИМАННЯ РЕГЛАМЕНТІВ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ, РОЗРОБКА І РЕАЛІЗАЦІЯ КОМПЛЕКСУ ЙОГО МОДЕЛЕЙ ЯК ОБ'ЄКТА РЕГУЛЮВАННЯ

### 2.1 Розробка структурної (координатної) схеми об'єкта керування

#### 2.1.1 Конкретизація мети і завдань керування об'єктом, виявлення регульованих координат.

Загальною метою функціонування підприємства в цілому є виконання планових завдань або договірних зобов'язань щодо номенклатури та обсягу продукції, а також отримання при цьому певного економічного ефекту.

Останній характеризується показниками прибутку, собівартості, розмірами виплачених штрафних санкцій і т.і.

Декомпозиція загальної («глобальної») мети для більш низьких рівнів виробничої ієрархії приводить до наступних цілей:

- а) на рівні окремого технологічного процесу – координація роботи технологічного обладнання, на якому реалізується процес;
- б) на рівні окремих одиниць технологічного обладнання (локальні цілі) – виконання технологічних, експлуатаційних і техніко-економічних регламентів, зокрема – оптимізація роботи окремих одиниць обладнання («локальна» оптимізація).

Звідси задачу керування можна розділити на:

- а) задачу регулювання, коли необхідно забезпечити виконання вимог регламенту. Отже, змінні, що відповідають технологічним і експлуатаційним параметрам будуть регульованими змінними об'єкту керування;
- б) завдання умовної оптимізації, коли при дотриманні вимог регламенту додатково відповідно до наперед сформульованим критерієм, вирішуються задачі оптимізації процесу.

Для процесу фасування цементу в якості регульованої координати доцільно обрати  $W_m$  – вагу цементу в мішку.

#### 2.1.2 Вибір управляючих дій об'єкту керування.

Необхідною умовою вирішення задач керування є наявність управляючих дій, що повинні бути вхідними змінними (координатами) ОК. Для задач

регулювання, такими управляючими діями, насамперед, є дії, щ

Еквівалентні координатні неконтрольовані збурення вважають діючими адитивно або управляючих дій, або керованих змінних. У цих випадках їх фізична інтерпретація і розмірність їх одиниць вимірювання визначається точкою їх додавання. У деяких випадках зручно детерміновану складову збурень приводити до управляючих дій, а стохастичну – до керованої змінного.

Параметричні збурення при керуванні ТП, як правило, є неконтрольованими. Спектральний склад збурень істотно різний для різних груп збурень. Для «наших умов» усі збурення, що прикладаються до об'єкта оптимізації, є істотно більше низькочастотними у порівнянні зі збуреннями, що прикладаються до об'єкта регулювання. Параметричні збурення є істотно більше низькочастотними у порівнянні з координатними. Самими найвисокочастотнішими збуреннями є шуми.

Всі вхідні дії, крім управляючої дії, віднесемо до неконтрольованих збурень  $f_n$ . Детерміновану складову цих збурень адитивно прикладемо до управляючих дій, а стохастичну складову – до регульованої координати.

#### 2.1.4. Складання структурної (координатної) схеми об'єкту регулювання

Структурна схема ОК є найбільш загальною (концептуальною) моделлю технологічного процесу як ОК.

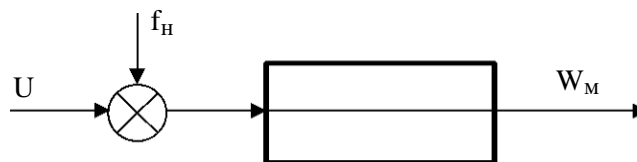


Рис.2.1 - Структурна схема процесу фасування цементу як об'єкта регулювання

$U$  – керуюча дія на зміну частоти обертання приводу  $M_3$  фасувальника;

$f_n$  - вектор неконтрольованих збурень;

$W_m$  – регульована змінна – вага цементу в мішку

### 2.1.5 Априорний аналіз статичних і динамічних властивостей об'єкту керування

Перед початком експериментів доцільно на основі априорних даних, виходячи з фізичної суті об'єкту, оцінити властивості каналів, моделі яких підлягають ідентифікації. По суті справи - провести попередню структурну ідентифікацію. Для технологічного процесу фасування цементу після нанесення керуючої дії динамічна рівновага регульованої змінної не встановлюється, тому за цим каналом управління цей ОК не має властивості до самовирівнювання.

### 2.1.6. Априорний аналіз і прийняття рішення про допустимість лінеаризації статичних властивостей каналів управління та збурень об'єкту регулювання, виходячи, зокрема, з діапазонів зміни регульованих змінних при роботі САР, для розробки якої ведеться ідентифікація об'єкту

Виходячи з фізичної суті процесів, які протікають в ОК, відомо, що відносно повною мірою математично ці процеси можуть бути описані нелінійними диференціальними рівняннями.

Але також відомо, що номінальний режим функціонування ОК пов'язаний з порівняно вузьким діапазоном зміни регульованих координат. А в такому діапазоні процеси можуть бути описані лінійними диференціальними рівняннями, тобто припустимою є лінеаризація моделі ОК. Таке спрощення моделі, з одного боку, приводить до неточності опису процесів, а з іншого боку, дозволить істотно спростити процедуру ідентифікації моделі з використанням інженерних методик.

## 2.2. Ідентифікація лінеаризованих моделей динаміки каналів управління об'єкта регулювання в околиці його робочих режимів

### 2.2.1 Короткий порівняльний аналіз і вибір доцільних вхідних дій для експериментального дослідження з метою отримання необхідної інформації про властивості каналів об'єкту для випадку, коли вхідні змінні каналів доступні для цілеспрямованої зміни

Активні експерименти можуть проводитися тільки з каналами, вхідні змінні яких доступні для цілеспрямованої зміни. Ними, насамперед, є керуючі дії. З розуміння простоти організації цих змін, мінімізації витрат часу на проведення експерименту та обробки його результатів, доцільно використовувати ступінчасті дії.

Для спрощення обробки результатів доцільно подавати дії по черзі на кожен з каналів окремо, фіксуючи реакції всіх виходів. 2.2.2 Планування активного експерименту на об'єкті для обраних вхідних дій і отримання реакцій на них у ході натурального або уявного експериментів.

План активного експерименту:

1. Змінюємо управляючу дію ступінчастим чином від  $U=0\%$  х.р.о. на 10%, відзначивши при цьому момент початку його зміни.
4. Реєструємо зміну вихідної змінної  $y=W_M$  до настання нового сталого динамічного режиму.

Результати активного експерименту при зміні  $U 0\% \rightarrow 10\%$  х.р.о. наведені на рис. 2.2.1

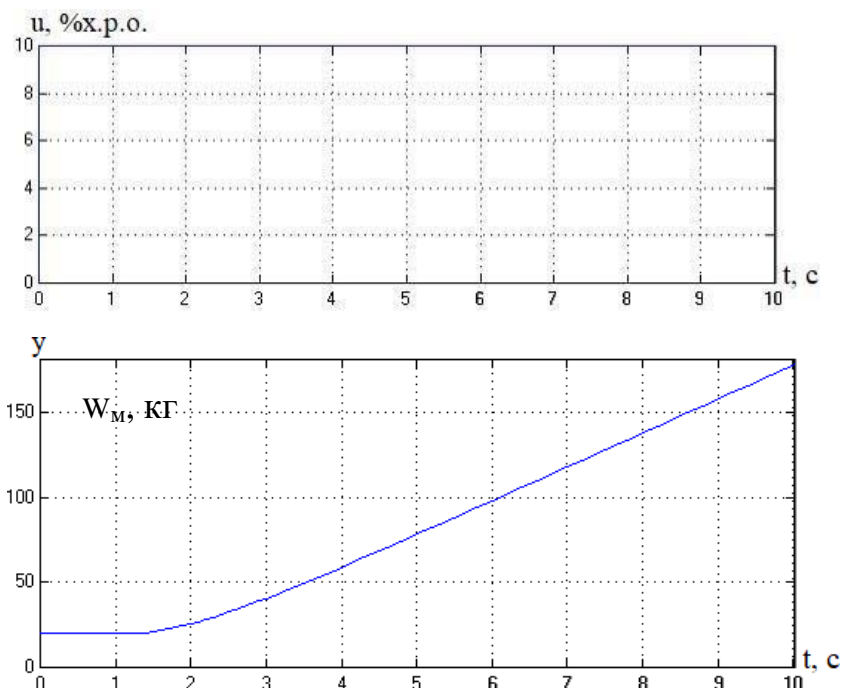


Рис. 2.2.1 – Результати активного експерименту за каналом  $U - W_M$

З рис. 2.2.1 видно, що платформа фасувальника і пустий мішок мають вагу 20 кг.

2.2.3 Аналіз отриманої в ході експерименту інформації, обґрунтування та вибір структури моделей каналів (структурна ідентифікація моделей).

За результатами активних експериментів видно (рис. 2.1), що ОК по каналу керування  $U - W_m$  не має властивості до самовирівнювання, а отже модель цього каналу може бути описана передатними функціями інтегруючої ланки першого та другого порядку:

$$W_0(p) = \frac{e^{-\tau p}}{T_p}; \quad W_0(p) = \frac{e^{-\tau p}}{T_p * (T_1 p + 1)}$$

2.2.4 Вибір методики і проведення параметричної ідентифікації моделей першого та другого порядків

Для параметричної ідентифікації моделей ОК з астатичними властивостями необхідно знайти сталу інтегрування (розмірний коефіцієнт)  $T$ , сталу часу  $T_1$  і час запізнення  $\tau$ .

Для моделі першого порядку каналу  $U - W_m$  проводимо наступні графічні побудови, знаходимо  $\tau$ ,  $\Delta t$  і  $\Delta y = \Delta W_m$

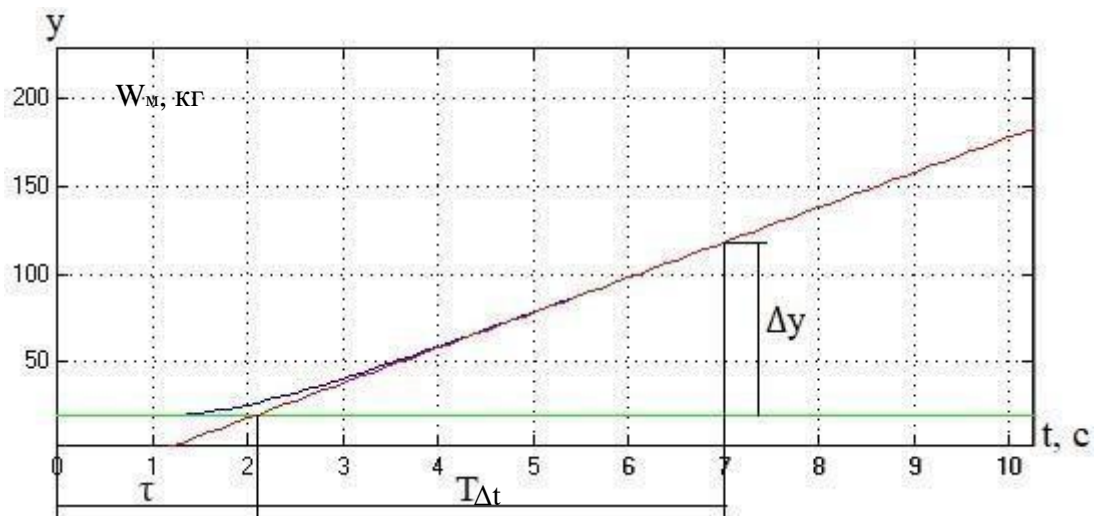


Рис. 2.2.2 Графічні побудови для параметричної ідентифікації моделі 1-го порядку ОК

$$\tau = 2,096 \text{ с} \approx 2,1 \text{ с};$$

$$\Delta t = 7 - 2,1 = 4,9 \text{ с};$$

$$\Delta y = \Delta W_m = 118,1 - 20 = 98,1 \text{ кг.}$$

$$T = \frac{\Delta u}{\Delta y} * \Delta t = \frac{10 - 0}{118,1 - 20} * (7 - 2,1) = \frac{10}{98,1} * 4,9 = 0,5 \text{ \%хро*с/кг}$$

Передатна функція моделі 1-го порядку ОК буде мати вигляд:

$$W_0(p) = \frac{e^{-2.096p}}{0.5p}$$

Для отримання моделі 2-го порядку спочатку проводимо графічне диференціювання залежності  $\Delta y(t)/\Delta t$ .

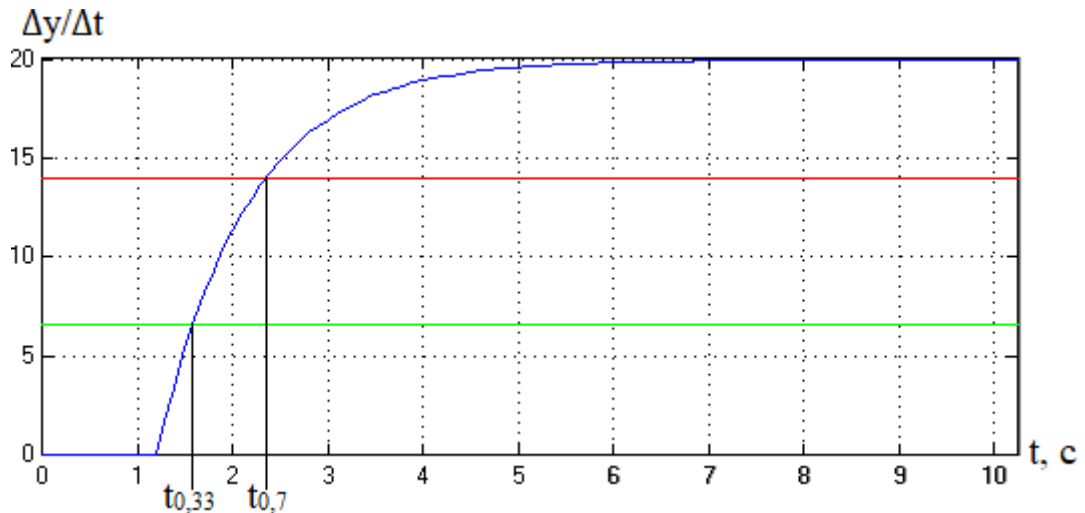


Рис. 2.2.3 Графік залежності  $\Delta y(t)/\Delta t$  з додатковими побудовами

Визначаємо рівні

$$\Delta y(\Delta t_{0,33}) = 0,33 \cdot \Delta y^{\text{МАКС}} = 0,33 \cdot 20 = 6,6 \text{ кг/с};$$

$$\Delta y(\Delta t_{0,7}) = 0,7 \cdot \Delta y^{\text{МАКС}} = 0,7 \cdot 20 = 14 \text{ кг/с}.$$

Стала інтегрування для моделі другого порядку і для моделі першого порядку однакова  $T = 0,5 \text{ \%хро} \cdot \text{с/кг}$ .

Проводимо графічні побудови (див. рис. 2.3 ) і знаходимо на осі часу значення  $t_{0,33}$  і  $t_{0,7}$ .

$$t_{0,33} = 1,581 \text{ с}; t_{0,7} = 2,344 \text{ с}. \text{ Тоді відповідно до методики визначаємо}$$

$$\tau = 0,5 \cdot (3 \cdot t_{0,33} - t_{0,7}) = 0,5 \cdot (3 \cdot 1,581 - 2,344) = 1,2 \text{ с};$$

$$T_1 = (t_{0,7} - \tau) / 1,2 = (2,344 - 1,2) / 1,2 = 0,95 \text{ с}.$$

Відповідно передатна функція моделі ОК другого порядку буде

$$W_0(p) = \frac{e^{-1.2p}}{0.5p \cdot (0.95p + 1)}$$

Ідентифікація моделей статички каналів управління об'єкту регулювання з істотно нелінійними властивостями

Для ОК, які не мають властивості до самовирівнювання, ідентифікація

моделей статички не проводиться.

## «ІДЕНТИФІКАЦІЯ МОДЕЛЕЙ ВХІДНИХ ДІЙ ОБ'ЄКТУ КЕРУВАННЯ»

### 2.3 Априорний аналіз вхідних впливів

2.3.1 Аналіз фізичної суті, можливих діапазонів зміни та частотних властивостей вхідних дій і їх наслідків.

Для технологічного процесу фасування цементу як ОК всі вхідні дії, крім управляючої дії, були віднесені до неконтрольованих збурень  $f_n$  (див. рис. 1.8).

Слід зазначити, що неконтрольовані збурення на ОК не є чітко певними фізичними параметрами. Вони є певною математичною абстракцією, про яку відомою є лише реакція системи регулювання на них. Діапазон їх зміни доцільно прив'язувати до фізичного діапазону зміни управляючих дій. З теорії автоматичного керування відомо, що фізичний діапазон зміни управляючої дії лежить у межах 0...100 %х.р.о., а діапазон зміни неконтрольованих збурень може становити приблизно 10...15% від діапазону зміни управляючої дії, тобто становить приблизно 10...15 %х.р.о. (якщо неконтрольовані збурення прикладені до управляючої дії), або  $(10...15) u_0$  (якщо неконтрольовані збурення прикладені до регульованої координати).

В інженерній практиці моделі вхідних дій звичайно представляють у формі процесу з декількома адитивними складовими, наприклад:

$$f(t) = f(t) + f^n(t) + f^c(t) + f^w(t),$$

де  $f(t)$  – детермінована повільно змінювана складова;

$f^n(t)$  – квазидетермінована складова;

$f^c(t)$  – стохастична центрована складова – центрований випадковий процес;

$f^w(t)$  – високочастотна стохастична складова – шум.

Варто враховувати, що складові  $f(t)$ ,  $f^n(t)$  є відносно низькочастотними діями, несприятливі наслідки яких САК повинна і може компенсувати.

Складова  $f^w(t)$  пов'язана або із шумами у вимірювальних каналах, або з високочастотними збуреннями, наслідки яких, у силу особливостей каналів керування ОК, скомпенсовані бути не можуть. В обох випадках  $f^w(t)$  в САК повинні бути відфільтровані. Визначивши властивості  $f^w(t)$ , можна підібрати параметри фільтрів низької частоти, які не «пропускають» шуми на вхід регулятора.

Вплив складової  $f^c(t)$  на регульовану змінну повністю не компенсується регулятором САР і якраз зменшення цього впливу і визначає ефективність алгоритмів, які реалізує система регулювання.

### 2.3.2 Аналіз можливостей ідентифікації моделей координатних збурень і вибір загальної структури моделей

Моделі неконтрольованих збурень доцільно представляти як суму чотирьох складових. Причому детерміновану  $f(t)$  та квазидетерміновану  $f^n(t)$  складові доцільно привести до управляючої дії, а стохастичну складову  $f^c(t)$  та шум  $f^w(t)$  доцільно привести до регульованої координати.

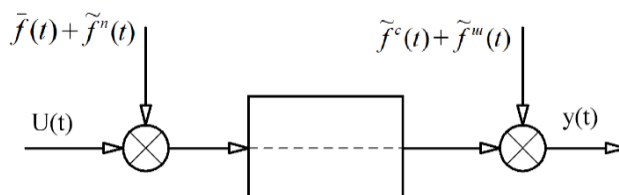


Рис. 2.3.1 Структурна схема моделі неконтрольованих збурень

На практиці дуже часто прагнуть використовувати найбільш прості моделі детермінованих складових. У таких умовах в якості моделі  $f(t)$  приймають модель ступінчастої дії,

$$f(t) = \begin{cases} a, & t > 0 \\ 0, & t < 0 \end{cases}, \text{ прикладеного до входу ОК.}$$

Ступінчасте збурення є одним з найбільш несприятливих з обмежених по величині дій на ОК (з погляду виникаючих у САК динамічних відхилень і тривалості виникаючих перехідних процесів).

Наближена ідентифікація  $\Delta f_n$  для ступінчастого збурення може бути проведена у такий спосіб:

1. Визначають (приблизно) середнє значення управляючої дії  $u(t) = u(t)$ , якому відповідає номінальне значення регульованої координати.

2. Задають значення  $\Delta f_n = 0,2u(t)$ .

Квазідетермінована складова  $f^n(t)$  може мати, наприклад, гармонійний характер, що відбиває вплив на регульовану змінну  $u(t)$  періодично працюючого обладнання. Модель  $f^n(t)$  знаходять при цьому безпосередньо із графіків  $u(t)$ , отриманих при проведенні пасивного експерименту.

Для відтворення моделей  $f^c(t)$  або  $f^m(t)$  досить часто використовують метод формуючого фільтру. Його можна подати у вигляді такої структурної схеми моделювання

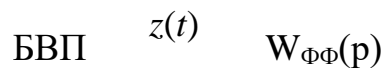


Рис. 2.3.2 – Базова модель випадкового процесу

де БВП - генератор базового випадкового процесу («білого шуму»);

$W_{\text{ФФ}}(p)$  – передаточна функція формуючого фільтра.

В подальшому для ТП фасування цементу як ОК розглядаються лише зовнішні впливи детермінованої  $f(t)$  та стохастичної  $f^c(t)$  складових неконтрольованого збурення  $f_n(t)$  на регульовану змінну  $W_m(t)$ .

2.3.4 Планування пасивного експерименту для збору інформації про координатні збурення, структурна і параметрична ідентифікація неконтрольованих збурень

2.3.5 План проведення пасивного експерименту:

а) при проведенні пасивного експерименту необхідно вивести об'єкт у зону робочого режиму  $U=0$  %хро,  $W_m = 20$  кг.

- б) провести пасивний експеримент, у ході якого отримати реалізації регульованої змінної  $W_m(t)$ .
- в) за отриманими перехідними процесами визначити час початку динамічно сталого процесу, крок квантування, тривалість реалізацій.

Якщо немає можливості провести натурні пасивні експерименти, то з навчальною метою сигнали збурень, як динамічні процеси, можна згенерувати за допомогою програми `rgen`, задавши необхідні параметри про властивості ОК по каналу регулювання.

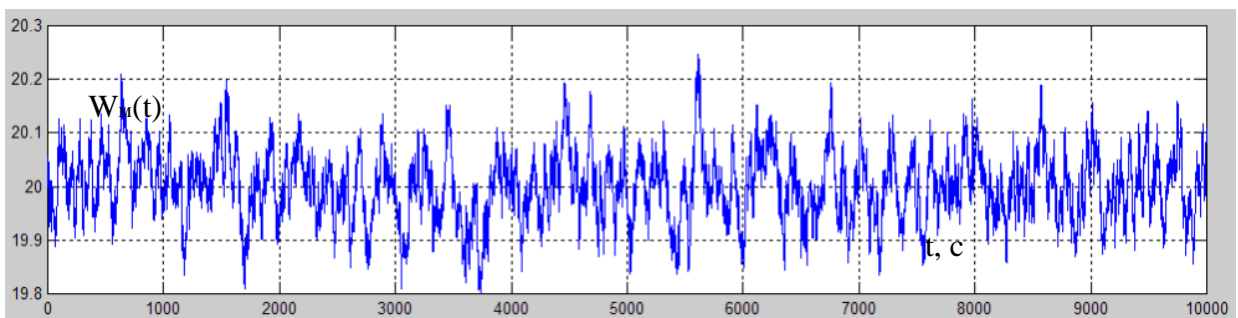


Рис. 2.3.3 Результати пасивного експерименту змін регульованої змінної  $W_m(t)$  під впливом неконтрольованого збурення  $f_n(t)$

2.3.6 Оцінювання імовірнісних характеристик випадкових складових координатних збурень, аналіз оцінок щільностей імовірності, кореляційних функцій, спектральних щільностей і уточнення структури моделі (структурна ідентифікація)

Для ідентифікації моделей випадкових складових неконтрольованих координатних збурень використаємо додаток `IdSoft` пакету Матлаб. Першим етапом ідентифікації моделі контрольованих і неконтрольованих збурень є оцінювання імовірнісних характеристик відповідних випадкових процесів, а саме щільності імовірності, математичного очікування, дисперсії, середньоквадратичного відхилення, середньоквадратичного періоду (СКП), кількості СКП у реалізації випадкового процесу, нормованої кореляційної функції і спектральної щільності випадкового процесу.

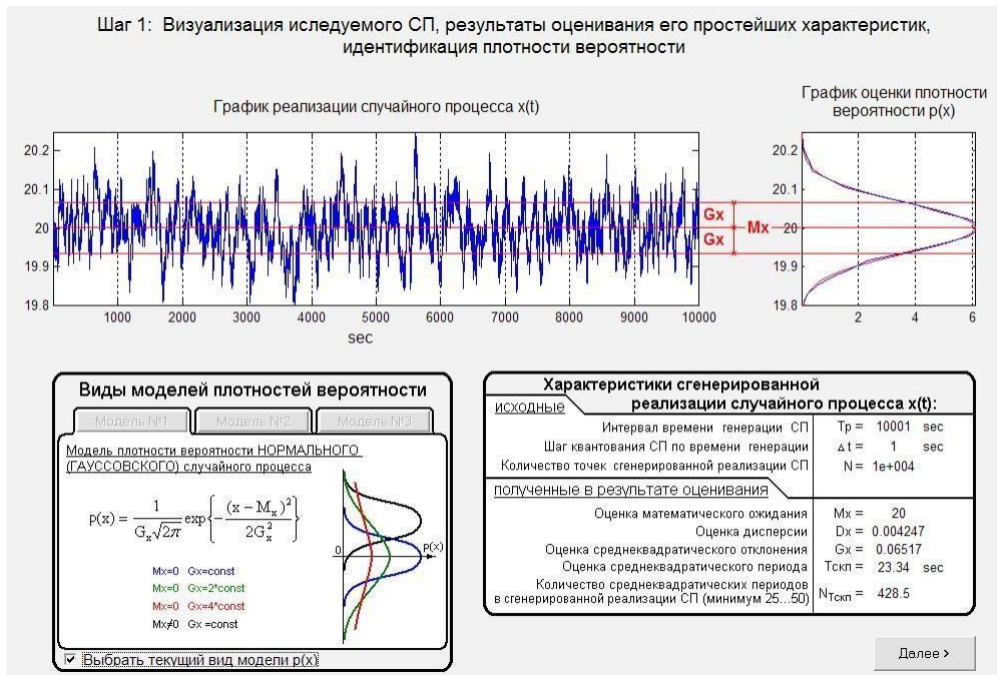


Рис. 2.3.4 Вибір моделі щільності імовірності розподілу процесу  $W_M(t)$

На другому кроці вибираємо модель №1 із семи можливих моделей автокореляційної функції та спектральної щільності процесу  $W_M(t)$

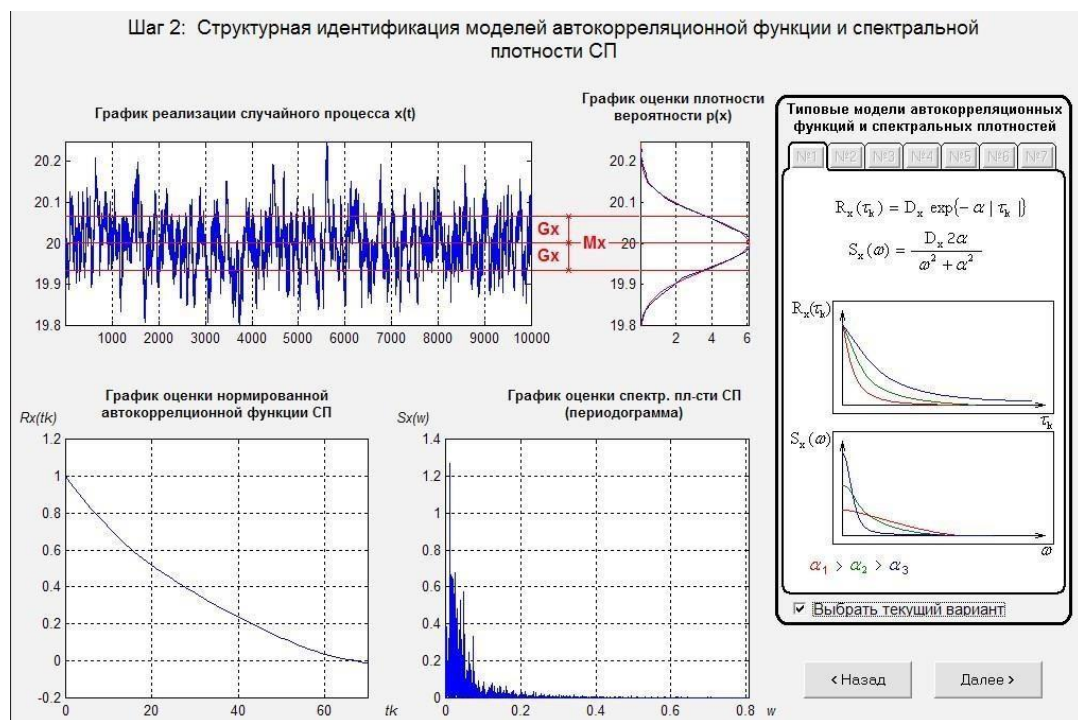


Рис. 2.3.5 Структурна ідентифікація автокореляційної функції та спектральної щільності процесу  $W_M(t)$

2.3.7 Параметрична ідентифікація моделей імовірнісних властивостей випадкових складових, оцінка результатів ідентифікації.

На третьому кроці проводимо параметричну ідентифікацію параметра  $\alpha$  автокореляційної функції процесу  $T_n(t)$   $R_{T_n}(\tau_k) = D_{T_n} \cdot e^{-\alpha \cdot \tau_k}$ .

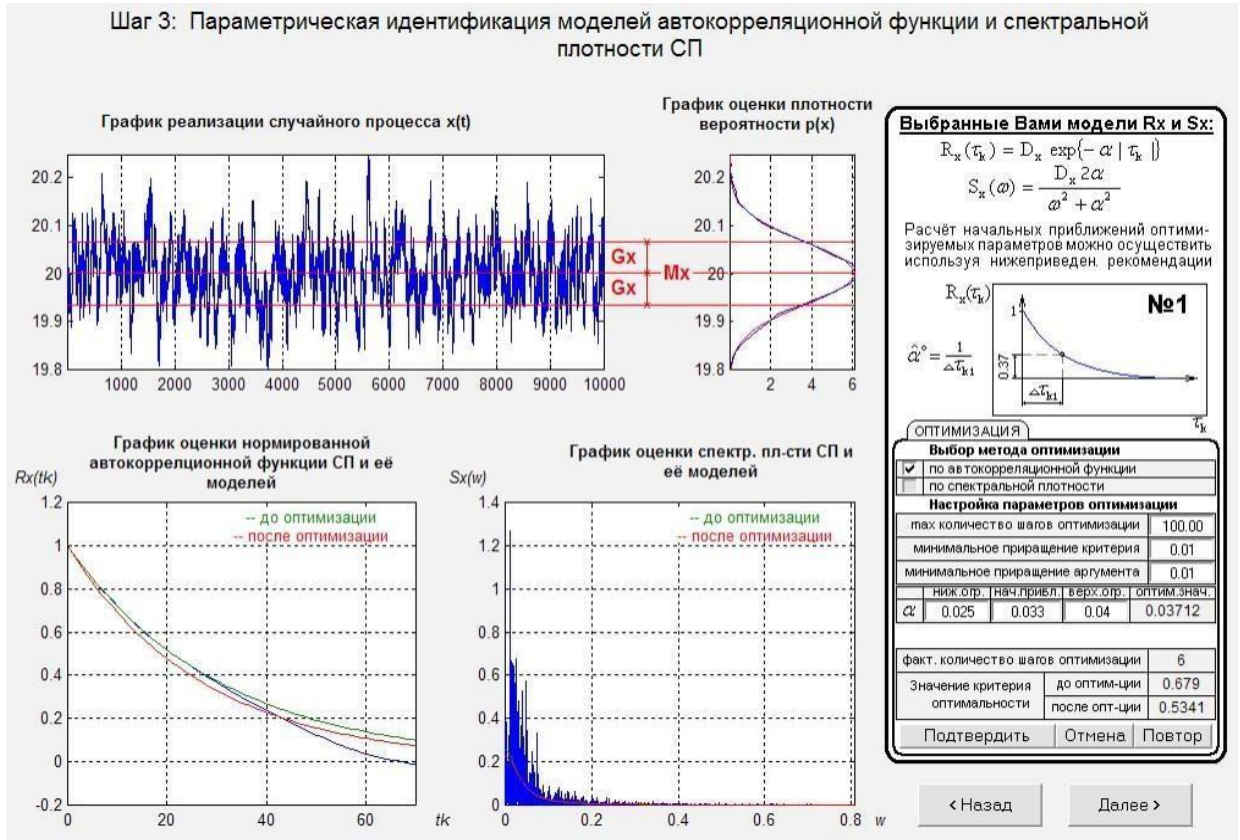


Рис. 2.3.6 Результаты параметричної ідентифікації АКФ процесу  $W_M(t)$

На 4 кроці отримуємо кінцеві результати ідентифікації випадкової складової процесу  $W_M(t)$ .

Шаг 4: Итоги идентификации моделей случайных процессов

**Таблица результатов идентификации**

Номер модели	$\alpha$	$\beta$	A	C	$\gamma$	Значение критерия оптимизации
<input checked="" type="checkbox"/> 1	0.037125	-	-	-	-	0.2853
<input type="checkbox"/>						
<input type="checkbox"/>						

Рис. 2.3.7 Результати ідентифікації випадкової складової процесу  $W_M(t)$

Ці результати в подальшому дадуть змогу визначити передатну функцію формуючого фільтру  $W_f(p)$ , який із псевдобілого шуму, що продукує генератор

## «РЕАЛІЗАЦІЯ МОДЕЛЕЙ У СЕРЕДОВИЩІ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ І ПІДТВЕРДЖЕННЯ ЇХ ВІДПОВІДНОСТІ ЕКСПЕРИМЕНТАЛЬНИМ ДАНИМ»

2.4 Реалізація у середовищі імітаційного моделювання моделей каналів перетворення дій і підтвердження їх відповідності експериментальним даним

2.4.1 Розробка схем моделювання динаміки каналів перетворення дій, отримання перехідних характеристик моделей і їх порівняння з експериментальними перехідними характеристиками

У цьому підрозділі роботи в пакеті Simulink середовища Матлаб для каналу «U-  $W_m$ » ОК розробляються схеми моделювання як для моделі 1-го, так і для моделі 2-го порядку.

При моделюванні вихід моделей виводиться на графік в одній системі координат з експериментальними даними. Для цього експериментальні дані попередньо готують в табличному виді і реалізують їх у компоненті Look-Up Table (див. табл. 2.4.1). Схеми моделювання каналів управління ОК наведені на рис. 2.4.1. Результати моделювання наведені на рис. 2.4.2.

Табл. 2.4.1 Дані обробки експериментальної перехідної характеристики ОК по каналу U –  $\varphi_{пр}$

t, c	0	0,5	1,0	1,5	2,0	3,0	4,0	5,0	6,0
$W_m(t)\%$	20,0	20,0	20,39	21,1	25,75	40,7	58,95	75,32	98,1

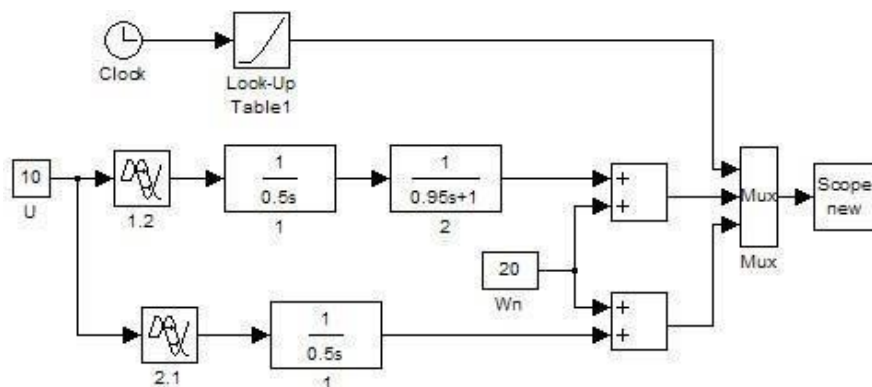


Рис. 2.4.1 – Схема моделювання каналу «U-  $W_m(t)$ »

$W_m, \text{ кг}$

2 пор.

Експ.

1 пор.

t, с

Рис. 2.4.2 – Результати моделювання ОК по каналу «U -  $W_m(t)$ »

Як видно з рис. 2.4.2, моделі ОК як 1-го, так і 2-го порядків досить точно описують експериментальні дані.

2.4.2 Представлення моделей випадкових складових дій у виді послідовно включених генератора базового випадкового процесу („білого шуму”) і формуючого фільтра

Для відтворення моделі неконтрольованих збурень як стохастичних процесів (СП) із заданими властивостями будемо використовувати метод формуючому фільтру.

БВП  $z(t)$   $W_f(p)$   $f_n(t)$

Рис. 2.4.3 – Структурна схема моделі формування неконтрольованого збурення

Спочатку треба визначити передаточну функцію формуючого фільтра  $W_f(p)$ . Для моделі кореляційної функції №1 передатна функція формуючого фільтра має вигляд:

$$W_f(p) = \frac{k}{T_f p + 1};$$

де:

$$k_f = 2 \cdot \sqrt{\frac{D}{\alpha}} = 2 \cdot \sqrt{\frac{0,004247}{0,037125}} = 0,67; \quad T_f = \frac{1}{\alpha} = \frac{1}{0,037125} = 26,9 \text{ c};$$

$D_f = D_{W_M}$  – дисперсія процесу  $W_M(t)$ . У підсумку:

$$W_f(p) = \frac{k_f}{T_f p + 1} = \frac{0,67}{26,9p + 1} \quad (1)$$

2.4.3 Розробка структурних схем моделювання, генерація випадкових складових моделей вхідних дій, параметрична ідентифікація їх імовірнісних характеристик і підтвердження відповідності моделей експериментальним даним

Для перевірки коректності визначення параметрів передатної функції (1) проведемо моделювання за такою схемою:

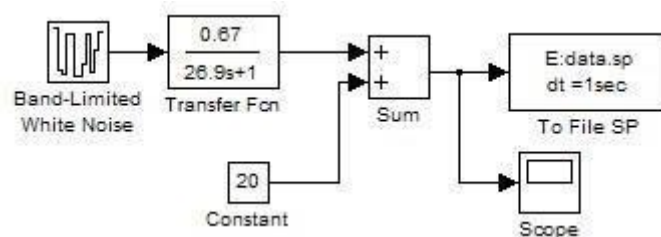


Рис. 2.4.4 Схема моделювання для перевірки коректності визначення параметрів передатної функції  $W(p)$

Далі було проведено оцінювання імовірнісних характеристик випадкових складових координатних збурень. По його результатам виявилось, що похибка в оцінках математичного очікування, середньоквадратичного відхилення і коефіцієнту спаду кореляційної функції моделі відрізняється від аналогічних експериментальних показників більша, ніж на 5%. Тому схема моделювання була скоригована:

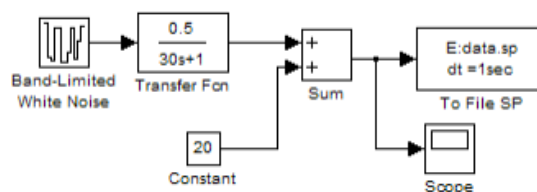


Рис. 2.4.5 Скоригована схема моделювання для перевірки коректності визначення параметрів передатної функції  $W(p)$

Оцінювання імовірнісних характеристик випадкових складових координатних збурень по схемі моделювання, яка представлена на рис. 4.5, дало наступні результати:

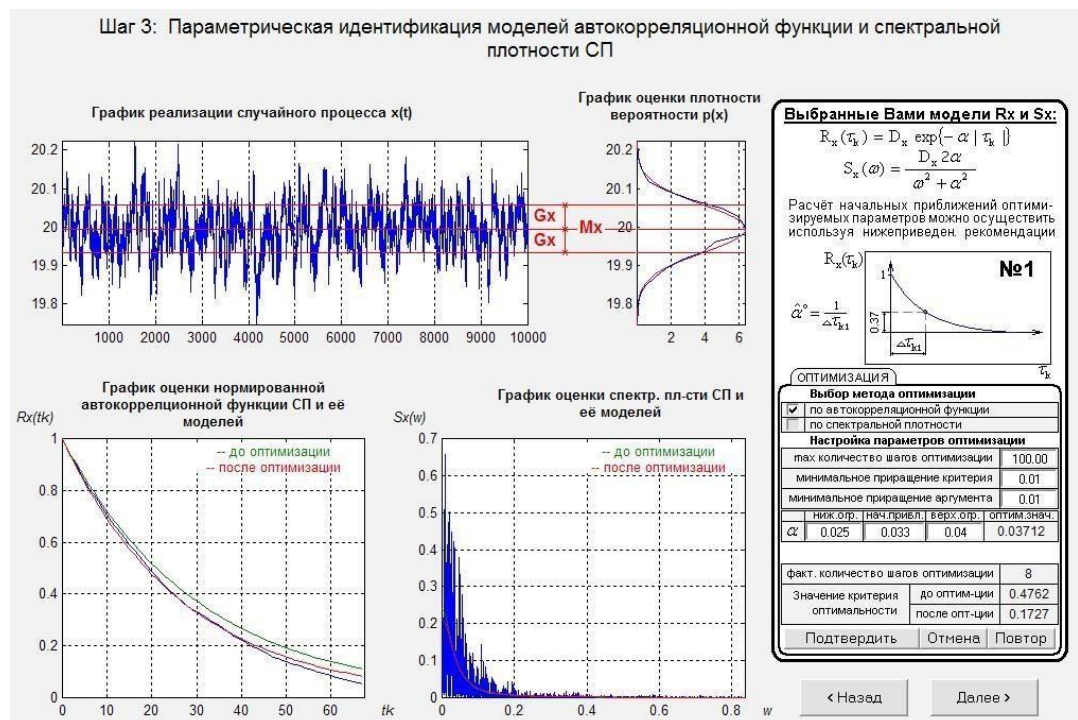
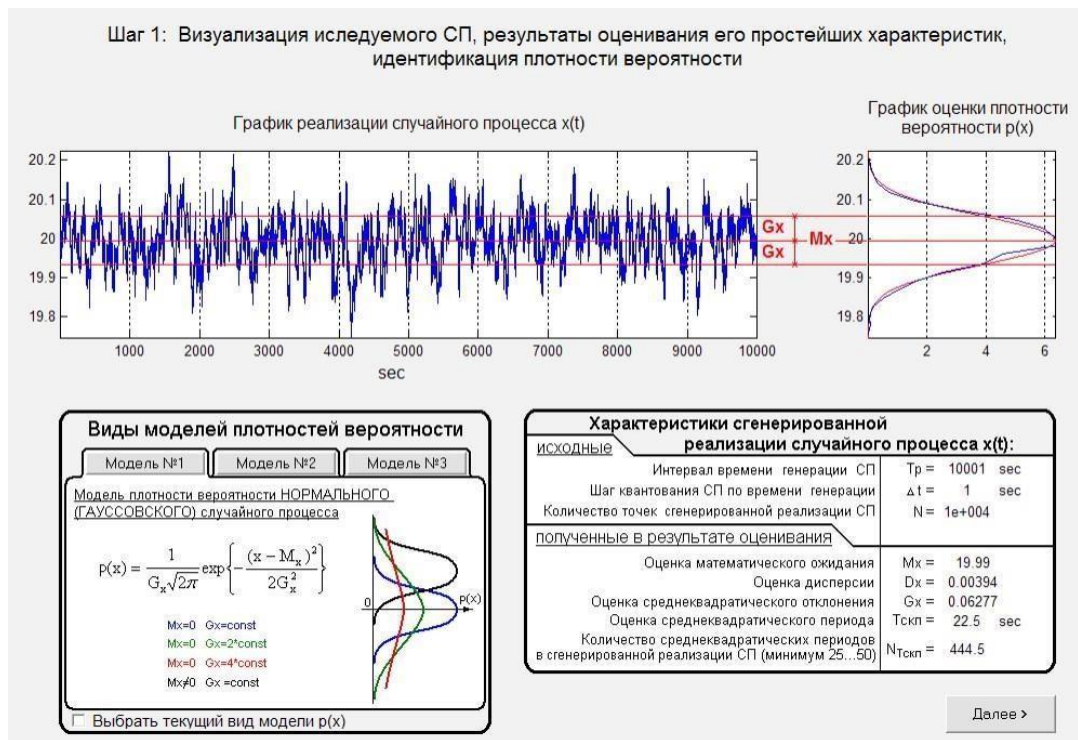


Рис. 2.4.6 Результаты проверки передатной функции формирующего фильтра  $\mathcal{W}(p)$

Порівнюючи отримані результати із аналогічними по п. 3.3, можна дійти висновку, що похибка в оцінках математичного очікування, середньоквадратичного відхилення і коефіцієнту спаду кореляційної функції

моделі відрізняється від аналогічних експериментальних показників менше, ніж на 5%.

2.4.4 Реалізація у середовищі імітаційного моделювання повної моделі об'єкту керування і підтвердження її адекватності

2.4.5. Повна модель ОК містить у собі повну модель каналу керування і модель вхідних дій. Схема моделювання повної моделі ОК наведена на рис. 2.4.7. Експериментальні дані з табл. 2.4.1 були введені в елемент Look-Up Table.

Оскільки в цій моделі крок квантування 0,1 с, а при моделюванні формуючого фільтру неконтрольованих збурень він був 1 с, то коефіцієнт передачі фільтру збільшений у 10 раз.

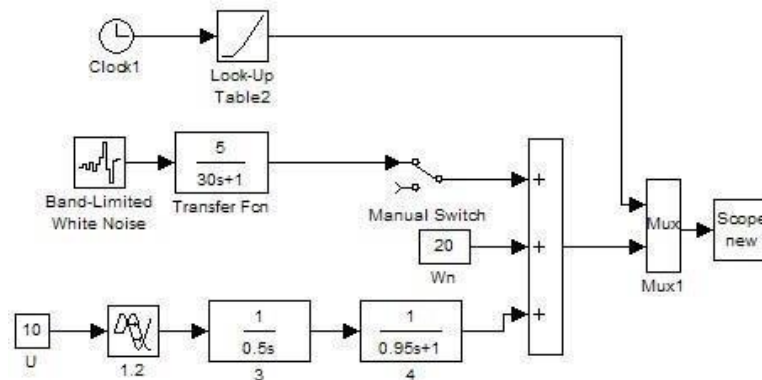


Рис. 2.4.7– Схема моделювання повної моделі ОК

2.4.6 Проведення з моделлю об'єкта машинних експериментів, аналогічних тим натурним, при яких здійснювався збір інформації про властивості об'єкту, порівняння результатів цих експериментів, прийняття висновку про відповідність моделі експериментальним даним.

Проведемо експерименти з повною моделлю ОК. Результати експерименту при одночасній зміні керуючої дії  $U$  детермінованого характеру наведені на рис. 4.6а. Якщо на ОК діють тільки збурення  $f_H(t)$ , то загальний результат приведений на рис. 2.4.9.

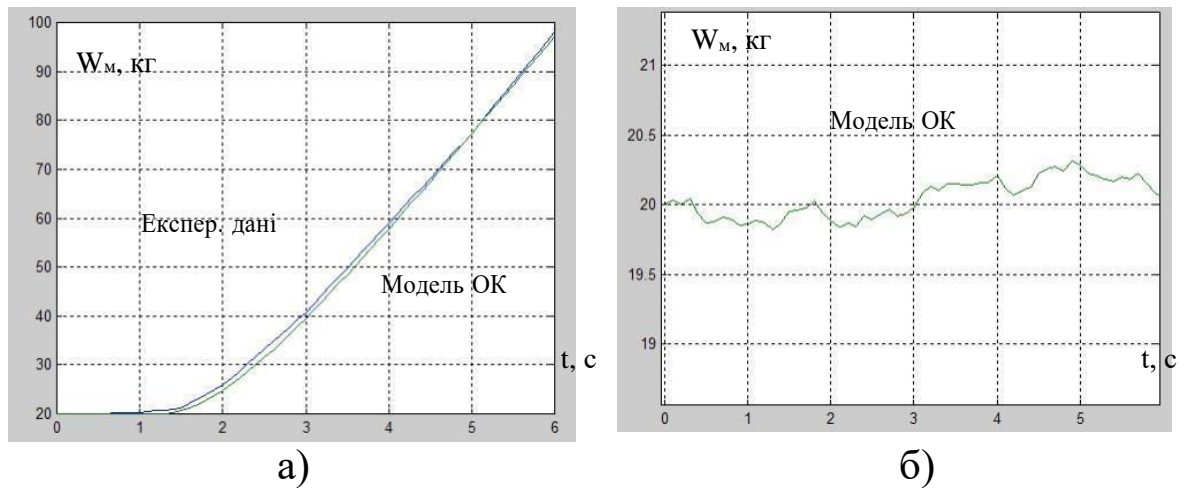


Рис. 2.4.9 – Результати експерименту при зміні керуючих дій  
 $U$  детермінованого характеру 10%х.р.о. та впливі  
 неконтрольованих збурень  $f_H(t)$

Як видно з результатів моделювання, повна модель ОК достатньо точно відображає експериментальні дані з табл. 4.1. Це означає, що отримана повна модель ОК є адекватною.

«ФОРМУЛЮВАННЯ ЗАДАЧ КЕРУВАННЯ, ВИМОГ ДО ЇХ  
РОЗВ'ЯЗКУ І ВИБІР ОСНОВНИХ ПРИНЦИПІВ ПОБУДОВИ  
АВТОМАТИЧНИХ СИСТЕМ»

2.5 Формулювання задач керування технологічним агрегатом

2.5.1 Декомпозиція загальної задачі керування технологічним агрегатом і формулювання приватних завдань.

Загальним завданням керування є керування матеріальними і енергетичними потоками для отримання властивостей продукту необхідного виду і якості.

Із загальної мети керування можна виділити приватні задачі:

- підтримання вихідних змінних об'єкта керування ( $W_m$ ) на їх заданих значеннях ( $W_m^z$ ) - задача регулювання;
- пристосування в процесі роботи об'єкта керування до його змінних властивостей завдання - задача адаптації;
- забезпечення найкращих (оптимальних) в певному (заданому) сенсі режимів роботи об'єкта керування - задача оптимізації;
- забезпечення включення і відключення при пусках, зупинках, аварійних ситуаціях - задача логічного керування.

2.5.2 Обґрунтування необхідності і доцільності автоматизації кожної з приватних задач керування.

Необхідно і доцільно автоматизувати кожну із задач керування.

Для задачі регулювання автоматизація необхідна для автоматичного регулювання заданої змінної в околиці бажаного значення із заданою точністю без фізичної участі людини. Для якісної і економічно доцільної реалізації технологічного процесу фасування цементу необхідно регулювати вагу цементу в мішку  $W_m$ .

Для задачі адаптації необхідна автоматизація пристосування системи керування до змінних властивостей об'єкта керування. Це завдання можна вирішити, змінюючи настройки регуляторів в процесі роботи установки.

Для задачі оптимізації необхідно автоматизувати пошук оптимального

проходження процесу, мінімізувати витрати енергії, брак продукції, збільшити продуктивність обладнання. Для технологічного процесу фасування цементу цю задачу можна вирішити, змінюючи завдання системі регулювання ваги цементу в мішку  $W_M$ .

Процес зважування і фасування цементу є об'єктом періодичної дії. Для нього є характерним режимом пуску і зупинки. Підвищення економічної ефективності роботи цього процесу в режимах пуску і зупинки можна досягти, автоматизувавши ці задачі. Тому для даного технологічного процесу актуальна автоматизація задачі логічного керування.

2.5.3 Формалізація вимог до вирішення задач регулювання і вибір принципу побудови системи автоматичного регулювання

2.5.4 Формалізація вимог до гранично-допустимим статичним і динамічним відхиленням регульованих змінних від відповідних заданих значень перехідних і динамічно сталих процесів і подання їх у формі регламентних зон регульованих змінних.

Відповідно до нормативів ведення технологічних процесів (див. Таблицю 1.2):

- вага цементу в мішку по закінченню фасування  $W_M$  повинна підтримуватися на рівні 50 кг з точністю  $\pm 0,1$  кг. За цим параметром допустимі короточасні відхилення до  $-0,5$  кг протягом часу не більше 30 с. Регламентна зона за цим параметром приведена на рис. 2.5.1.

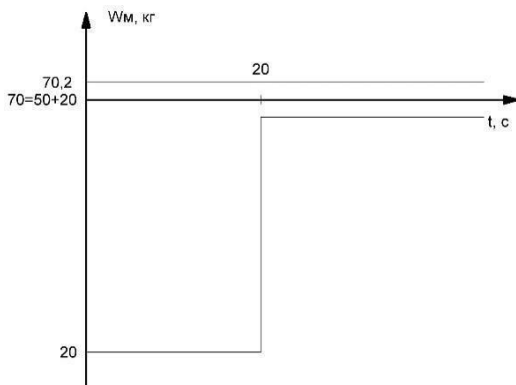


Рис. 2.5.1 – Регламентна зона для  $W_M$

2.5.5 Формалізація інтегральних вимог до перехідних і динамічно сталих процесів регулювання у формі інтегрального критерію оптимальності САР.

В якості інтегрального критерію застосуємо інтегральний модульний критерій зі штрафом швидкості, який забезпечує стійкість САР та аперіодичні перехідні процеси в ній. Тому ефективність роботи САР і підвищення її динамічної точності доцільно вести за таким критерієм:

$$= \int_0^T (|\Delta W_M(t)| + |k \cdot \dot{y}|) dt$$

$\Delta W_m(t)$  – відхилення ваги цементу у мішку від заданого значення;

$k$  – ваговий коефіцієнт.

2.5.6 Вибір, обґрунтування і представлення у вигляді узагальненої структурної схеми принципу побудови системи автоматичного регулювання.

Основу керування становить інформація про цілі керування або про бажаний стан ОК  $y^b$ , про поточний стан ОК  $y$  і про збурення  $f$ . Залежно від обсягу використовуваної керуючим пристроєм (КП) інформації про ОК виділяють чотири принципи побудови САР: принцип розімкнутого жорсткого (програмного) керування; принцип розімкнутого керування за збуренням; принцип замкнутого керування за станом ОК або керування зі зворотним зв'язком; комбінований принцип керування.

Для САР фасування цементу, крім інформації про бажаний стан ОК, є інформація про поточний стан ОК. Такий обсяг інформації для формування керуючого впливу достатній для реалізації замкнутого принципу керування. Тому саме цей принцип і буде покладено в основу розроблюваної САР базової структури. Структурна схема цього принципу керування приведена на рис. 2.5.2.



Рис. 2.5.2 – Структурна схема САР з замкнутим принципом керування

## РОЗДІЛ 3. РОЗРОБКА І ПАРАМЕТРИЧНА ОПТИМІЗАЦІЯ АЛГОРИТМІВ РЕГУЛЮВАННЯ, ПОРІВНЯЛЬНИЙ АНАЛІЗ САР

### 3.1 Конкретизація структури САР і алгоритмів регулювання

3.1.1 Розробка на основі обраного загального принципу побудови САР, її найпростішої конкретної структури.

Структурна схема САР базової структури приведена на рис. 3.1.1. На рис.  $W_{az}(p)$  – передатна функція аперіодичного задатчика;  $W_p(p)$  – передатна функція регулятора ваги цементу у мішку;  $W_{OK}(p)$  – передатна функція ОК;  $W_m^z$  – задане значення ваги цементу у мішку;  $\Delta W_m$  – помилка регулювання;  $U$  – керуюча дія;  $f_H$  – неконтрольовані збурення.

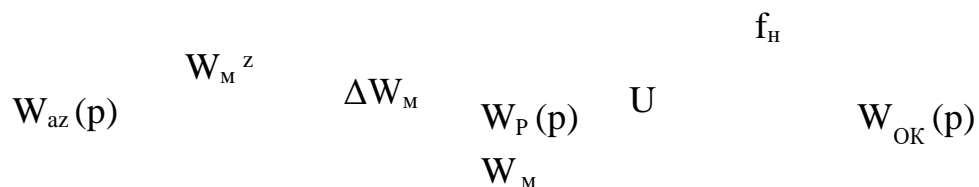


Рис. 3.1.1 – Структурна схема САР процесу фасування цементу базової структури

3.1.2 Вибір кількох альтернативних варіантів типових алгоритмів регулювання, запис їх рівнянь і передаточних функцій.

В інженерній практиці при управлінні технологічними процесами в більшості випадків використовують типові алгоритми регулювання або регулятори. До них відносяться пропорційний (П), пропорційно-інтегральний (ПІ), пропорційно-диференціальний (ПД) і пропорційно-інтегрально-диференціальний (ПІД) алгоритми регулювання.

На процес фасування цементу діють неконтрольовані зовнішні збурення. Щоб компенсувати їх вплив, в якості альтернативних варіантів алгоритмів регулювання обираємо пропорційно-інтегральний (ПІ) і пропорційно-інтегрально-диференціальний (ПІД) алгоритми регулювання.

Рівняння руху ПІ-регулятора

$$u(t) = K \left( \int_0^t \Delta y(t) dt + u \right);$$

Передатна функція ПІ-регулятора

$$W(p) = K \cdot \frac{1}{p}$$

Рівняння руху ПІД-регулятора

$$u(t) = K \left( \int_0^t \Delta y(t) dt + T_{np} \frac{d\Delta y(t)}{dt} + u \right);$$

Передатна функція ПІД-регулятора

$$W_p(p) = K_p \cdot \left( 1 + \frac{T_{np}}{p} + \frac{0,2 \cdot T_{np}}{p+1} \right)$$

3.1.3 Розробка структурної схеми і програми цифрового імітаційного моделювання САР при детермінованих і стохастичних вхідних впливах.

В результаті виконання п2.2.4 була проведена ідентифікація моделей каналів перетворення координатних впливів процесу фасування цементу. Результати ідентифікації зведені в таблицю 6.1.

Табл. 3.1 Результати ідентифікації динамічних властивостей ОК

Канал ОК	Передатна функція 1го порядку	Передатна функція 2го порядку
U- фпр	$W(p) = \frac{e^{-2,1p}}{0,5p + 1}$	$W(p) = \frac{e^{-1,2p}}{0,5p(0,95p + 1)}$

Структурна схема моделювання САР з ПІ-регулятором приведена на рис. 6.3, а з ПІД-регулятором - на рис. 6.4.

Рис. 3.1.2 – Структурна схема моделювання САР з ПІ-регулятором

Рис. 3.1.3 – Структурна схема моделювання САР з ПІД-регулятором

3.1.4 Параметричний синтез САР з різними варіантами типових алгоритмів регулювання та їх порівняльний аналіз для детермінованих вхідних впливів

3.1.5 Вибір початкових наближень параметрів налаштування алгоритмів регулювання на основі інженерних методик параметричного (неоптимального) синтезу САР.

Скористаємося методикою Копеловіча А.П. по розрахунку параметрів налаштування регуляторів. Вибираємо параметри ОК для моделі 1-го порядку та аперіодичний перехідний процес в САР.

Розрахунок параметрів ПІ-регулятора для каналу «U – W<sub>m</sub>»:

$$kr = \frac{0,4 \cdot T_D}{r_o} = \frac{0,4 \cdot 0,5}{2,1} = 0,1 \frac{\% \text{хро}}{\text{кг}}$$

$$T_{iz} = 6 \cdot r_o = 6 \cdot 2,1 = 12,6 \text{ с}$$

Розрахунок параметрів ПІД-регулятора для каналу «U – W<sub>m</sub>»:

$$kr = \frac{0,6 \cdot T_D}{r_o} = \frac{1 \cdot 0,5}{2,1} = 0,14 \frac{\% \text{хро}}{\text{кг}}$$

$$T_{iz} = 5 \cdot r_o = 5 \cdot 2,1 = 10,5 \text{ с}$$

$$T_{pr} = 0,5 \cdot r_o = 0,5 \cdot 2,1 = 1 \text{ с}$$

3.1.6 Параметричний оптимальний синтез САР з альтернативними алгоритмами регулювання для детермінованих (східчастих) вхідних впливів.

Для оптимального параметричного синтезу САР процесу сушіння чаю з ПІ-регулятором будемо використовувати структурну схему моделювання, наведену на рис. 3.1.2. Результати оптимізації налаштувань ПІ-регулятора наведені на рис. 3.1.4.

рис. 3.1.4 – Результати оптимізації налаштувань ПІ-регулятора Для

оптимального параметричного синтезу САР процесу сушіння чаю з ПІД-регулятором будемо використовувати структурну схему моделювання,

наведену на рис. 3.1.3. Результати оптимізації налаштувань ПІД-регулятора

наведені на рис. 3.1.5.

Рис. 3.1.5 – Результати оптимізації налаштувань ПІД-регулятора

3.1.7 Порівняльний аналіз перехідних процесів в параметрично оптимальних системах за значенням критерію оптимальності і показниками, для яких встановлені гранично-допустимі значення.

Для проведення порівняльного аналізу САР з ПІ- і ПІД-регулятором будемо використовувати структурну схему моделювання, наведену на рис. 3.1.6. Результати порівняння варіантів САР наведені на рис. 3.1.3.6.7 і в таблиці 3.1.2.

Рис. 3.1.6 - Структурна схема моделювання для порівняльного аналізу

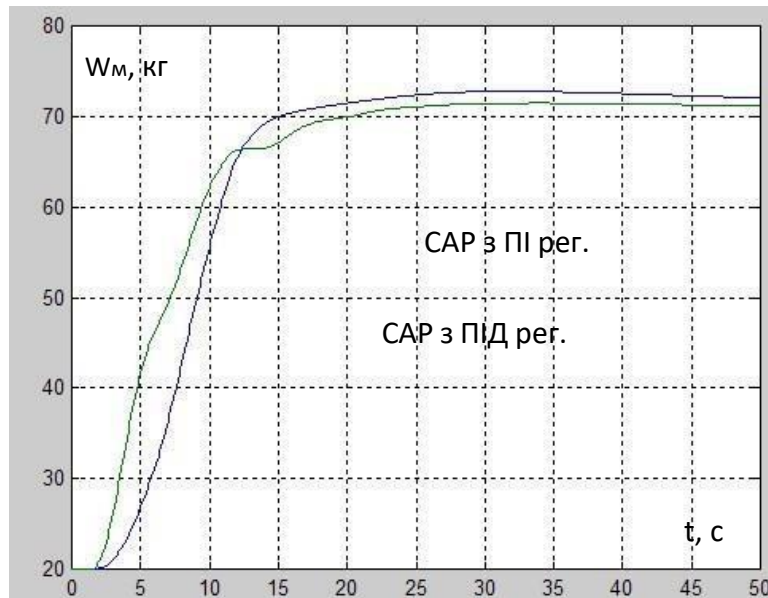


Рис. 3.1.7 – Результати порівняння варіантів САР по каналу «U –  $W_m$ »

Із рис 3.1.7 видно, що в умовах дії збурень детермінованого характеру перехідні процеси в САР з ПІ- та ПІД -регулятором виходять за рамки регламентних зон.

САР з ПІД-регулятором по прямим і інтегральним показникам краща, ніж САР з ПІ-регулятором.

Таблиця 3.1.2 - Результати порівняння варіантів САР

Алгоритм регулювання	Оптимальні параметри регулятора	$\Delta W_m^{\text{МАКС}}$ , кг	$T_{\text{пн}}$ , с	Критерій
ПІ	$kr=0,09$ $Tiz=50$	2,8	22	5672
ПІД	$kr=0,17$ $Tiz=50$ $Tr=2,14$	1,5	16	3992

### 3.1.8 Аналіз грубості САР до варіацій параметрів ОК

Вибір параметрів ОК, по яким доцільно оцінити грубість САР до варіацій їх значень і планування машинного експерименту з оцінки грубості.

В процесі роботи за рахунок зміни параметрів ОК може змінюватися час запізнювання в каналах ОК до  $\pm 20\%$ . При цьому дослідження САР на грубість будемо проводити в умовах дії детермінованих ступінчастих

неконтрольованих збурень при виході САР на задане значення. Структурні схеми моделювання для аналізу САР з ПІ- і ПІД-регуляторами на грубість наведені на рис. 3.1.6. 3.1.3.

3.1.8 Порівняльний аналіз перехідних процесів за критерієм оптимальності та показниками, для яких встановлені гранично-допустимі значення.

Результати оцінки САР з ПІ-регулятором на грубість наведені на рис. 3.1.8, а САР з ПІД-регулятором - на рис. 6.9. Як видно з результатів, САР і з ПІ-регулятором, і з ПІД-регулятором є грубими, оскільки в умовах варіації запізнення в ОК до  $\pm 20\%$  перехідні процеси в САР сходяться до заданих значень.

Рис. 3.1.8 - Аналіз на грубість САР з ПІ-регулятором

Рис. 6.9 - Аналіз на грубість САР з ПІД-регулятором

Однак збільшення запізнення ОК погіршує стійкість САР і показник критерію J. Для обох типів регулятора перехідні процеси в САР виходять за рамки регламентних зон.

3.1.9 Вибір з розглянутих поєднань параметрів ОК найбільш «несприятливих» і «сприятливих» для керування.

Як видно з результатів оцінки САР на грубість, "найгіршим" для керування є більший на 20% час запізнювання ОК, а "найкращим" - найменший.

Схема моделювання САР базової структури з ПД-регулятором для порівняння варіантів з номінальними параметрами, а також з параметрами ОК "найсприятливішими" та "найнесприятливішими" для керування наведена на рис. 3.1.10. Результати порівняння варіантів САР наведено на рис. 6.11 і зведено в таблицю 6.3.

Рис. 3.1.10 Структурна схема моделювання для порівняльного аналізу варіантів САР з різними значеннями часу запізнення ОК



Рис. 3.1.11 – Результати порівняння варіантів САР по каналу « $u - W_M$ » з різними значеннями часу запізнення ОК

Табл. 3.1.3 Результати порівняння варіантів САР базової структури з ПІД-регулятором з різними значеннями часу запізнення ОК

Сполучення параметрів ОК	Прямі показники якості		Критерій
	$\Delta W_M$ , кг	$T_{пп}$ , с	
номінальні	1,5	16	3992
сприятливі	1,5	16	3532
несприятливі	1,5	16	4584

3.1.10 Аналіз сталих процесів в САР при стохастичних вхідних впливах

3.1.11 Для оцінки якості функціонування САР в сталому режимі необхідно

в моделі ОК підключити замість постійної складової випадкову складову неконтрольованого  $f_n(t)$  збурення (див. рис. 4.3).

Для цього підключим фільтр, що формує  $f_n(t)$ , і проведемо моделювання САР по каналу « $f_n - W_M$ » для сприятливих, номінальних і несприятливих параметрах ОК. Схема моделювання САР з ПІД-регулятором приведена на рис. 3.1.12, а результати моделювання - на рис. 3.1.13.

Рис. 3.1.12 - Схема моделювання САР з ПІД-  
регулятором в сталих режимах

Порівнюючи між собою результати моделювання САР (рис. 6.13) при дії зовнішніх збурень випадкового характеру, можна зробити висновок, що середнє значення регульованої змінної в САР відповідає заданому значенню.

Рис. 6.13 - Результати моделювання САР з ПІД-регулятором по каналу « $f_n - W_m$ » в сталих режимах

3.1.12 Порівняльний аналіз сталих процесів для поєднань параметрів ОК, відповідних сприятливим, номінальним і несприятливим властивостями ОК за інтегральним показником і показниками, для яких встановлені гранично-допустимі значення.

Порівняння результатів моделювання проведемо по математичному очікуванню, середньоквадратичному відхиленню, середньоквадратичному періоду і інтегральному критерію.

Таблиця 3.1.4 - Результати оцінки САР з ПІД-регулятором на грубість в сталих процесах за каналом « $f_k - W_m$ »

Показник якості	Значення параметрів ОК		
	номінальні	сприятливі	несприятливі
Математичне очікування	20,0	20,0	20,0
середньоквадратичне відхилення	0,25	0,23	0,27
середньоквадратичний період	2,8	2,6	3,2
інтегральний критерій	21470	21350	21540

#### Висновки по розділу

В даному розділі курсової роботи була побудована САР найпростішої структури за принципом замкнутого керування. В процесі розробки САР в

якості альтернативних регуляторів використовувались ПІ і ПД-регуляторами, так як вони забезпечують практичну нульову похибку статизму для ОК із самовирівнюванням.

При порівнянні прямих, інтегральних та статистичних показників якості можемо зробити висновок, що САР з ПД-регулятором дає їх кращі значення.

При перевірці САР на грубість було відмічено, що САР з ПІ- і ПД-регулятором грубі до варіації параметрів ОК. При збільшенні запізнення в ОК показники якості функціонування САР погіршуються

## СИНТЕЗ І АНАЛІЗ САР ПІДВИЩЕНОЇ ДИНАМІЧНОЇ ТОЧНОСТІ

### 3.2 Структурний синтез САР підвищеної динамічної точності

3.2.1 Аналіз особливостей об'єкта регулювання, що знижують динамічну точність і вибір способів її підвищення за рахунок введення в структуру САР додаткових зв'язків, що забезпечують, наприклад, її інваріантність, автономність, компенсацію запізнювання і т. д.

Причиною недостатньої точності САР можуть бути зовнішні умови: дія інтенсивних контрольованих і неконтрольованих збурень, інтенсивна зміна завдання САР, а також внутрішні особливості ОК і САР (значне співвідношення  $\tau_o / T_o$ , взаємний вплив каналів в багатомірній САР).

В нашому випадку причиною недостатньої динамічної точності є якраз інтенсивна зміна завдання. САР базової структури не в змозі точно відтворити зміни завдання, через що в системі спостерігається похибка відтворення заданого значення ваги цементу в мішку.

Можна вказати на два шляхи подолання цього недоліку САР базової структури.

Перший пов'язаний із можливістю підвищення астатизму регулятора, тобто в структуру, наприклад, ПІД-регулятора можна додати ще одну інтегральну складову.

Другий передбачає створення САР, інваріантних до передісторії змін заданого значення.

3.2.2 Розробка на основі обраних способів підвищення динамічної точності САР її структурної схеми і формулювання (в аналітичній формі) умов, що забезпечують необхідні властивості САР.

Розглянемо більш детально другий спосіб. Структурна схема САР, інваріантна до змін  $u^{зд}$ , наведена на рис. 3.2.1.

Рис. 3.2.1 – Структурна схема САР, інваріантної до передісторії змін заданого значення

3.2.3 Вивід передаточних функцій коригуючих зв'язків, аналіз їх структури з умов фізичної можливості бути реалізованим, приведення до фізично реалізується увазі, подання до формі з'єднання типових динамічних ланок та отримання перехідних характеристик.

Для виконання умов інваріантності необхідно, щоб передатна функція по каналу завдання  $=1$ :

$$W_{zd,y}(P)=1. \quad (1)$$

Розкриємо цю передатну функцію:

$$W_{zd,y}^{ck}(P) = \underbrace{W^{co}(P)}_{1 \text{ й канал}} + \underbrace{W^K(P) \cdot W^{co}(P)}_{2 \text{ й канал}} =$$

$$\frac{W^p(P) \cdot W^o(P)}{1 + W^p(P) \cdot W_y(P)} + \frac{W^K(P) \cdot W^p(P) \cdot W^o(P)}{1 + W^p(P) \cdot W_y(P)}$$

Для виконання умови (1) необхідно, щоб чисельник був рівен знаменнику, тобто:

$$\left(1 + W^K(P)\right) \cdot W^p(P) \cdot W^o(P) = 1 + W^p(P) \cdot W^o(P);$$

$$W^p(P) \cdot W^o(P) + W^K(P) \cdot W^p(P) \cdot W^o(P) = 1 + W^p(P) \cdot W^o(P);$$

$$W^K(P) \cdot W^p(P) \cdot W^o(P) = 1 \Rightarrow W^K(P) = \frac{1}{W^p(P) \cdot W_y(P)}$$

Для ПІД-регулятора і моделі ОК першого порядку можна отримати:

$$W^K(P) = \frac{1}{W^p(P)W_y(P)} = \frac{K_p}{(1+T_zP + T_{np}T_zP^2)e^{\tau_{uy}P}}$$

Як бачимо, коректуючий зв'язок є диференціюючим і фізично нездійсненним. Його можна спростити і привести до фізично здійсненого вигляду. Для нашого випадку передатна функція коригуючого зв'язку набуває вигляду  $W(p) \approx$

3.2.4 Розробка структурної схеми і програми цифрового імітаційного моделювання та параметричний оптимальний синтез САР підвищеної динамічної точності для детермінованих (східчастих) вхідних впливів.

Схема моделювання САР підвищеної динамічної точності (САР ПДТ) для оптимального параметричного синтезу приведена на рис. 3.2.2., а результати оптимізації - на рис. 3.2.3.

Рис. 3.2.2. – Схема моделювання САР  
ПДТ для оптимального параметричного  
синтезу

Рис. 3.2.3. - Результати оптимізації параметрів САР ПДТ

3.2.5 Аналіз грубості САР підвищеної динамічної точності до варіацій параметрів об'єкта регулювання

3.2.6 Вибір параметрів ОК, по який доцільно оцінити грубість САР значень їх

варіацій і планування машинного експерименту з оцінки грубості.

Як і у випадку з САР базової структури, САР підвищеної динамічної точності досліджуємо на грубість в умовах варіації параметрів ОК, а саме, часу запізнювання ОК  $\pm 20\%$ .

7.3.2 Порівняльний аналіз перехідних процесів за критерієм оптимальності та показниками, для яких встановлені гранично-допустимі значення.

Результати перевірки САР підвищеної динамічної точності на грубість в умовах варіації параметрів ОК наведені на рис. 7.4. Як видно з результатів оцінки на грубість, в умовах варіації параметрів ОК САР ПДТ дає східні перехідні процеси, а значить, САР є грубою.

Рис. 3.2.4 – Аналіз САР ПДТ на грубість в умовах варіації параметрів ОК

3.2.7 Аналіз сталих процесів в САР при стохастичних вхідних впливах

3.2.8. Вплив неконтрольованих збурень на показники якості функціонування САР в сталих режимах

Проведемо моделювання САР ПДТ та САР базової структури з ПІД - регулятором в сталих режимах при дії неконтрольованих збурень  $f_n(t)$ .

Рис. 3.2.5. - Схема моделювання САР ПДТ та САР базової структури з ПД-регулятором в сталих режимах при дії неконтрольованих збурень  $f_n(t)$

Рис. 3.2.6. - Результати моделювання САР ПДТ та САР базової структури з ПД-регулятором в сталих режимах при дії неконтрольованих збурень  $f_n(t)$

### 3.2.9 Порівняльний аналіз сталих процесів в параметрично оптимальних САР

Для порівняння якості роботи САР базової структури і підвищеної динамічної точності в сталих режимах поєднаємо в одній таблиці дані з рис. 7.5 і 7.6

Таблиця 3.2.1 - Результати порівняння САР базової структури і САР підвищеної динамічної точності в сталих режимах

Показник якості	Структура САР	
	Базова	Підвищеної динамічної точності
математичне очікування	20,0	20
середньоквадратичне відхилення	0,25	0,3
середньоквадратичний період	2,8	3,1
інтегральний критерій	21470	21730

3.2.10 Порівняльний аналіз САР базової структури і підвищеної динамічної точності

3.2.11 Порівняльний аналіз перехідних процесів параметрически оптимальних САР по величинам критеріїв і показників, на які встановлено гранично-допустимі значення.

Для порівняння якості роботи САР базової структури і підвищеної динамічної точності розроблено структурну схему моделювання, наведена на рис. 7.7. Результати моделювання наведені на рис. 7.8 та в таблиці 7.1.

Рис.3.2.7 – Структурна схема моделювання для порівняння САР базової структури і САР підвищеної динамічної точності в перехідних режимах

$W_m, \text{ кг}$

САР ПДТ

САР з ПД рег.

t, с

Рис. 3.2.8 – Перехідні процеси в САР базової структури і САР підвищеної динамічної точності

Таблиця 3.2.9 - Результати порівняння САР базової структури і САР підвищеної динамічної точності в перехідних режимах

Структура САР	Показники		
	$\Delta W_k^{\text{МАКС}}$	Тпп, с	Критерій
Базова	1,5	16	3992
Підвищеної динамічної точності	1,5	12	4210

### 3.2.12 Порівняльний аналіз грубості параметрично оптимальних САР

Як показують результати досліджень, САР базової структури та САР підвищеної динамічної точності є грубими до варіацій запізнення в ОК.

Висновки по розділу:

За рахунок створення додаткового каналу впливу на систему керування вдалося частково покращити швидкість виходу регульованої змінної на з значенні

### ВИСНОВОК ЗА РОЗДІЛОМ 1.2.3

У ході виконання було встановлено, що якісна реалізація технологічного процесу фасування цементу залежить від точності підтримки ваги цементу в мішку.

Технологічний процес фасування цементу як об'єкт керування має астатичні властивості по каналу керування. Ідентифікація моделей каналів ОК, доступних для цілеспрямованої дії, проводилася методом активного експерименту з нанесенням на вхід каналу 10%-ї ступінчастої дії, для параметричної ідентифікації моделей каналу управління застосовувалися методики двох загальних точок для моделей першого і другого порядку. Статистичними методами була отримана модель стохастичної складової неконтрольованих збурень.

Відтворення отриманих моделей у середовищі імітаційного моделювання здійснювалося в програмі SIMULINK/MATLAB.

Була побудована САР базової структури за принципом замкнутого керування. В процесі розробки САР в якості альтернативних регуляторів використовувались ПІ і ПІД-регулятори.

При порівнянні прямих, інтегральних та статистичних показників якості можемо зробити висновок, що САР з ПІД-регулятором дає їх кращі значення.

При перевірці САР на грубість було помічено, що САР з ПІ- і ПІД-регулятором грубі до варіації параметрів ОК. При збільшенні запізнення в ОК показники якості функціонування САР погіршуються.

За рахунок створення додаткового каналу впливу на систему керування вдалося частково покращити швидкість виходу регульованої змінної на задане значення. Система підвищеної динамічної точності може бути рекомендована до впровадження.

## РОЗДІЛ 4 КОНКРЕТИЗАЦІЯ ЗАДАЧІ І РОЗРОБКА АЛГОРИТМІВ ЛОГІЧНОГО КЕРУВАННЯ ТЕХНОЛОГІЧНИМ ПРОЦЕСОМ

4.1. Короткий опис технологічного процесу з наведенням детальних процедур технологічного пуску, технологічної зупинки та аварійної зупинки у позаштатних ситуаціях

### **ТЕХНОЛОГІЧНИЙ ПРОЦЕС**

**Технологічний процес** - це послідовність дій, які необхідно виконати для виробництва продукції. Технологічний процес може бути складним і включати в себе багато різних операцій. Тому, перед початком виробництва, потрібно провести технологічний пуск.

### **ТЕХНОЛОГІЧНИЙ ПУСК**

**Технологічний пуск** - це послідовність дій, які необхідно виконати для запуску виробництва. Основні процедури технологічного пуску включають:

- Перевірка обладнання: перевірка обладнання на наявність пошкоджень або несправностей.
- Підготовка сировини: перевірка якості сировини та її підготовка до подальшої обробки.
- Налаштування обладнання: налаштування обладнання на правильний режим роботи.
- Тестування: випробування обладнання на працездатність та перевірка якості виробленої продукції.

### **ТЕХНОЛОГІЧНА ЗУПИНКА**

**Технологічна зупинка** - це зупинка виробництва для технічного обслуговування, налаштування або ремонту обладнання. Основні процедури

технологічної зупинки включають:

- Перевірка обладнання: перевірка обладнання на наявність пошкоджень або несправностей.
- Зупинка обладнання: зупинка обладнання та відключення його від електромережі.
- Проведення ремонту: проведення ремонтних робіт або заміна пошкоджених деталей.
- Налаштування обладнання: налаштування обладнання на правильний режим роботи

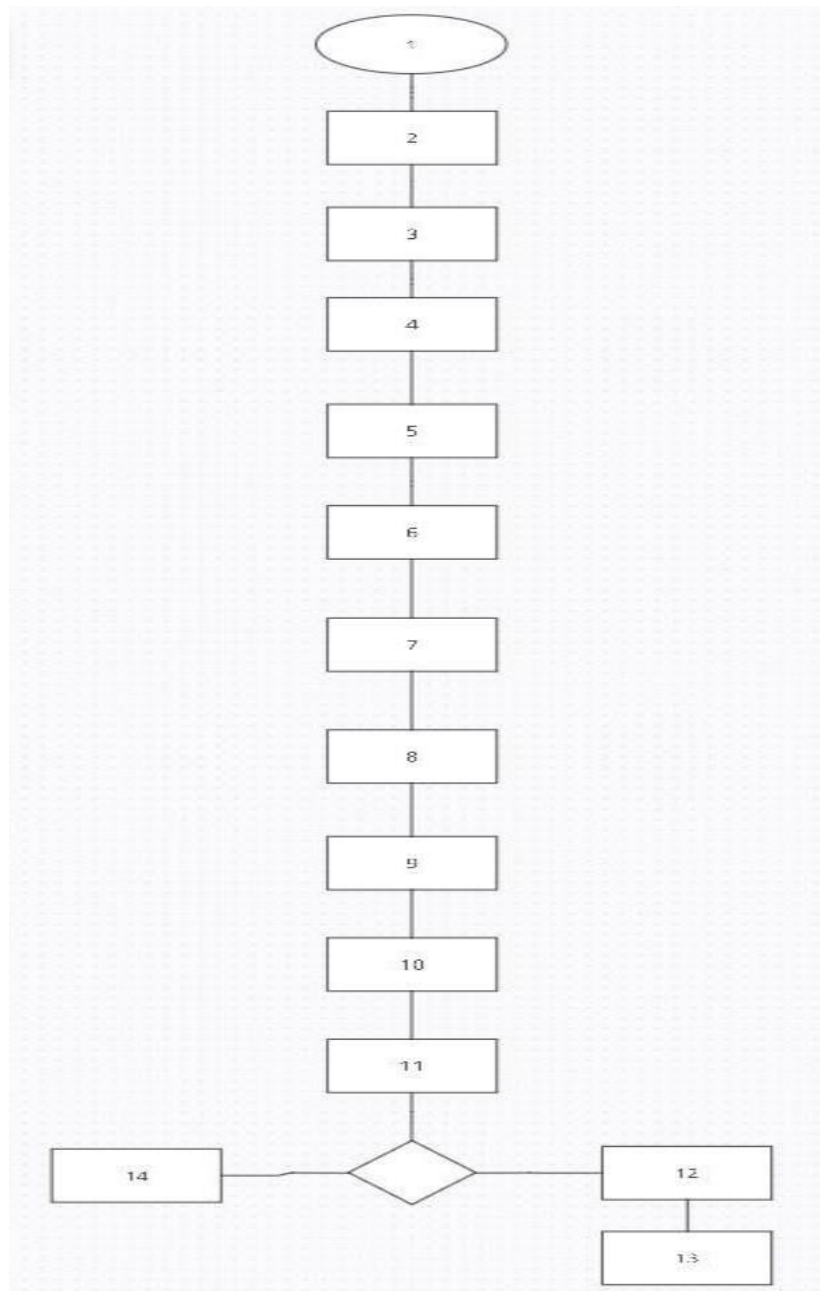


Рис.4.1 – Блок-схема технологічного пуску РТК (початок зміни ).

## **Пояснення та опис алгоритму технологічного пуску :**

1. Оператор перевіряє наявність заготовок.
2. Оператор кладе/позичіонує заготовки до зони розташування заготовок на стіл.
3. Оператор вмикає джерело живлення.
4. Оператор вмикає компресор.
5. Оператор перевіряє регулятор тиску та встановлює потрібний рівень
6. Оператор вмикає РТК
7. Оператор покидає зону РТК
8. Оператор натискає кнопку запуску тестової програми
9. Робот отримує інформацію про старт і починає процес роботи
10. Оператор чекає поки робот завершить роботу.
11. Робот завершує тестову програму
12. Оператор запускає основну програму
13. Робот отримує інформацію про старт і починає процес роботи
14. Оператор вимикає РТК та вирішує питання з приводу проблеми

## **АВАРІЙНА ЗУПИНКА**

**Аварійна зупинка** - це негайна зупинка виробництва у випадку аварійної ситуації. Її основна мета полягає в негайному припиненні руху або роботи всіх механізмів та обладнання в разі виникнення небезпечних ситуацій, щоб запобігти травмам людей, пошкодженню обладнання або надмірному втраті матеріалів.

Її процедура передбачає:

- Негайне вимкнення обладнання та відключення його від мережі живлення.
- Відключення комунікаційних ліній та засобів контролю.
- Звернення до інженерів з питань ремонту та забезпечення безпеки на робочому місці.

працівники повинні бути навчені користуватися аварійними зупинками і бути свідомими про їх місце розташування.

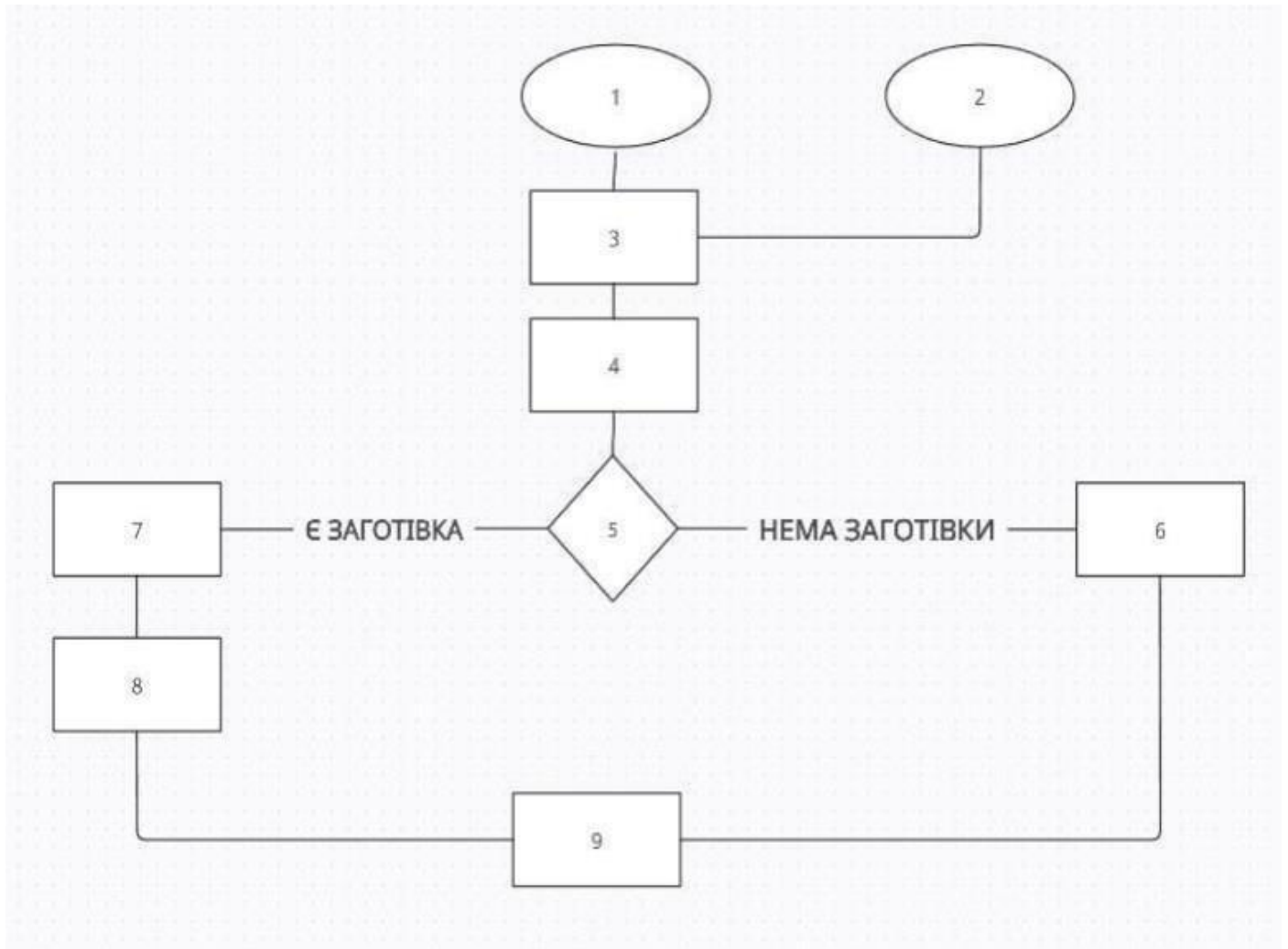


Рис.4.2– Блок-схема аврїйної зупинки **emergency stop** і світлові штори

### Пояснення та опис алгоритму аварійної зупинки :

1. Оператор натиснув на кнопку аварійної зупинки .
2. Система захисту (світлові штори) помічає об'єкт.
3. Подається сигнал до контролеру.
4. Подається звуковий сигнал світлофору та загорється жовтого кольору
5. Система РТК зупиняє роботу.
6. Робот повертається до вихідної позиції

7. Робот викидає заготовівлю у спеціально відведене місце, що передбачає тару для бракованої продукції.
8. Робот повертається до вихідної позиції
9. Оператор запскає основну програму
10. Для продовження праці , оператор запускає систему за інструкцією технологічного пуску РТК.

### **Процес насипу цементу у мішок передбачає:**

1. Підготовка мішків: Мішки для цементу зазвичай мають стандартний розмір та форму , це 25кг. . Мішки зазвичай підготовляються перед насипанням цементу шляхом їх розгортання та перевірки на відсутність пошкоджень.\
  2. Підготовка цементу: Цемент зазвичай зберігається в спеціальних силосах на виробництві. Перед насипанням в мішки цемент потрібно підготувати шляхом його транспортування до місця насипання та перевірки якості.
  3. Насипання цементу: Цемент з силоса транспортується до місця насипання, де він рівномірно насипається в мішки. Насипання здійснюється за допомогою автоматичного обладнання, яке відмірює необхідну кількість цементу для кожного мішка, завдяки ваги.
  4. Закріплення мішків: Після того, як цемент насипаний в мішки, мішки закриваються і закріплюються. Це робиться за допомогою машини для закріплення мішків, яка зшиває мішок та закріплює його за допомогою кліпса або іншого кріплення.
  5. Пакування мішків: Після того, як мішки закріплені, вони готові для пакування. Мішки зазвичай пакують в палети та забезпечують їх безпечну транспортування до складу або клієнта.
- 4.2 Формалізація регламентів функціонування Робототехнічного комплексу (РТК): за пуском, технологічною та аварійною зупинками технологічного процесу у вигляді блок-схем та їх опис.

#### 4.2.1 ВХІДНІ ДАНІ ДІЛЯНКИ

**Опис ділянки:** є ділянка на якій розташована система для насипу цементу у мішок та конвеєр. За неї відповідальні два працівника, які підставляють порожні заготовки для заповнення їх сипучим матеріалом (цемент). Вага мішку з цементом – 25 кг. Алгоритм праці працівників:

1. Взяти порожній мішок.
2. Підставити до системи для насипу цементу.



Рис.4.3 – Ділянка.

СХЕМА ДІЛЯНКИ на рис.4.3:

1. C1\C2 – стіл для заготовок.
2. O1\O2 - система для насипу цементу у мішок.
3. Конвеєр

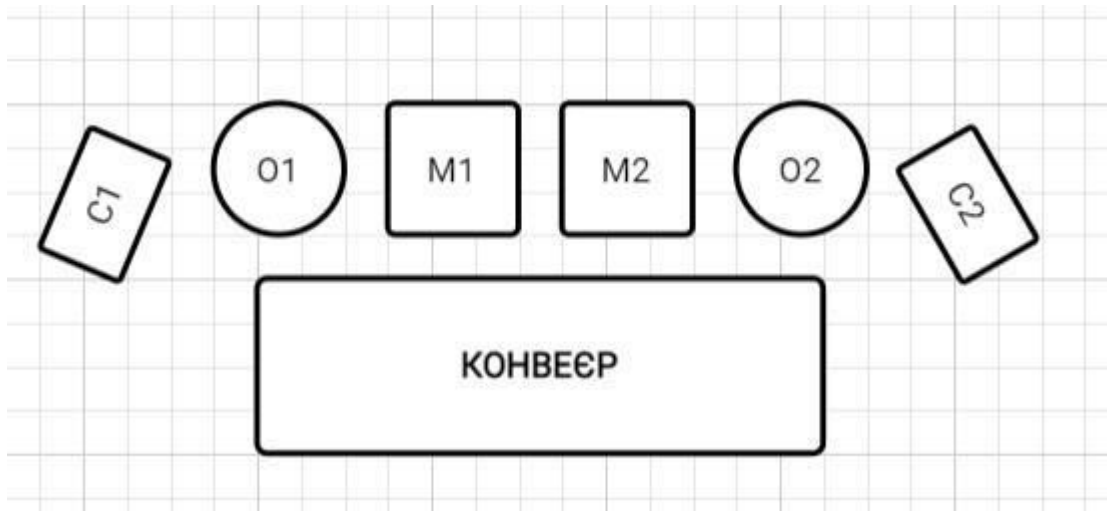


Рис.4.4. – Схема ділянки.

**Ціль РТК:** розробити РТК, що зможе замістити рішення з працівниками.

**Основні вимоги:**

- підвищення ефективності та якості виробництва.
- зменшення витрат на зарплату працівників
- зменшення ризиків для людського здоров'я та безпеки на роботі

## РОЗРОБКА АЛГОРИТМІВ ЛОГІЧНОГО КЕРУВАННЯ ТЕХНОЛОГІЧНИМ ПРОЦЕСОМ

Для розробки рішення , насамперед, я вирішив визначитися з типом і моделю робота, кобот чи промисловий робот. Основна різниця між ними полягає в їх призначенні та способі взаємодії з людьми.

Колаборативні роботи (коботи) призначені для роботи в оточенні з людьми та співпраці з ними. Вони здатні виконувати завдання разом з людьми на основі сенсорних входних сигналів, наприклад, реагувати на дотик або рух людини. Коботизація зазвичай мають більш дрібну конструкцію та меншу масу, ніж промислові роботи, що дозволяє їм працювати поруч з людьми без ризику травмування або пошкодження майна.

Промислові роботи призначені для автоматизації виробничих процесів та виконання рутинних, небезпечних або важких завдань, що зазвичай виконувалися людьми *рис.3*. Вони зазвичай працюють в зоні відокремлення від людей і мають обмежену взаємодію з ними, зазвичай здійснюється через програмне забезпечення або інтерфейси з використанням датчиків безпеки.

Інша важлива різниця між колаборативними та промисловими роботами полягає в їх вартості та складності налаштування. Коботи зазвичай менш складні у використанні та можуть бути більш доступні для підприємств з невеликим бюджетом. В той же час, промислові роботи зазвичай коштують більше та потребують складнішого налаштування та програмування, але можуть мати більшу продуктивність та високу точність виконання завдань.

Спираючись на вимоги від виробництва , я вирішив , що потрібно використовувати промисловий робот, а не колабаротивний.

Для реалізації даної задачі , я розробив РТК на рис.4.5 (роботизований технологічний комплекс), що складається з таких основних елементів, як:

# РТК

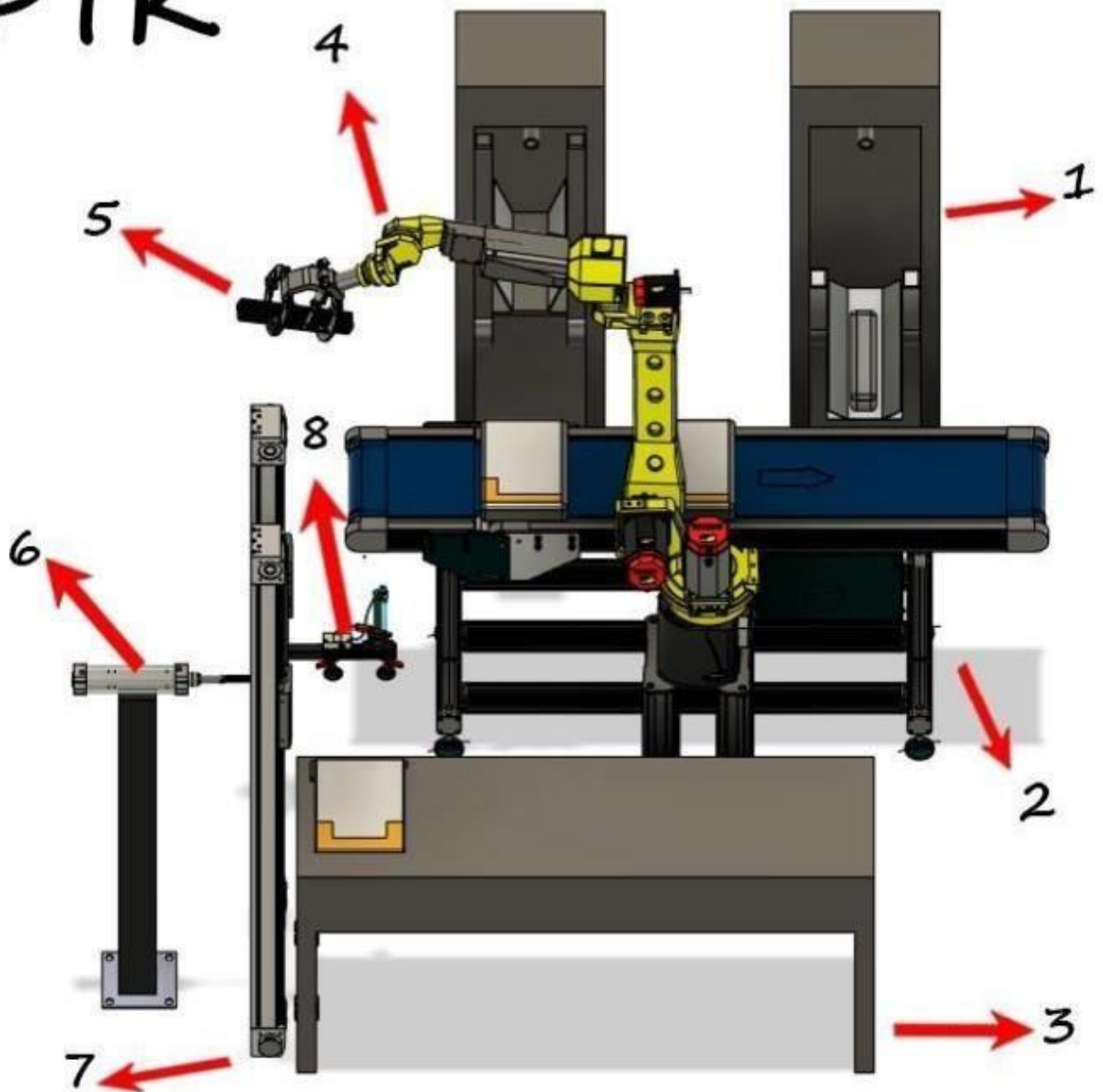


Рис.4.5 – РТК.

## ПОЯСНЕННЯ ОБ'ЄКТІВ РТК

1. Система заповнення мішка та розсіпання цементу. Система заповнення мішка і розсіпання цементу використовується на виробництвах для автоматичного розсіпання цементу або інших сипучих матеріалів

2 Конвеєр. Механічна система, яка використовується для переміщення заповнених мішків з одного місця в інше в промислових, виробничих або логістичних процесах.

Основна мета конвеєра - автоматизувати транспортування мішків з одного місця до іншого, зменшити ручну працю та збільшити продуктивність.

3 Стіл для розташування заготовок (порожніх мішків) Стіл - це предмет, що складається з горизонтальної поверхні, яка підтримується одним або декількома ногами. Він використовується для розміщення **порожніх мішків**.

4 Робот Fanuc M-10iA. Маніпулятор, що дозволяє автоматизувати виробничий процес, що призводить до підвищення продуктивності та зниження залежності від ручної праці. Вони можуть працювати безперервно і з високою швидкістю, забезпечуючи ефективне виконання завдань.

5 Електропневматичний захват. Для робота є пристроєм, який поєднує в собі електричні та пневматичні компоненти для захоплення та утримання об'єктів. Він використовується для виконання різних завдань, таких як збирання, переміщення об'єктів..

6 Пневмоциліндр FESTO. Пристрій, який використовує стиснений повітря для генерації лінійного руху.

7 Лінійний привід FESTO . Лінійний привід FESTO є одним з продуктів, які вони пропонують, і він використовується для створення лінійного руху в автоматизованих системах.

8 Пневматичний захват. Пристрій, що працює на основі пневматичного тиску і використовується для захоплення та утримання предметів під час виробничих процесів.

4.3. Розробка алгоритмів автоматичного керування: за пуском, технологічною та аварійною зупинкою (циклом роботи) ТОК у вигляді блок-схем та їх опис.

1. Світлофор подає сигнал та відтворює , що почала роботу РТК. Про це свідчить червоний колір та звуковий сигнал.



Рис.4.6 – Світлофор для виробництва

2. Піднімається проміжний мішок , завдяки пневматичного захвату, який використовує стиснене повітря для захоплення та утримання об'єкту під час переміщення на виробничій лінії. Рис.4.7

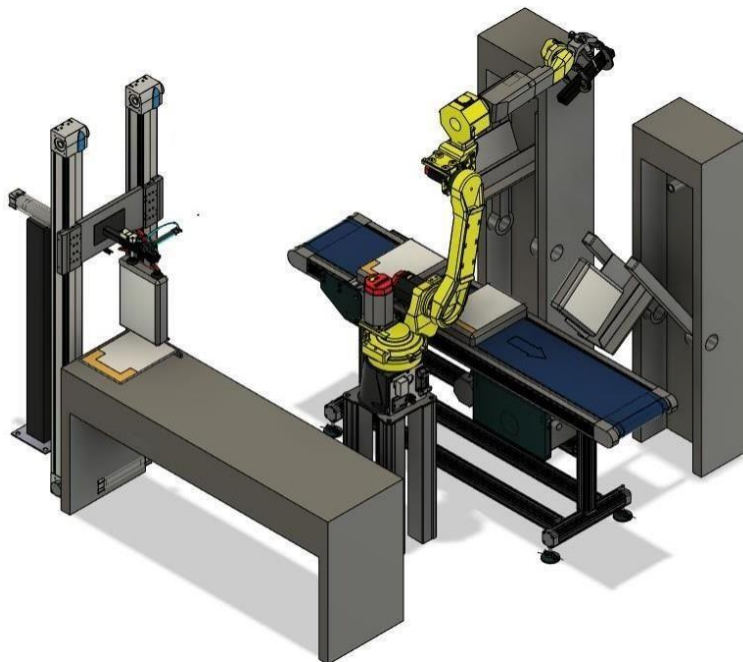


Рис.4.7 – Підняття мішку завдяки пневматичного захвату

3. Позиціонування мішка пневматичним циліндром , що перетворює енергію стисненого повітря на механічну енергію, що використовується для переміщення робочого елемента циліндра. Рис.4.8

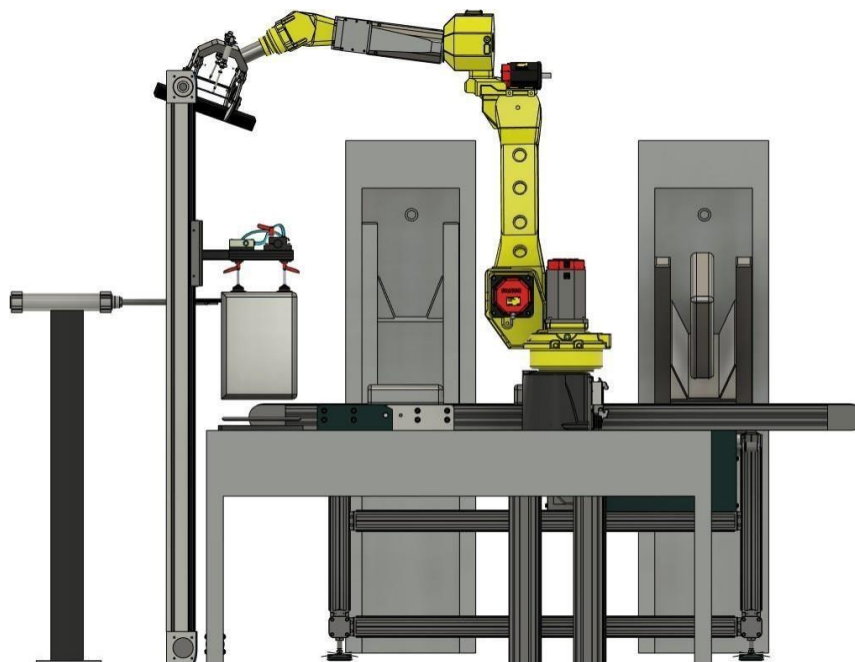


Рис.4.8 – Позиціонування мішка пневматичним циліндром

4. Взяття мішка за допомогою маніпулятора. Рис.4.9

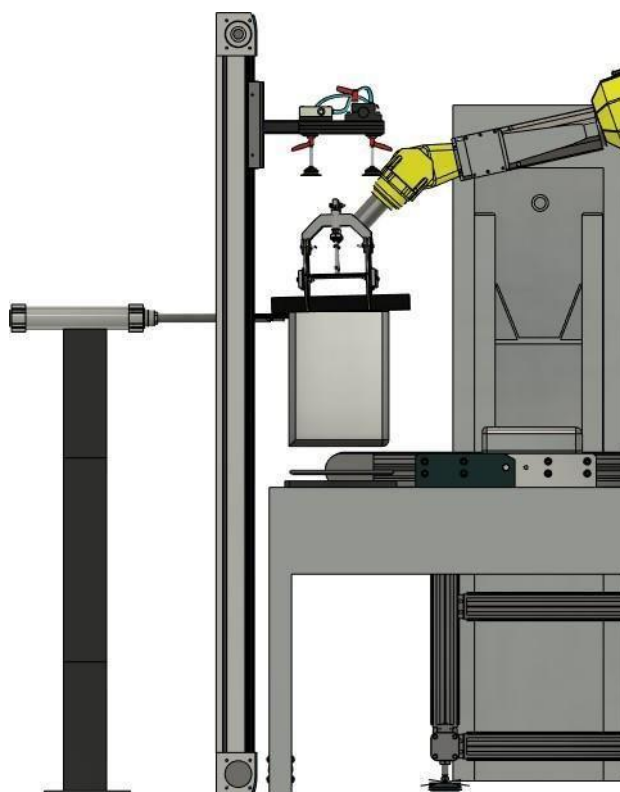


Рис.4.9 – Взяття мішка за допомогою маніпулятора

5. Перенесення мішка, завдяки роботі до системи заповнення мішка та розсіпання цементу. Рис.4.10

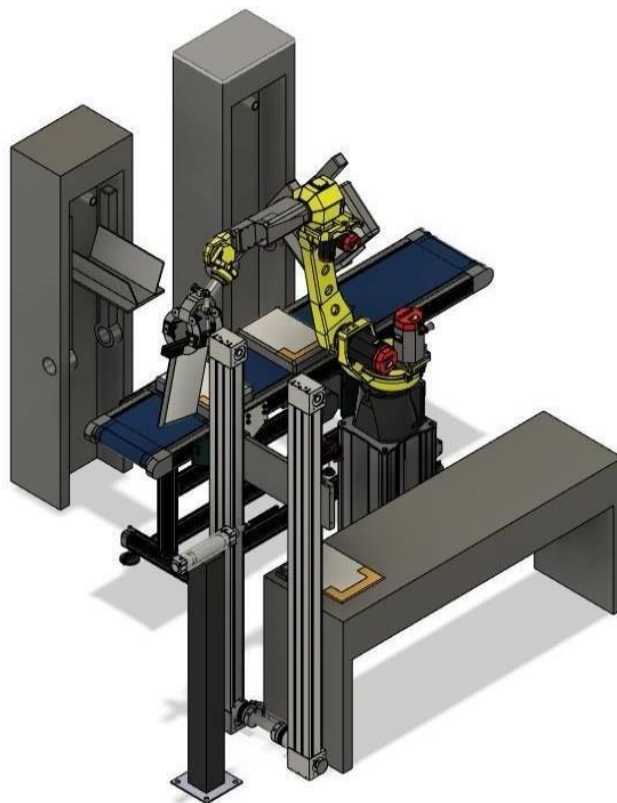


Рис.4.10 – Перенесення мішка

6. Закріплення порожнього мішка до системи заповнення мішка та розсіпання цементу Рис.4.11

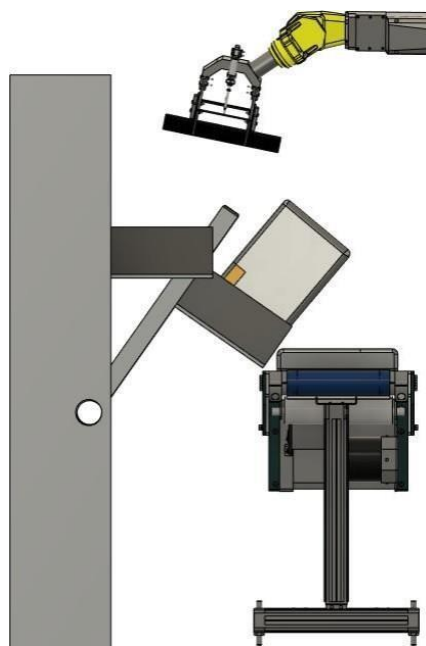


Рис.4.11 – Закріплення порожнього мішка

7. Заповнення мішка цементом.
8. Після закріплення порожнього мішка , система розсіпання цементу починаєзаповнювати матерілом мішок.
9. Як мішок досягає потрібної нам ваги , система зупиняється. Про це свідчить тензодатчик, який вимірює силу, яка діє на нього. Він складається з металевого корпусу і деформуючого елементу, такого як сталевий пружинний стрижень абомембрана. Коли на навантажувальний датчик діє сила, він деформується, і ця деформація перетворюється на електричний сигнал, який можна виміряти.
10. Система опускає наповнений матеріалом мішок на конвеєр.
11. Конвеєр відправляє заготовку на наступну ділянку. Рис.4.12

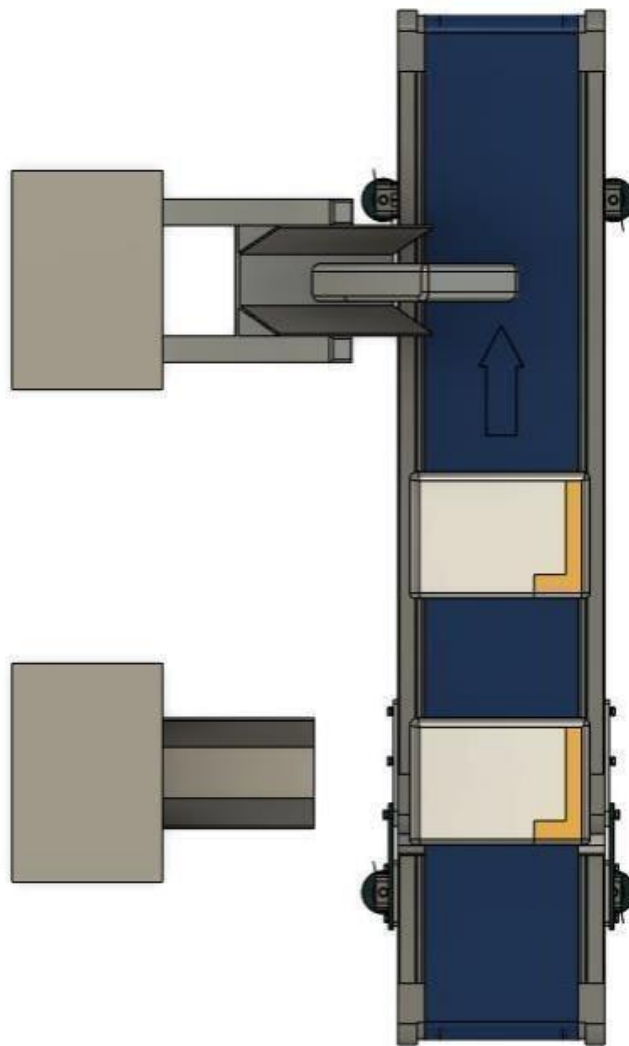


Рис.4.12 – Відправлення мішка конвеєром

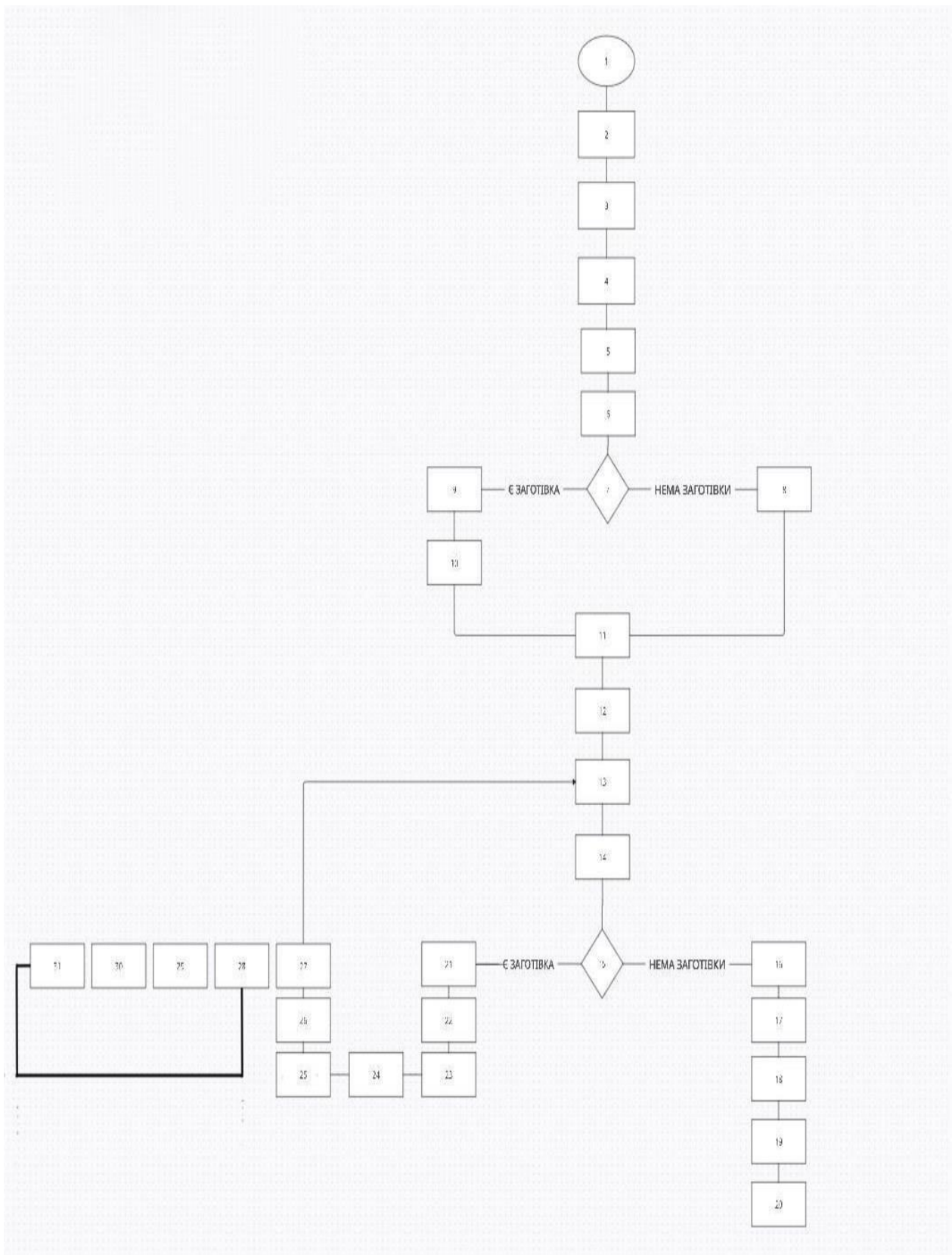


Рис.4.13 – Загальна блок-схема алгоритму праці РТК

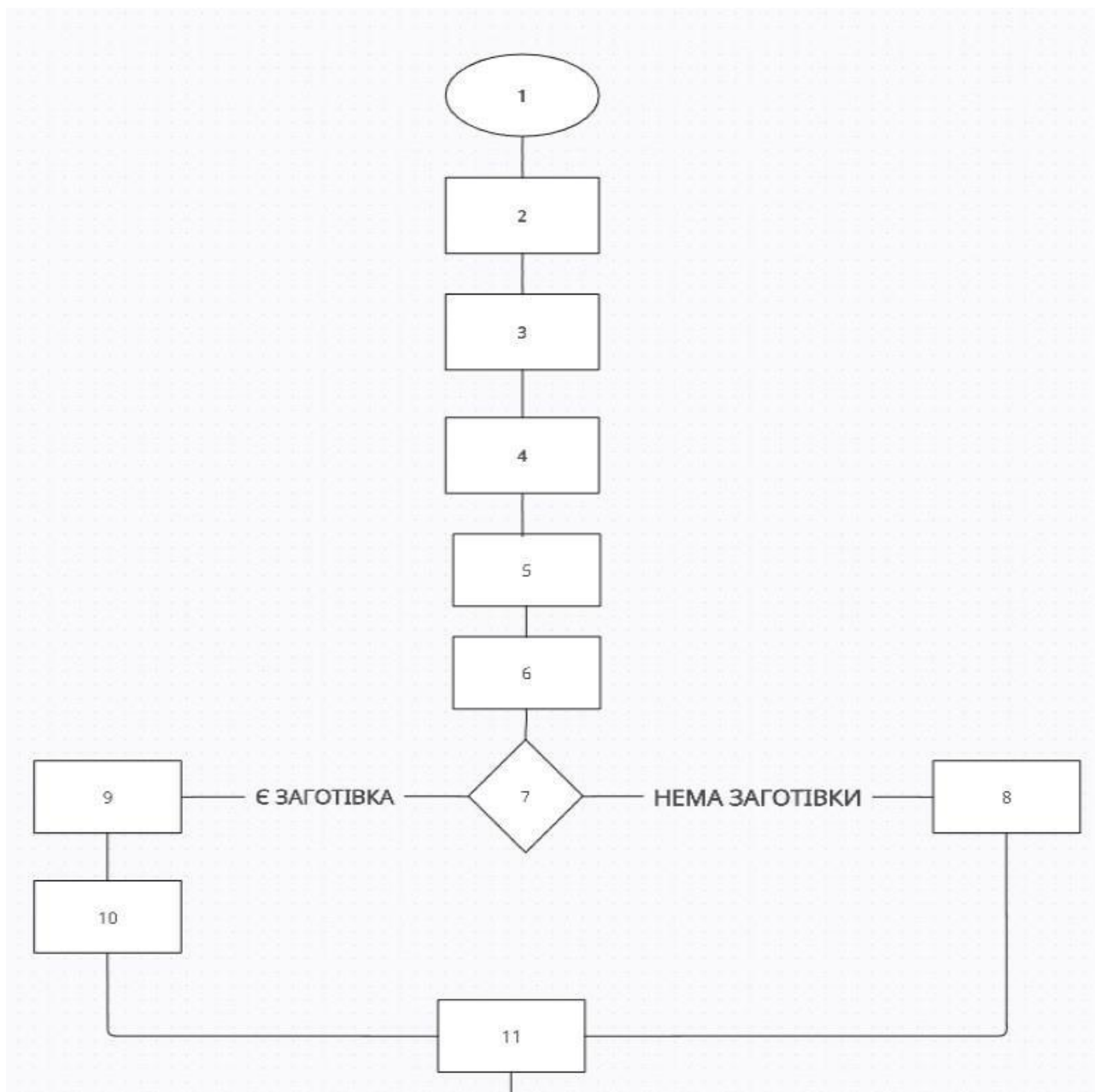


Рис.4.14 – Загальна блок-схема алгоритму праці РТК

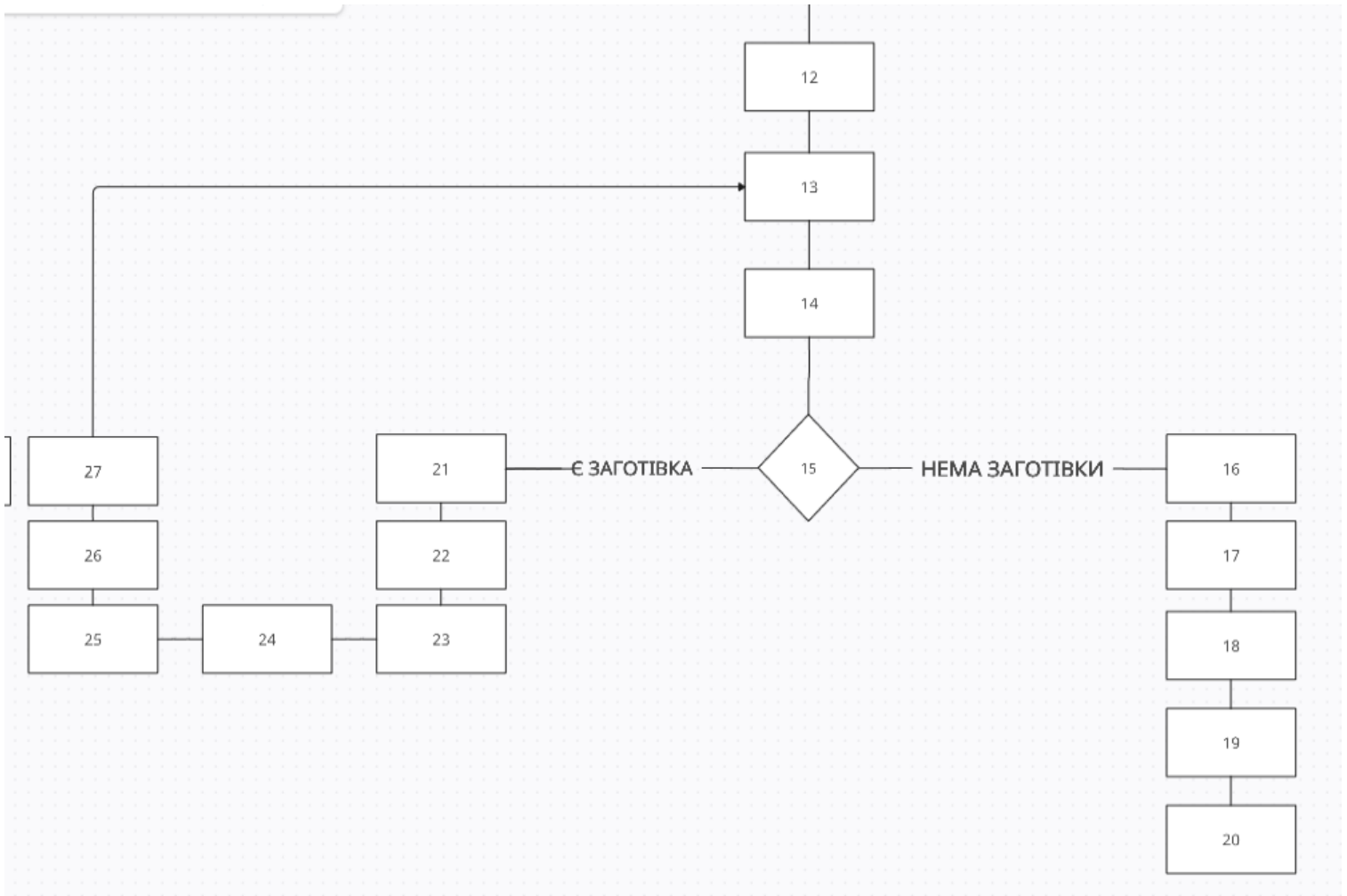


Рис.4.15 – Загальна блок-схема алгоритму праці РТК

### ПОЯСНЕННЯ ТА ОПИС АЛГОРИТМУ АВАРІЙНОЇ ЗУПИНКИ

1. Оператор вмикає РТК
2. Вмикається контроллер та робот
3. Контролер робота очікує розблокування осей 10 секунд
4. Осі розблакованні
5. Вмикається система захисту
6. Подається звуковий сигнал світлофору та загорється жовтого кольору
7. Система РТК зупиняє роботу.
8. Робот повертається до вихідної позиції

9. Робот викидає заготовівлю у спеціально відведене місце, що передбачає тару для бракованої продукції.
10. Робот повертається до вихідної позиції
11. Оператор запускає основну програму
12. Світлофор подає сигнал та відтворює , що почала роботу РТК. Про це свідчить червоний колір та звуковий сигнал
13. Опускається пневматичний захват
14. Вмикається вакуум
15. Аналізує датчик наявності об'єкту
16. Відправляє сигнал до контролеру
17. Вмикається червоний колір світлофору та звуковий сигнал
18. Зупиняється система
19. Робот повертається до вихідної позиції
20. Оператор кладе мішок
21. Захват бере мішок
22. Піднімається проміжний мішок , завдяки пневматичного захвату, який використовує стиснене повітря для захоплення та утримання об'єкту під час переміщення на виробничий лінії.
23. Виїжджає пневмоциліндр для позиціонування мішка .
24. Обертається робот та опускається до мішку
25. Завдяки електропневматичного захвату бере мішок
26. Робот разом з мішком повертається до системи заповнення мішка та розсипання цементу
27. Закріплює мішок до машини
28. Після закріплення порожнього мішка , система розсипання цементу починає заповнювати матеріалом мішок
29. Як мішок досягає потрібної нам ваги , система зупиняється. Про це свідчить тензодатчик, який вимірює силу, яка діє на нього.
30. Світлофор подає сигнал та відтворює , що мішок заповнений. Про це свідчить зелений колір.
31. Отримувач мішку опускається та скидає мішок до конвеєру

## ВИСНОВОК ЗА РОЗДІЛОМ 4

Роботехніка - це галузь, що займається розробкою, конструюванням і використанням роботів для виконання різноманітних завдань у різних сферах життя промисловості. У цьому розділі ми розглянули деякі аспекти роботехніки, що стосуються технологічного процесу, пневматичних захватів та датчиків ваги.

В ході виконання розділу було розроблено та спроектовано роботизований комплекс, алгоритм дій РТК.

Розроблений комплекс включає основні елементи для автономної праці та заміни працівників, а саме: промисловий робот, система заповнення мішка та розсипання цемента, конвеєр, кастомний пневматичний захват, пневмоциліндр FESTO, пневматичний захват, лінійний привід FESTO. В результаті виконання отримані навички з розробки рішення для виробництва, моделювання роботизованих об'єктів.

## РОЗДІЛ 5 ВИБІР ТЕХНІЧНИХ ЗАСОБІВ ОТРИМАННЯ ІНФОРМАЦІЇ ПРО ЗМІННИ ПРОЦЕСУ ТА РЕАЛІЗАЦІЇ КЕРУЮЧИХ ВПЛИВІВ, РОЗРОБКА ТЕХНІЧНОЇ СТРУК-ТУРИ КОНТРОЛЕРНО-КОМП'ЮТЕРНОЇ МЕРЕЖІ

5.1 Характеристики середовищ, з якими взаємодіють засоби автоматизації й обслуговуючий персонал, які повинні бути враховані при виборі технічних засобів і місць їх розміщення

На цементному заводі наступні характеристики середовища, які потрібно враховувати при виборі технічних засобів і місць їх розміщення:

- Висока температура: Виробництво цементу може вимагати високих температур під час процесу випалювання. Тому технічні засоби повинні бути стійкими до високих температур або розміщені в зоні з контрольованою температурою.
- Висока вологість: Процес виробництва цементу може супроводжуватись підвищеною вологістю, особливо під час змішування і подрібнення сировини. Обладнання повинно мати захист від вологи, а також можливість відведення вологи, щоб уникнути корозії та пошкоджень.
- Пил та забруднення: Виробництво цементу може бути супроводжуване утворенням пилу та інших забруднень. Технічні засоби повинні бути відповідно захищені, з можливістю фільтрації повітря та забезпечення чистоти робочих поверхонь.
- Вібрації: На цементному заводі можуть бути великі вібрації від обладнання, такого як дробарки, млини, транспортери тощо. Технічні засоби мають бути відповідно зміцненими та стійкими до впливу вібрацій.

### ВИБІР ТЕХНІЧНИХ ЗАСОБІВ

#### 1. МАНІПУЛЯТОР FANUC СЕРІЯ M-10IA/12

Мій вибір припав на FANUC серія M-10iA/12, це механічний блок, що складається з шарнірів і рук, що рухаються сервомоторами.

Місце з'єднання рук називається шарніром чи ступенем рухливості. Основні ступеня рухливості: J1, J2 та J3. Базова конфігурація робота визначається видом руху (поступальний або обертальний) основних шарнірів.

Шарніри зап'ястя використовують для переміщення робочого органу (інструменту), встановленого на фланці зап'ястя. Саме зап'ястя може обертатися за одним ступенем рухливості зап'ястя, а робочий орган - щодо іншого ступеня рухливості зап'ястя .

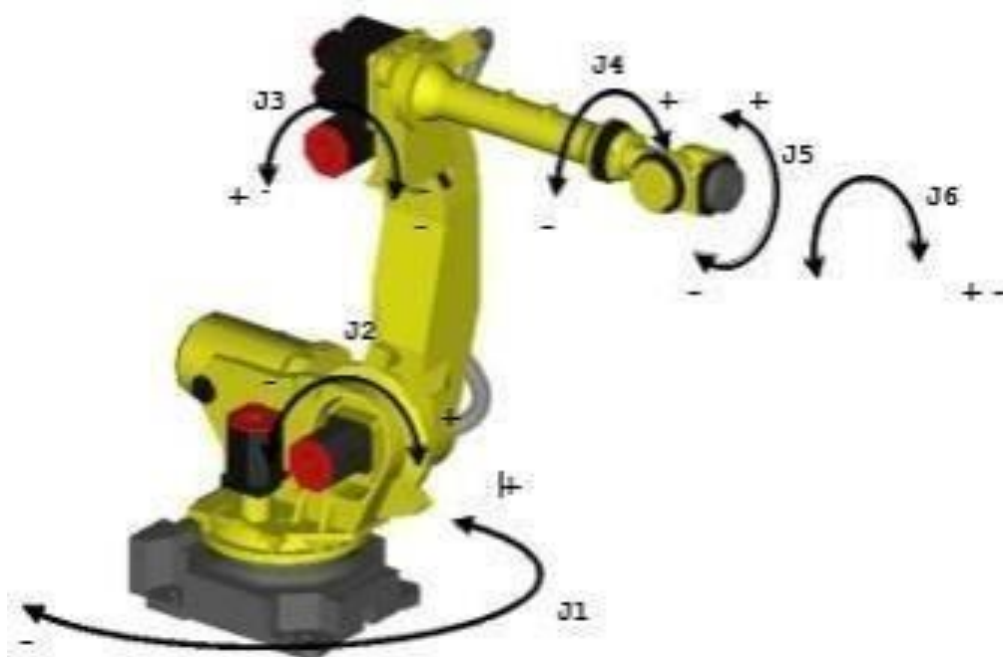


Рис.5.1 – Основні ступеня рухливості та ступеня рухливості зап'ястя

Його технічні характеристики ідеально підходять до рішення задачі та у себе включають наступне :

- Навантаження: FANUC M-10iA/12 може мати різні варіанти по вантажопідйомності, але загальною рисою є його здатність робити легкі або середні навантаження. Варіанти можуть варіюватися від приблизно 7 до 12 кілограмів.
- Радіус дії: Ця серія роботів має радіус дії, що означає, наскільки відстань вони можуть працювати від свого базового положення. Радіус дії FANUC M-10iA/12

може становити близько 1420 мм , що дозволяє нам його розташувати на дануділянку.

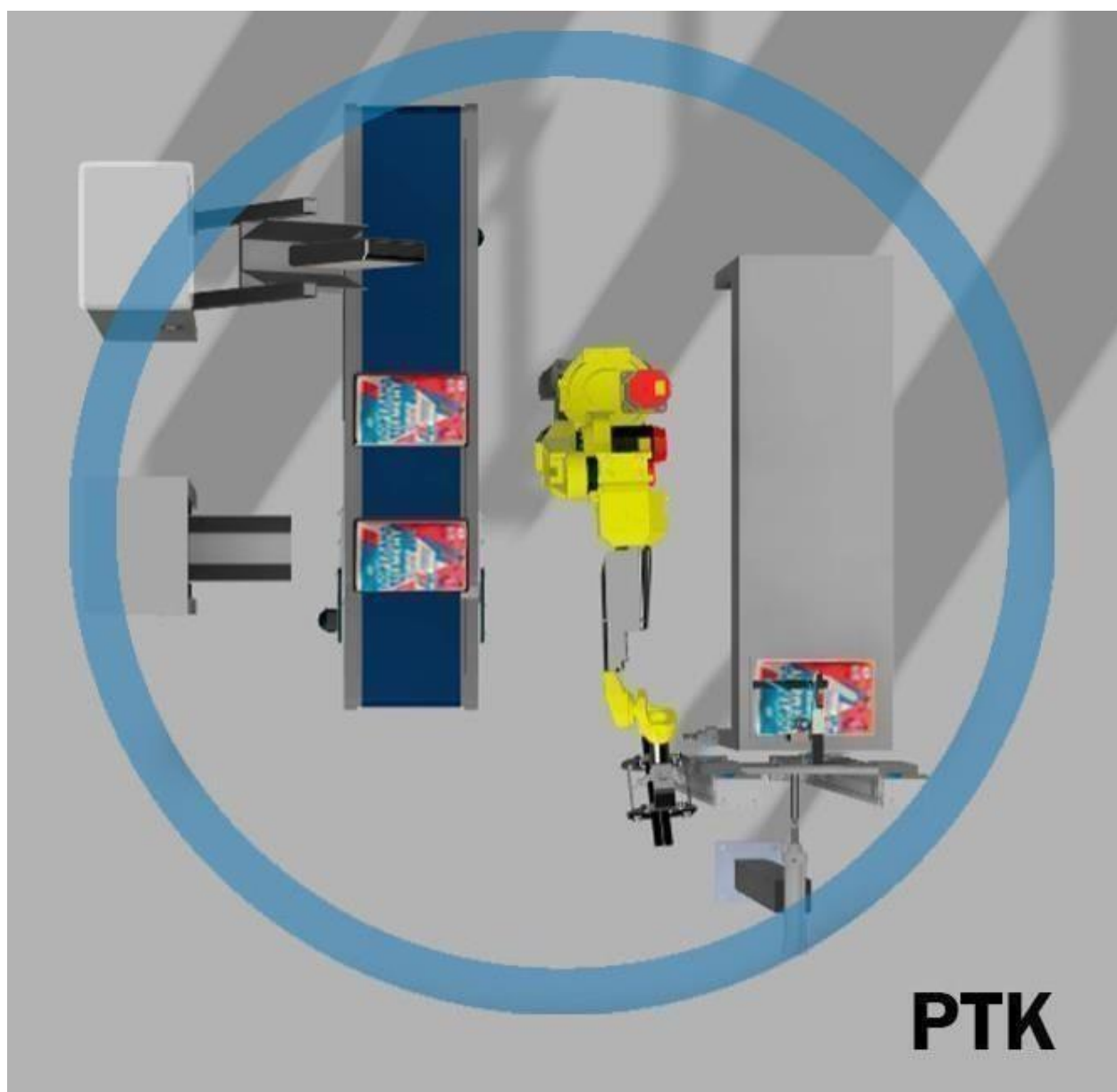


Рис.5.2 – Зона досяжності робота.

- Точність: FANUC M-10iA/12 відомий своєю високою точністю та повторюваністю рухів. Він здатний досягати високої точності позиціонування та повторюваності руху в своєму робочому просторі.

## **2. КОНТРОЛЕР FANUC R-30iB**

Робот FANUC M-10iA керується за допомогою контролера FANUC R-30iB, який забезпечує точне та надійне керування рухом робота. Контролер може мати різні варіанти, залежно від потреби та вимог автоматизаційного процесу.

Контролер робота складається з блоку живлення, схеми інтерфейсу користувача, схеми управління переміщенням, схеми пам'яті і схеми введення-виведення.

Управління контролером здійснюється через пульт навчання та щит управління.

Схема управління управляє сервопідсилювачем, який переміщає всі ступені рухливості робота, включаючи всі додаткові ступені рухливості, за допомогою друкованої плати центрального процесора.

Схема пам'яті використовується для зберігання програм і даних, заданих користувачем оперативної пам'яті КМОП на друкованій платі ЦП.

Схема введення-виведення пов'язує контролер з периферійними пристроями, приймаючи та передаючи сигнали за допомогою модулів введення-виведення (друковані плати введення-виведення). Сигнал дистанційного введення-виведення використовується для зв'язку з контролером дистанційного керування

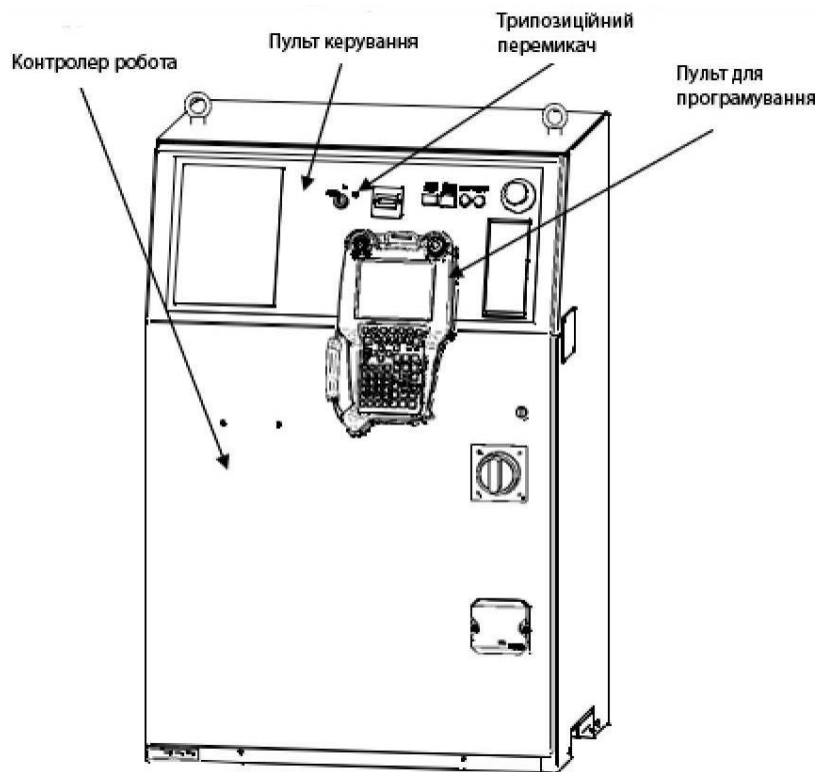


Рис.5.3. – Контролер робота R-30iB

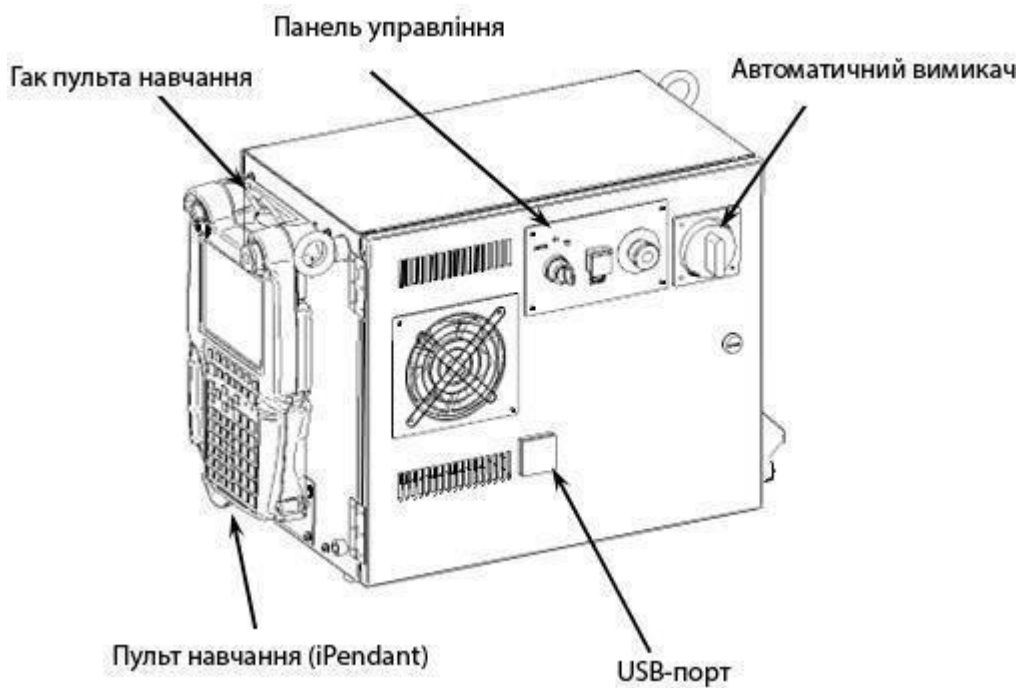


Рис.5.4 – Контролер робота R-30iB

Основні характеристики цього контролера включають наступне:

1. Керування роботом: Контролер R-30iB mate забезпечує точне та надійне керування рухами робота. Він відповідає за обробку команд, керування позиціонуванням, швидкістю та координацією рухів робота.
2. Програмування: Контролер R-30iB mate підтримує різні мови програмування, такі як FANUC Robotics' KAREL або FANUC's TP (Teach Pendant) Programming. Це дозволяє операторам програмувати робота для виконання різноманітних завдань із застосуванням зручного інтерфейсу.
3. Комунікація: Контролер R-30iB mate має вбудовані можливості зв'язку, що дозволяє взаємодіяти з іншими пристроями та системами. Він може бути підключений до комп'ютерів, НМІ (людина-машинний інтерфейс), PLC (програмований контролер логічних схем) та інших пристроїв через різні протоколи комунікації.
4. Висока продуктивність: Контролер R-30iB mate володіє потужними обчислювальними можливостями, що дозволяє йому ефективно керувати роботом та оптимізувати час виконання завдань.

IP66 - рівень захисту забезпечує надійну роботу обладнання в умовах вологості та пилу.

5.2 Характеристики середовищ, з якими взаємодіють засоби автоматизації й обслуговуючий персонал, які повинні бути враховані при виборі технічних засобів і місць їх розміщення та вибір технічних засобів реалізації керуючих дій на технологічний процес і управління обладнанням

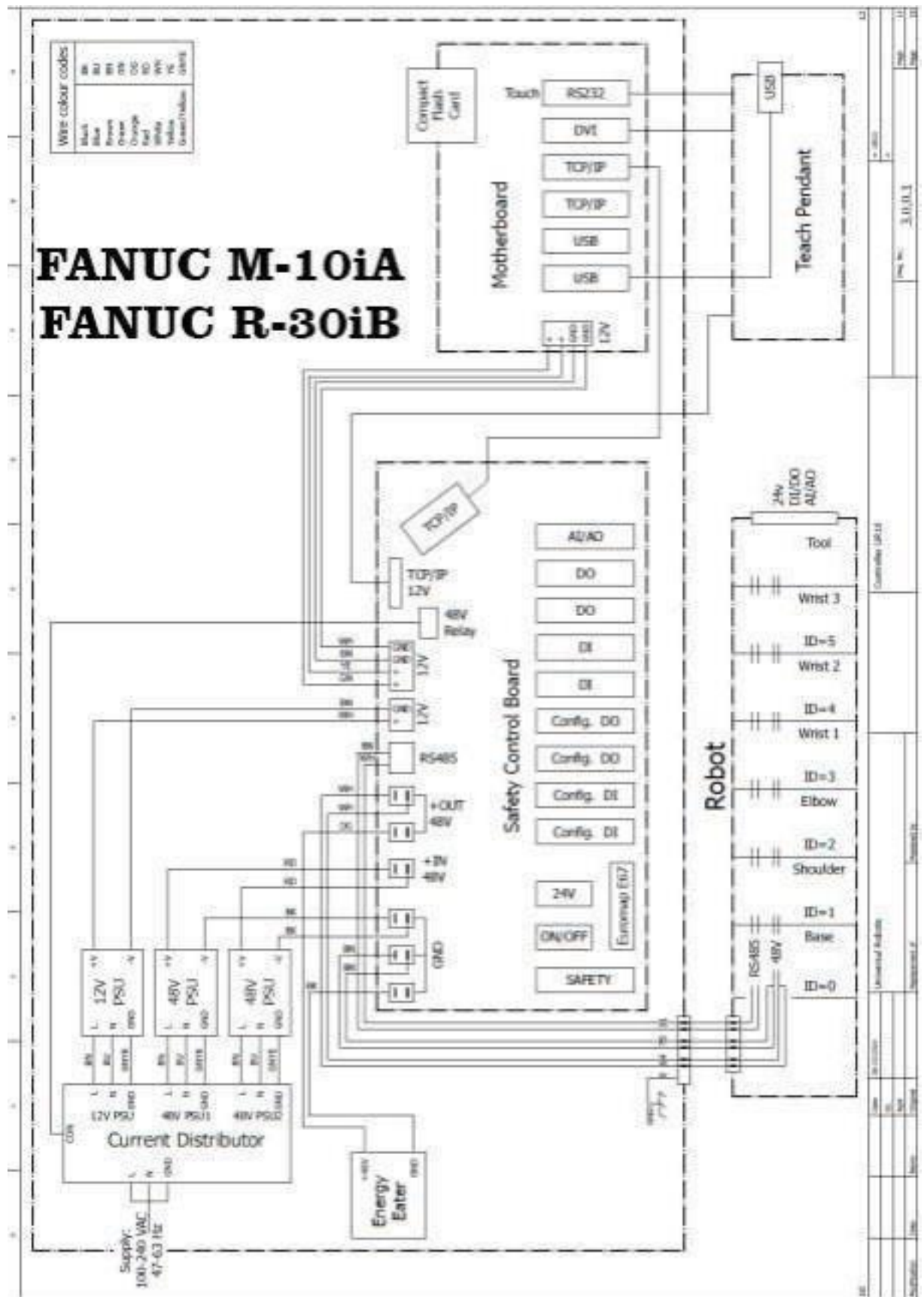


Рис.5.5 – Головна схема підключення до робота FANUC

## ПУЛЬТ НАВЧАННЯ I-PENDANT

Для управління роботом нам потрібно "Пульт навчання i-Pendant" компанії FANUC.

Пульт навчання служить інтерфейсом взаємодії оператора з прикладним інструментом ПЗ. Він з'єднується кабелем із контролером. Пульт навчання не входить до базового комплекту постачання деяких моделей контролера.

Пульт навчання використовується для виконання наступних операцій:

- Поштовхова подача робота
- Створення програм
- Тестування програм
- Зміна налаштувань
- Перевірка стану



Рис.5.6 – Вимикачі на пульті навчання

Основні характеристики нового пульта навчання включають наступне:

1. Сенсорний дисплей: Основною особливістю i-Pendant є наявність сенсорного дисплея, що дозволяє оператору взаємодіяти з роботом шляхом торкання та жестів. Це спрощує процес програмування та керування роботом.
2. Зручність використання: i-Pendant оснащений інтуїтивним меню та навігаційними кнопками, що робить його використання простим і зручним для оператора. Він також може мати налаштовувані гарячі клавіші для швидкого доступу до часто використовуваних функцій.
3. Підтримка мов програмування: i-Pendant підтримує різні мови програмування, включаючи FANUC Robotics' KAREL та FANUC's TP (Teach Pendant) Programming. Це дозволяє операторам програмувати робота на їхній власній мові або використовувати існуючі програми.
4. Комунікаційні можливості: i-Pendant може бути підключений до робота та інших пристроїв через різні інтерфейси, включаючи Ethernet та USB. Це дозволяє обмінюватися даними, завантажувати програми та здійснювати моніторинг робота.

Пульт навчання складається з наступних елементів:

- РК-дисплей роздільною здатністю 640 x 480 пікселів на контролері R-30iB
- РК-дисплей роздільною здатністю 1024 x 768 пікселів на контролері R-30iB Plus
- 2 Світлодіодні індикатори

- 68 клавiш



Рис.5.7 - Пульт навчання і-

PendantНа пульті розташовані такі перемикачі як:

- Вимикач пульта навчання: Цей вимикач включає або вимикає пульт навчання.

Якщо пульт навчання вимкнений, поштовх, складання або тестування програми недоступні.

- Перемикач аварійного блокування: При натисканні на середину цього вимикача вмикається трипозиційний перемикач аварійного блокування. Привключеному пульті навчання переміщення робота можливе лише, якщо оператор стискає перемикач аварійного блокування. Якщо відпустити або сильно натиснути на перемикач, робот негайно зупиниться

- Кнопка аварійного зупинки: При натисканні кнопка аварійного зупинки негайно зупиняє робот незалежно від стану пульта.

IP66 - рівень захисту забезпечує надійну роботу обладнання в умовах вологості та пилу.

## АВАРІЙНА ЗУПИНКА РОБОТА FANUC

У більшості випадків використання робота на виробництві потрібно підключити одну або кілька додаткових кнопок аварійної зупинки. На малюнку нижче показано підключення однієї або декількох кнопок аварійної зупинки.

Найчастіше аварійна зупинка організовується на виробництві, коли робот використовується разом з іншим обладнанням. Це допомагає оператору не думати про використання кнопки аварійної зупинки. Звичайний вхід аварійної зупинки має бути спільним, оскільки обидва пристрої чекатимуть один одного для виходу зі стану аварійної зупинки.

Це пов'язано з тим, що дві системи пристроїв чекають один одного виходу зі стану аварійної зупинки.

Для спільного використання функції аварійної зупинки з іншим обладнанням необхідно створити та контролювати такі функції конфігурованих входів/виходів у графічному інтерфейсі:

1. Пара настроюються входів: Зовнішній аварійний стоп.
2. Пара настроюються виходів: Аварійний стоп системи.

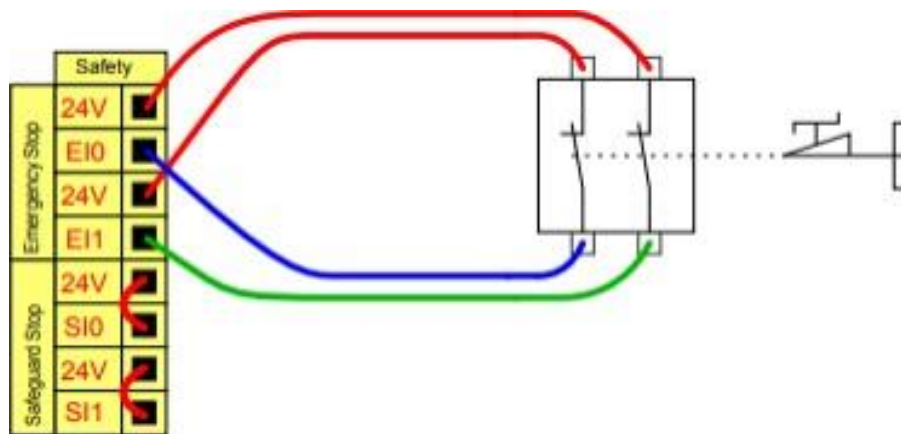


Рис.5.8 - Підключення однієї кнопки аварійної зупинки

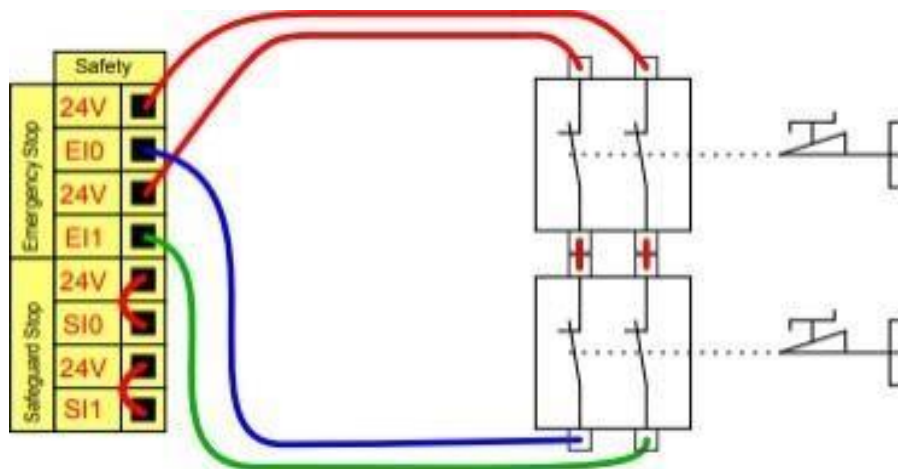


Рис.5.9 - Підключення декількох кнопок аварійної зупинки

5.3. Вибір технічних засобів реалізації керуючих дій на технологічний процес і управління обладнанням

## КАСТОМНИЙ ВАКУУМНИЙ ПНЕВМАТИЧНИЙ ЗАХВАТ

Пневматичний вакуумний захват - це пристрій, який використовує пневматичний тиск та вакуум для захоплення та утримання об'єктів. Є частиною системи, яка використовується під час виробничої діяльності. Це інструмент частини робота, який відповідає за транспортування порожнього мішка. Призначення захоплення, як випливає з назви, - маніпулювати об'єктами.

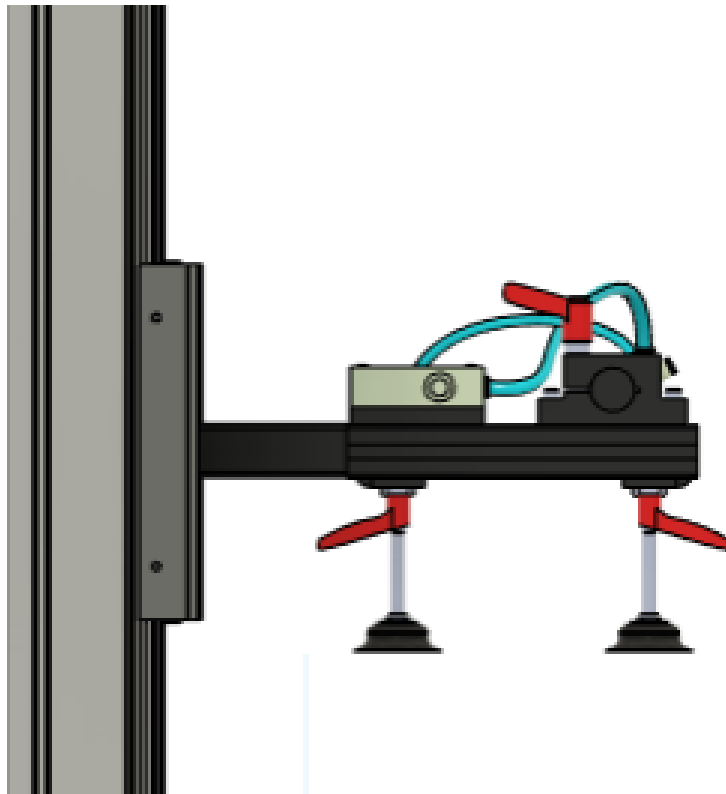


Рис.5.10 – Кастомний пневматичний захват

Складається з наступних основних компонентів:

Вакуумна головка: Це частина захвату, що контактує з об'єктом і створює вакуум для затримання. Вакуумна головка має присоски ( 2 шт), які створюють герметичне з'єднання з поверхнею об'єкта.



Рис.5.11 - Вакуумна присоска FESTO

Вакуумний ежектор: Цей компонент створює негативний тиск, що генерує вакуум для захоплення об'єкта. Вакуумний ежектор працює ежекторним принципами.



Рис.5.12 - Вакуумний ежектор FESTO

**Принцип дії вакуумного ежектора полягає в наступних кроках:**

1. Високотискне стисле повітря подається в ежектор через вхідний потік.
2. В ежекторній дюзі повітря прискорюється до дуже великої швидкості.
3. Після прискорення повітря утворюється зона низького тиску (вакуум) в області навколо ежекторної дюзи.
4. Об'єкт, який потрібно захопити, розміщується біля вакуумної камери, з'єднаної з ежектором.
5. Зона низького тиску (вакуум) створена ежектором відсмоктує повітря з об'єкта через вакуумну камеру. Це створює силу притягування, яка утримує об'єкт на місці. Коли вакуумна камера знаходиться поруч з поверхнею об'єкта, низький тиск у вакуумній камері витягує повітря з простору між об'єктом та поверхнею камери. Це створює притягуючу силу, яка утримує об'єкт на місці, надаючи ефективний захват.

## 5/2 електропневморозподільник Festo

5/2 електропневморозподільник Festo - це пристрій, який використовується для керування пневматичними циліндрами або іншими пневматичними пристроями з використанням електричних сигналів. Цей розподільник має п'ять портів та два станові положення, тому його часто називають 5/2 (п'ять портів, два станові положення).

У 5/2 електропневморозподільнику Festo є п'ять портів, які використовуються для підключення повітряних ліній і пневматичних пристроїв. Два з цих портів - це входиповітряного тиску, три інших порти використовуються для підключення до пневматичного пристрою (наприклад, циліндра) залежно від потреб керування.

Електропневморозподільник Festo забезпечує керування подачею повітря до пневматичного пристрою відповідно до отриманих електричних сигналів. Він має два станові положення, в яких різні порти пов'язані між собою залежно від стану керуючого сигналу. Це дозволяє керувати рухом пневматичного пристрою, наприклад, переміщенням циліндра вперед і назад або відкриттям і закриттям клапанів.

5/2 електропневморозподільники Festo є надійними та ефективними пристроями, які широко використовуються в промисловості для автоматизації процесів, де потрібне керування пневматичними пристроями за допомогою електричних сигналів.



Рис.5.13 - 5/2 електропневморозподільники Festo

**Блок підготовки повітря:** Використовується для забезпечення чистого таналежного тиску повітря для оптимальної роботи вакуумної системи.



Рис.5.14 - Блок підготовки повітря FESTO

**Принцип дії блоку підготовки повітря для пневматичного вакуумного захвату полягає в наступних кроках:**

1. **Фільтрація:** Стисле повітря спочатку проходить через фільтр, який видаляє бруд,пил, олію та інші забруднення. Фільтрація допомагає попередити засмічення та пошкодження вакуумних компонентів.
2. **Регулювання тиску:** Після фільтрації повітря подається до регулятора тиску. Регулятор тиску дозволяє налаштовувати потрібний тиск стислого повітря для вакуумного захвату. Це дозволяє контролювати силу притягування та утримання об'єктів.
3. **Сепарація вологи:** Деякі блоки підготовки повітря мають вбудований сепаратор вологи або зливний клапан. Він видаляє конденсат із стислого повітря, що утворюється під час компресії. Це допомагає запобігти пошкодженню вакуумних компонентів і підтримує стабільний вакуумний тиск.

4. Розподіл повітря: Блок підготовки повітря також може містити клапани або засувки для регулювання та розподілу повітря до різних компонентів вакуумної системи. Це допомагає забезпечити потрібний потік повітря для ефективного функціонування вакуумного захвату.
5. Моніторинг тиску: індикатори або манометри, які встановлені на блоку підготовки повітря, дозволяють операторам візуально спостерігати за поточним тиском у системі.

Моніторинг тиску важливий з кількох причин:

1. Контроль працездатності системи: Моніторинг тиску допомагає переконатися, що система блоку підготовки повітря працює належним чином. Оператор може перевірити, чи досягнутий потрібний тиск, і переконатися, що немає витоків або інших проблем у системі.
2. Виявлення проблем: Зміни в тиску можуть вказувати на проблеми у системі вакуумного захвату або інших компонентах. Наприклад, спад тиску може вказувати на витік повітря або несправність в системі. Швидке виявлення таких проблем дозволяє вчасно їх виправити та уникнути збоїв у роботі системи.
3. Забезпечення стабільного тиску: Моніторинг тиску дозволяє операторам забезпечувати стабільний тиск у системі. Вони можуть вчасно налаштувати регулятор тиску або вживати заходів для підтримки потрібного тиску. Це важливо для забезпечення оптимальної роботи вакуумного захвату і досягнення потрібної сили притягування об'єктів.
4. Моніторинг тиску забезпечує операторам контроль над роботою системи вакуумного захвату та допомагає уникнути проблем і збоїв у роботі.

Клапани і шланги: Клапани та шланги використовуються для керування потоком повітря і вакууму в системі. Вони забезпечують відкриття та закриття шляхів потоку, регулюють напрямлення і контролюють процес захвату та випуску об'єкта



Рис.5.15 – Шланги FESTO

**Загальний принцип дії пневматичного вакуумного захвату:**

1. Створення низького тиску (вакууму): За допомогою пневматичного ежектора , створюється негативний тиск (вакуум) в вакуумній камері .
2. Притягування до об'єкта: Коли вакуум створюється, вакуумна камера надає притягуючу силу до об'єкта, забезпечуючи його утримання.
3. Герметичне ущільнення: Зона низького тиску (вакуум) повинна бути герметично ущільнена навколо об'єкта для створення максимальної сили притягування. Це досягається за допомогою вакуумних присосок .
4. Утримання об'єкта: Після притягування об'єкта пневматичний вакуумний захват підтримує утримання об'єкта завдяки створеному вакууму. Це дозволяє переміщати утримувати об'єкт без необхідності постійної фізичної сили.
5. Відпуск об'єкта: Для відпуску об'єкта вакуум у вакуумній камері припиняється або порушується, що дозволяє зняти притягуючу силу і відпустити об'єкт

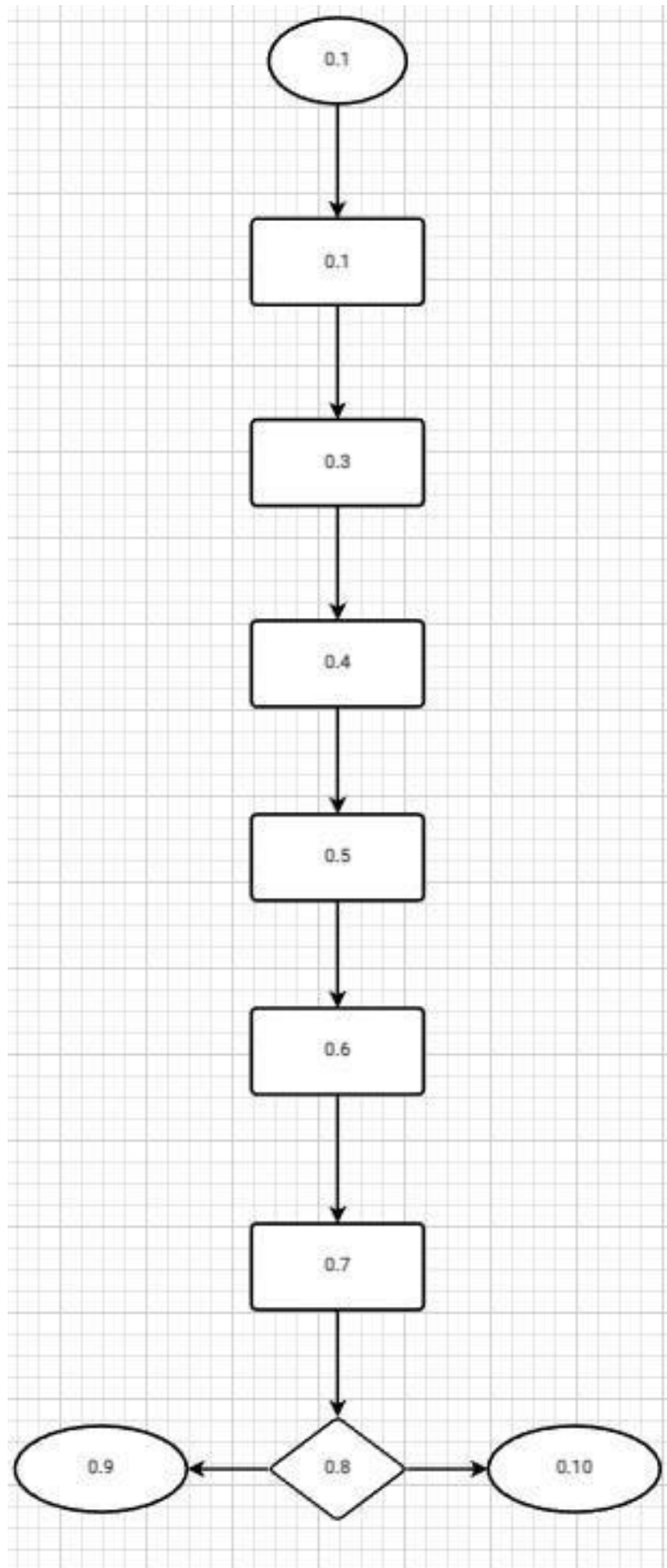


Рис.5.16 – Блок-схема послідовності дій при роботі пневматичного вакуумного захвату

Ця **блок-схема** ілюструє послідовність дій при роботі пневматичного вакуумного захвату. За допомогою пневматичного ежектора або вакуумного насоса створюється вакуум. Потім вакуумна камера притягує об'єкт, після чого вона герметично ущільнюється для забезпечення максимальної сили притягування. Об'єкт утримується завдяки створеному вакууму, а після необхідного періоду він може бути відпущений шляхом припинення або порушення вакууму.

Позначення блок-схеми:

- 0.1 Старт системи: Система пневматичного вакуумного захвату запускається, наприклад, шляхом включення живлення
- 0.2 Створення вакууму: Вакуумний ежектор створює негативний тиск або вакуум в системі.
- 0.3 Захоплення об'єкта: Вакуумна головка або присоска наближається до об'єкта та робить контакт з ним. За допомогою вакуумного тиску об'єкт притягується і утримується.
- 0.4 Моніторинг стану: Датчики виявляють присутність об'єкта та стан захвату. Це дозволяє контролювати, чи успішно здійснено захоплення об'єкта.
- 0.5 Транспортування об'єкта: Захоплений об'єкт може бути переміщений за допомогою пневматичних пристроїв, які забезпечують рух інструмента або робочого пристрою.
- 0.6 Відпуск об'єкта: Коли потрібно відпустити об'єкт, повітря може бути випущено, знижуючи вакуумний тиск та відпускаючи об'єкт.
- 0.7 Вимкнення вакууму: Вакуумний ежектор, що створює негативний тиск або вакуум в системі вимикається
- 0.8 Позначення “або”, що демонструє дві зупинки системи.
- 0.9 Зупинка системи: Після завершення роботи захват приходиться у стан старту системи

0.1 Зупинка системи: При виникненні аварійної ситуації систему пневматичного вакуумного захвату можна вимкнути, щоб зупинити всі функції.

IP66 - рівень захисту забезпечує надійну роботу обладнання в умовах вологості та пилу.

### **ЕЛЕКТРОПНЕВМАТИЧНИЙ ЗАХВАТ З ЦИЛІНДРАМИ**

Електропневматичний захват з циліндрами використовує комбінацію електричних та пневматичних компонентів для притягування та утримання об'єктів.

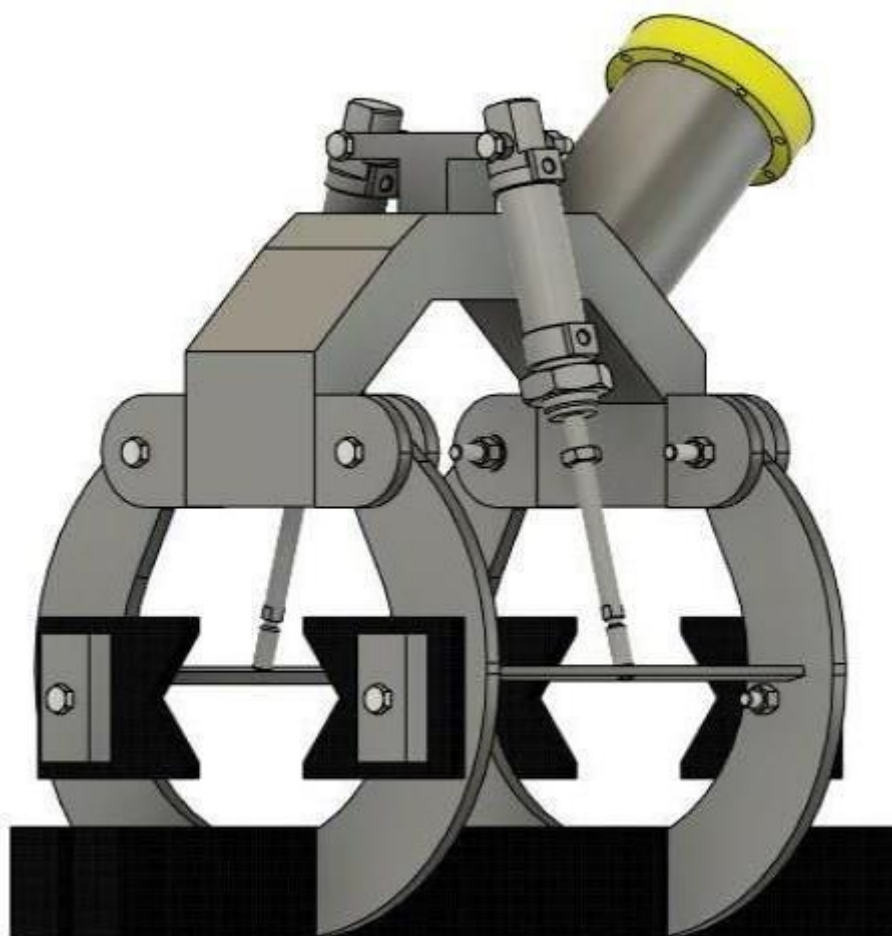


Рис.5.17. – Електропневматичний захват з циліндрами

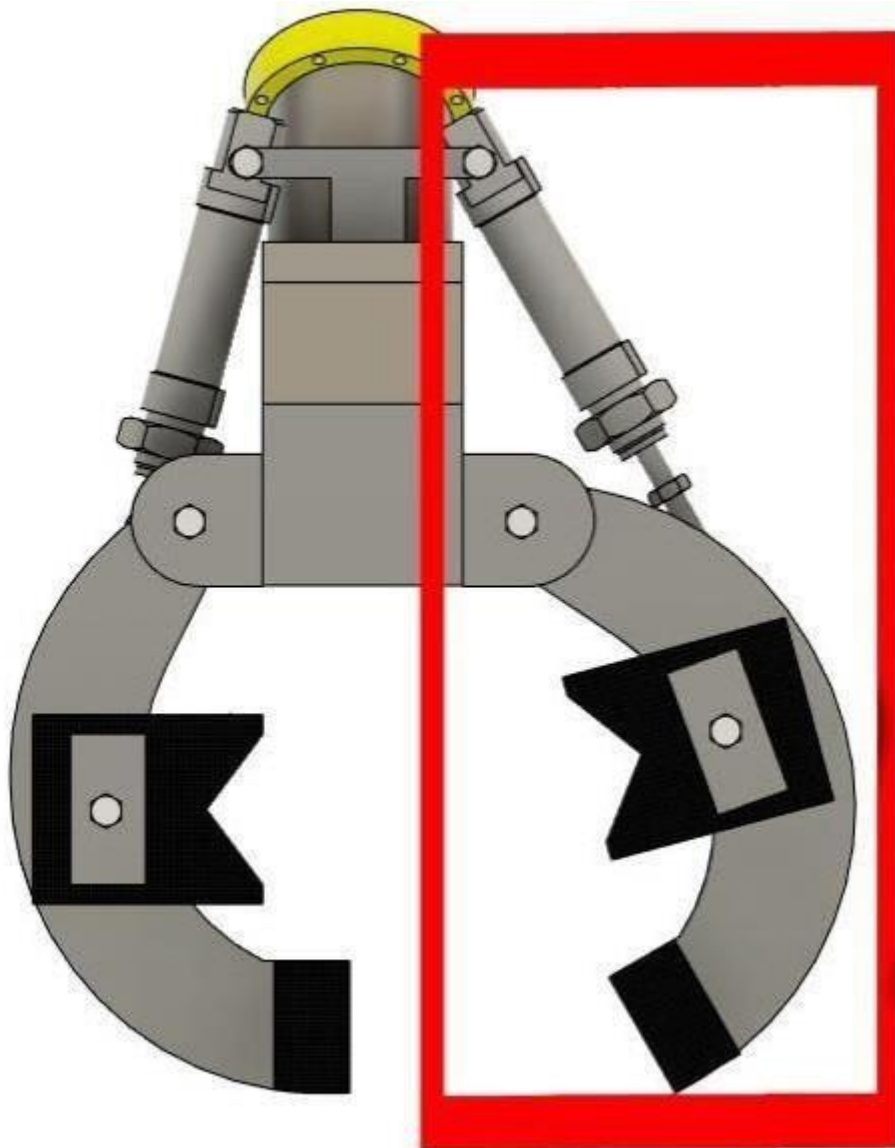


Рис.5.18 – Порівняння відкритої сторони захвату та закритої Основними

компонентами такого захвату є:

Електромагнітні клапани: Електромагнітні клапани відповідають за керування потоком стиснутого повітря до циліндрів. Вони можуть відкривати або закривати шлях для стиснутого повітря залежно від сигналів, що надходять з керуючої системи.

Конструкція електромагнітного (соленоїдного) клапана складається з наступних основних деталей: корпусу, кришки, мембрани (поршня), пружини, плунжера, штока і електричної котушки (соленоїда).

Корпуси і кришки клапанів відливають з латуні, нержавіючої сталі, чавуну або полімерів: поліпропілену, еколону, нейлону і ін. Клапани розраховані для використання при різних робочих середовищах, тисках і температурах. Для плунжерів і штоків застосовують спеціальні магнітні матеріали. Електрокотушки (соленоїди) для клапанів виготовляють в пило-захищеному або герметичному корпусі. Обмотка котушок виконана високоякісним емаль дротом з електротехнічної міді. Приєднання до трубопроводу різьбове або фланцеве. Для підключення до електричної мережі використовується штекер. Управління здійснюється подачею напруги (або імпульсу) на котушку.



Рис.5.19 - Конструкція електромагнітного клапана

Клапан має два положення : нормально закритий , або нормально відкритий

Закритий: В статичному положенні напруга на котушці відсутня - електроклапан закритий. Запірний орган (мембрана або поршень, в залежності від типу клапана) герметично притиснутий, силою дії пружини і тиску робочого середовища до сідлаущільнювальної поверхні. Пілотний канал закритий підпружиненим плунжером.

Тиск у верхній порожнині клапана (над мембраною) підтримується через перепускний отвір в мембрані (або через канал в поршні) і дорівнює тиску на вході в клапан. Клапан електромагнітний знаходиться в закритому положенні, поки котушка не опиниться під напругою.

Для відкриття клапана напруга подається на котушку. Плунжер, під впливом магнітного поля піднімається і відкриває пілотний канал. Так як діаметр пілотного каналу більше перепускного, тиск у верхній порожнині клапана (над мембраною) знижується. Під дією різниці тисків, мембрана або поршень піднімається вгору і клапан відкривається. Клапан залишиться у відкритому положенні, поки котушка знаходиться під напругою.

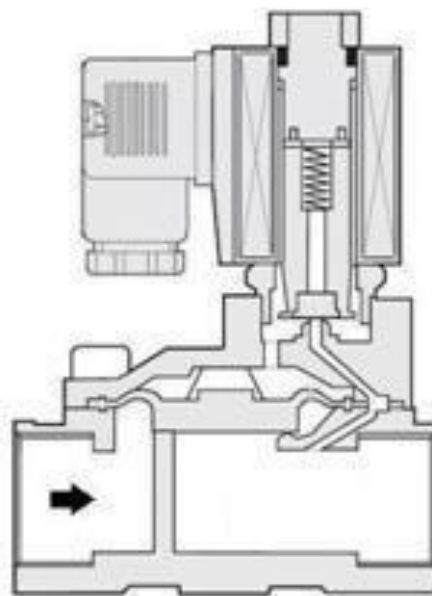


Рис.5.20 Клапан закритий

Відкритий: Принцип дії нормально відкритого клапана навпаки - в статичному положенні клапан знаходиться у відкритому положенні, а при подачі

напруги на котушку клапан закривається. Для утримання нормально відкритого клапана в закритому стані, напругу необхідно подавати на котушку довготривало.

Для правильної роботи будь-яких клапанів пілотного дії необхідний мінімальний перепад тиску,  $\Delta P$  - різниця тисків на вході і на виході клапана. Пілотні клапани називають клапанами непрямої дії, тому що крім подачі напруги, необхідним є дотримання умови по перепаду тиску. Підходить в більшості випадків, для експлуатації в системах водопостачання, опалення, системах ГВС, системах пневмокерування і ін. - всюди, де присутній тиск в трубопроводі. Циліндри:

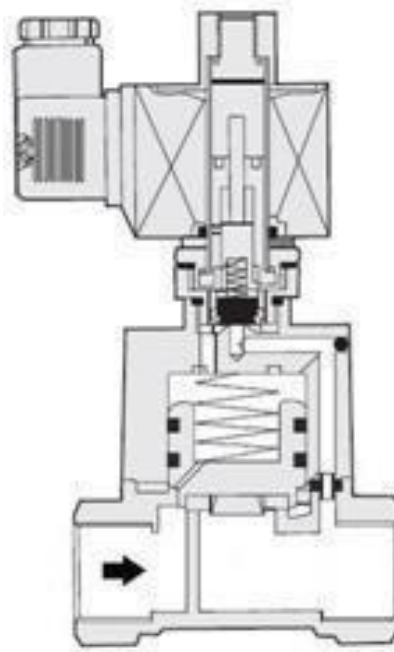


Рис.5.21 - Клапан відкритий

Циліндри використовуються для створення механічної сили притягування та утримання об'єктів. Вони мають поршневий механізм, який рухається під дією стиснутого повітря. Циліндри можуть бути односторонніми (з одним напрямом руху) або двосторонніми (з двома напрямками руху).

Датчики: Датчики можуть бути встановлені для контролю положення циліндрів або об'єктів. Вони виявляють наявність або відсутність об'єктів у захваті та передають сигнали в систему керування.

закритому стані, напругу необхідно подавати на котушку довготривало.

Керуюча система: Керуюча система, така як контролер, відповідає за керування роботою електропневматичного захвату. Вона приймає сигнали від датчиків і відповідно керує електромагнітними клапанами для керування циліндрами та створення потрібних дій.

Принцип дії електропневматичного захвату з циліндрами полягає в таких кроках:

1. Керуюча система отримує сигнал для притягування об'єкта до електропневматичного захвату.
2. Електромагнітні клапани відкриваються, дозволяючи стиснутому повітрю поступати до циліндрів.
3. Стиснене повітря направляється в циліндр, що призводить до руху поршня в циліндрі.
4. Рух поршня зумовлює витягування захватного механізму або присоски до об'єкта.
5. Коли захватний механізм чи присоска достатньо прижимаються до об'єкта, електромагнітні клапани закриваються, зупиняючи подачу стиснутого повітря до циліндрів.
6. Циліндр заблоковується у позиції, що забезпечує утримання об'єкта.
7. Система може відслідковувати положення циліндрів за допомогою датчиків, що дозволяє контролювати стан захвату.

Цей процес дозволяє електропневматичному захвату притягнути об'єкт та утримувати його до вимкнення подачі стиснутого повітря до циліндрів. Для відпуску об'єкта система може відкрити електромагнітні клапани, що дозволить випустити стиснене повітря з циліндрів та відпустити захвачений об'єкт.

IP66 - рівень захисту забезпечує надійну роботу обладнання в умовах вологості та пилу.

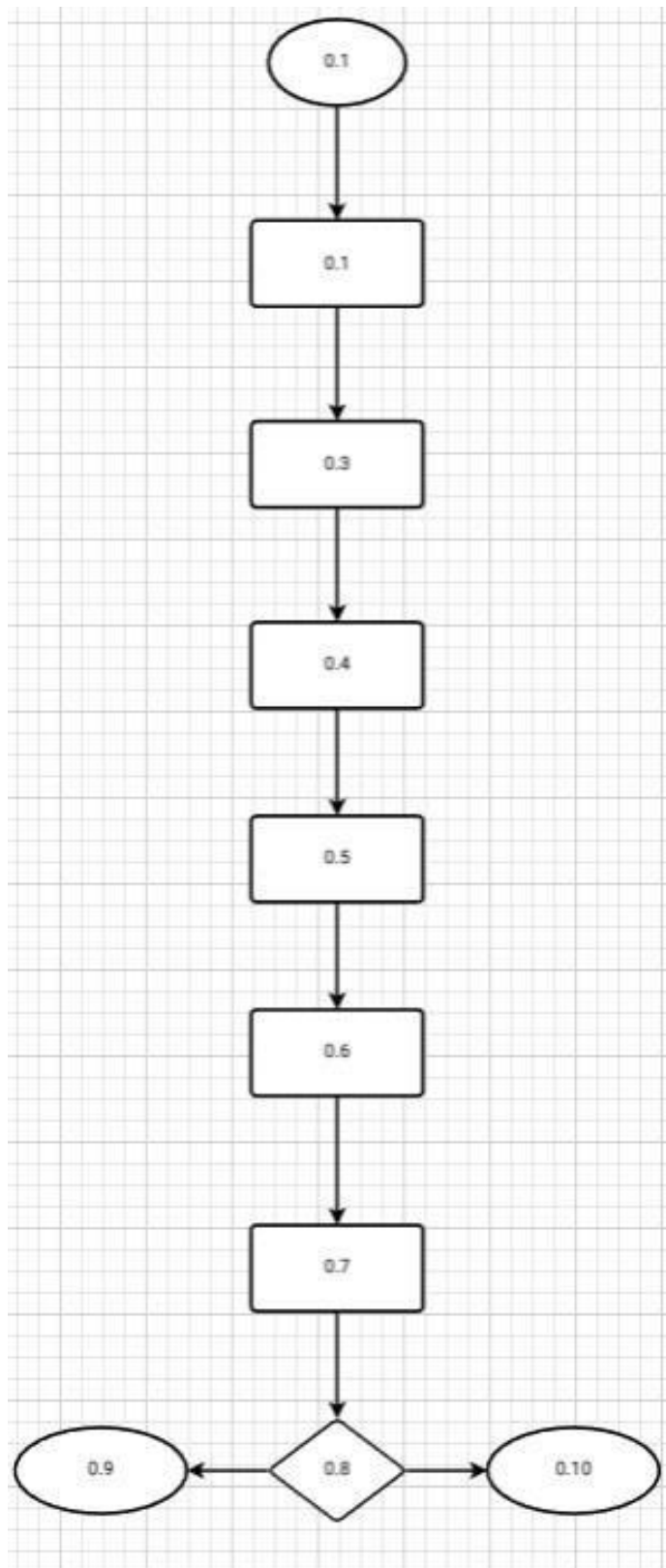


Рис.5.22 – Блок-схема послідовності дій при роботі електропневматичного захватуциліндрами:

Ось блок-схема послідовності дій при роботі електропневматичного захвату з циліндрами

Позначення блок-схеми:

- 0.1 Старт системи: Система електропневматичного захвату запускається, наприклад, шляхом включення живлення.
- 0.2 Застосування електричного сигналу: Електричний сигнал надходить до контролера або управляючого блоку, що викликає рух циліндрів.
- 0.3 Пневматична подача: Після отримання електричного сигналу пневматична подача повітря включається і надсилається до циліндрів.
- 0.4 Рух циліндрів: Пневматичний тиск, який подається до циліндрів, спричиняє їх рух. Залежно від напрямку руху, циліндр може витягуватися (захоплююча дія) або розтягуватися (відпускна дія).
- 0.5 Захоплення об'єкта: Після витягування циліндрів захватна частина електропневматичного захвату здійснює контакт з об'єктом і притягує його.
- 0.6 Моніторинг стану: Датчики виявляють стан захвату та інші параметри, які дозволяють контролювати процес захоплення об'єкта.
- 0.7 Відпуск об'єкта: Коли потрібно відпустити об'єкт, електричний сигнал вимикає пневматичну подачу, що призводить до відпускання об'єкта.
- 0.8 Позначення “або”, що демонструє дві зупинки системи.
  - 0.9 Зупинка системи: Після завершення роботи захват приходиться у стан старту системи
  - 0.10 Зупинка системи: При виникненні аварійної ситуації систему пневматичного вакуумного захвату можна вимкнути, щоб зупинити всі функції.

## КОНВЕЄР

Конвеєр для мешків цементу - це спеціальний тип конвеєра, призначений для транспортування мешків з цементом на виробництві. Основна мета такого конвеєра - автоматизація процесу переміщення мешків цементу з одного пункту до іншого, зниження витрат ручної праці та підвищення продуктивності виробництва.

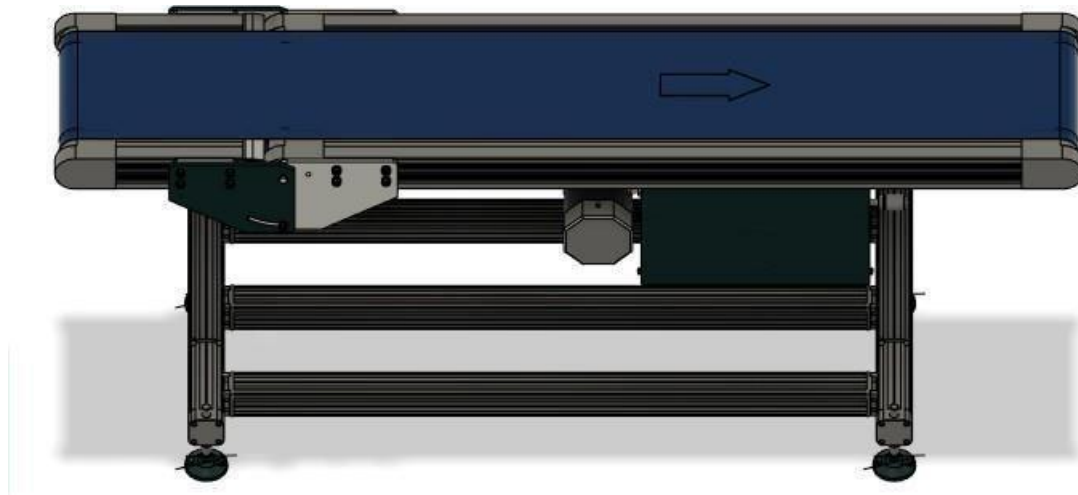


Рис.5.23- Конвеєр

Управління стрічкою конвеєра за допомогою аналогового виходу, що керує швидкістю, може бути виконане за допомогою таких кроків:

**Підключення аналогового виходу:** Підключіть аналоговий вихід контролера до приводу, який керує швидкістю конвеєра. Це може бути вбудований аналоговий вихід контролера або додатковий модуль аналогових виходів.

**Налаштування діапазону виходу:** Встановлення діапазон вихідного сигналу, який відповідає швидкості конвеєра. Наприклад, якщо швидкість вимірюється в діапазоні 0-100%, тоді налаштування аналогового виходу таким чином, щоб значення 0% відповідало зупинці конвеєра, а значення 100% - максимальній швидкості.

Калібрування: Калібрування аналогового виходу, якщо це необхідно, щоб забезпечити точність вихідного сигналу відповідно до вимог швидкості конвеєра.

Налаштування контролера: В програмі контролера налаштування алгоритму керування швидкістю на основі аналогового виходу. Це може включати PID-регулятор, алгоритми замкненого керування або інші методи контролю швидкості.

Моніторинг та зворотний зв'язок: Забезпечте моніторинг швидкості конвеєра за допомогою зворотного зв'язку. Використовуйте датчики швидкості або енкодери для отримання реальних значень швидкості та забезпечення зворотного зв'язку з контролером. Це дозволить контролеру реагувати на зміни швидкості та вносити коригуючі сигнали на аналог

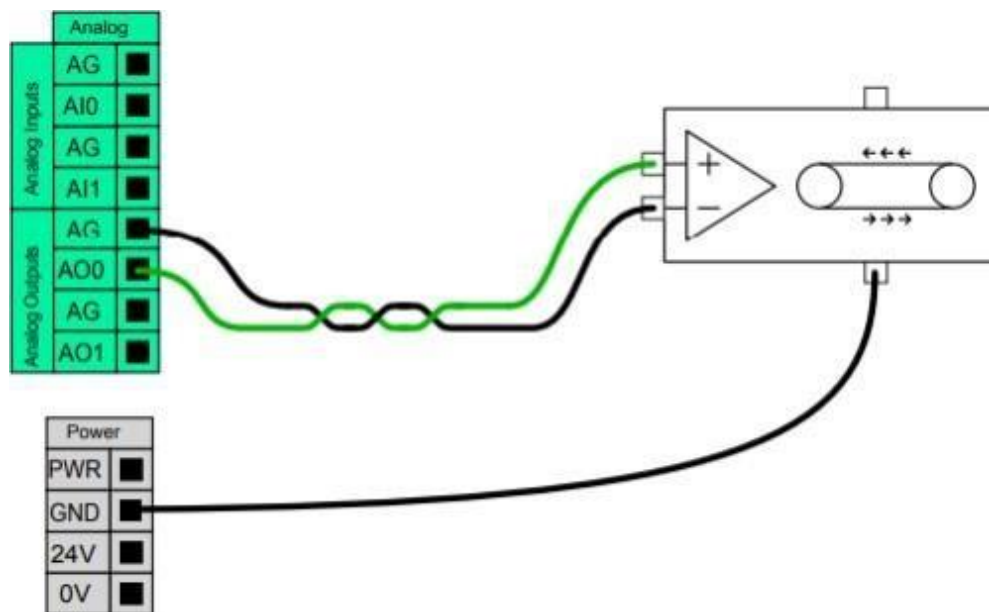


Рис.5.24 – Приклад управління стрічкою конвеєра, з використанням аналогового виходу, який керує швидкістю.

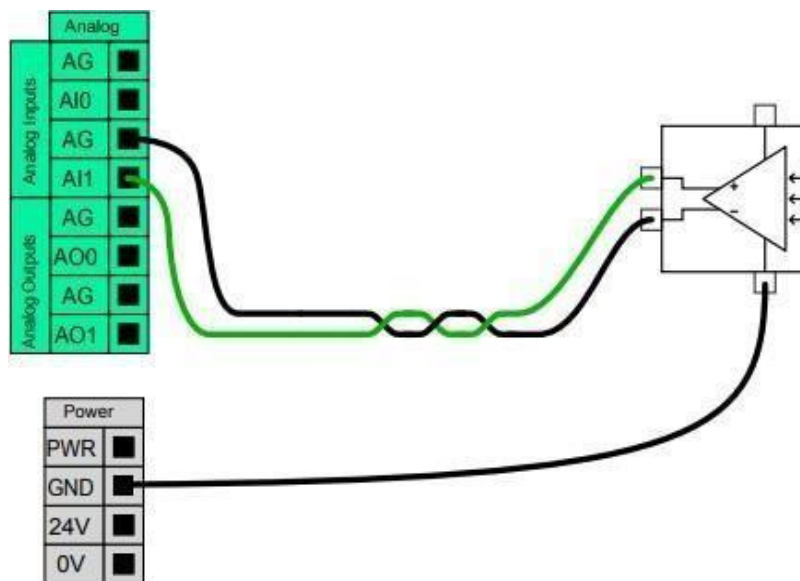


Рис.5.25 - Приклад підключення аналогового датчика.

## ПНЕВМОЦИЛІНДР FESTO ДЛЯ ВІДКРИТТЯ ЗАГОГОТІВКИ

Пневмоциліндри FESTO мають широкий спектр застосувань у промислових галузях. Основні області їх використання включають:

**Автоматизація виробничих ліній:** Пневмоциліндри використовуються для переміщення, підйому та спрямування різних виробів і компонентів на виробничих лініях. Вони можуть бути використані для відкриття та закриття клапанів, випуску продукції і контролю руху матеріалів.

**Робототехніка:** Пневмоциліндри використовуються в робототехніці для забезпечення руху та виконання дій роботів. Вони можуть контролювати рух сегментів робота, що дозволяє забезпечити точне позиціонування та виконання завдань з великою швидкістю.

**Упаковування та фасування:** Пневмоциліндри використовуються в системах упакування та фасування для контрольованого заповнення, запечатування та маркування упаковочних матеріалів. Вони забезпечують швидке і ефективне

виконання процесів фасування та забезпечують якість та безпеку упакованих продуктів.

Пневматичні системи транспортування: Пневмоциліндри використовуються у системах транспортування для переміщення матеріалів і виробів на довгі відстані. Вони можуть контролювати відкриття та закриття вентилів у трубопроводах, що дозволяє регулювати потік повітря і переміщувати матеріали.

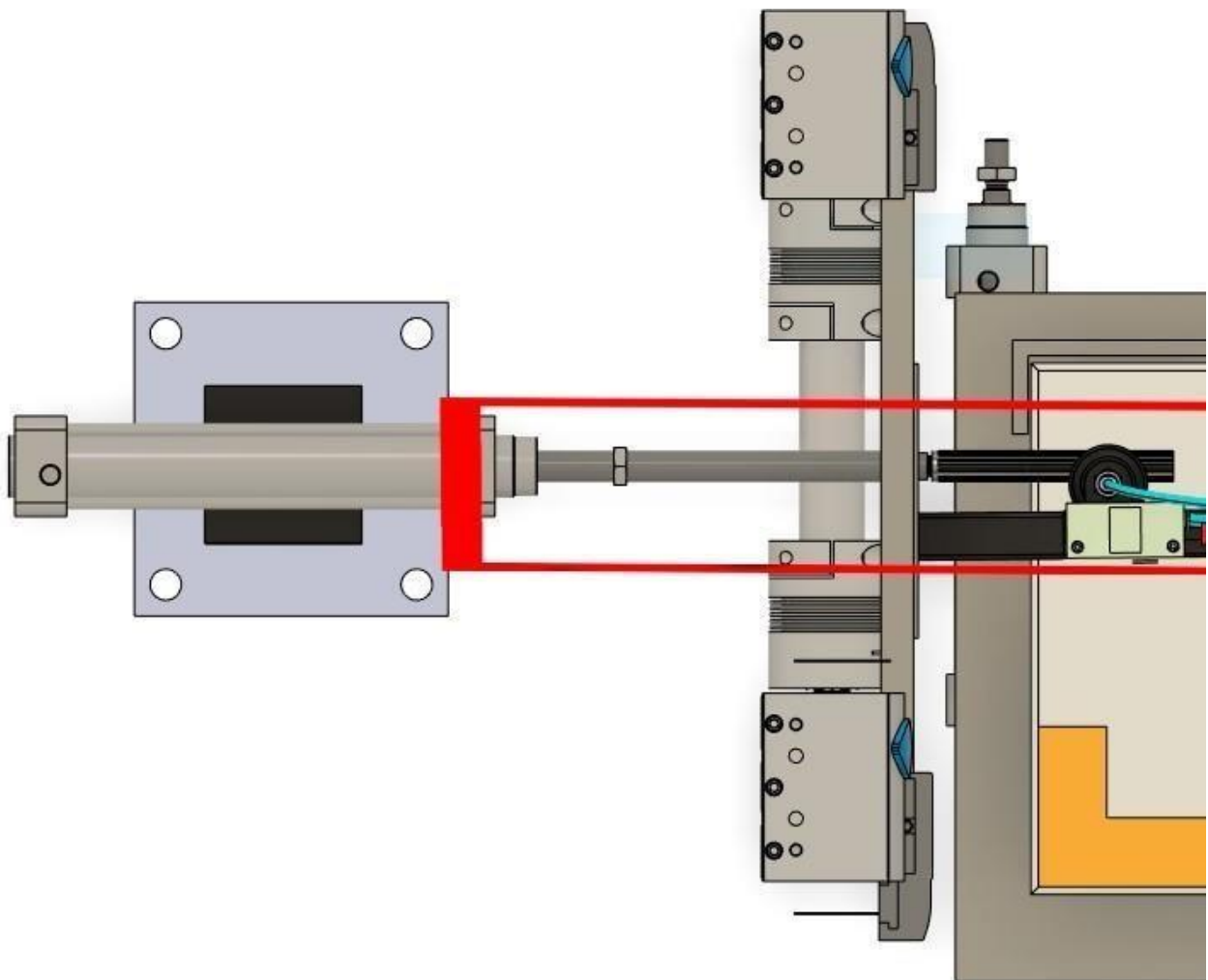


Рис.5.26 - Пневмоциліндр FESTO для відкриття заготовки.

Пневмоциліндр FESTO є пристроєм, що працює на основі пневматичного принципу, використовуючи стиснений повітря для виконання механічних рухів. FESTO є відомим виробником пневматичного обладнання і пропонує широкий спектр пневмоциліндрів для різних промислових застосувань.

Основні характеристики пневмоциліндра FESTO включають:

**Конструкція:** Пневмоциліндр FESTO має циліндричну форму і складається з циліндричного корпусу, поршня, тяги, ущільнюючих елементів та з'єднувальних деталей.

**Робочий принцип:** Пневмоциліндр FESTO працює за принципом подачі стисненого повітря в циліндр, що створює тиск і змушує поршень рухатися вперед або назад. Цей рух передається на тягу, яка може виконувати різні види механічних дій.

**Рухові характеристики:** Пневмоциліндр FESTO може мати різні рухові характеристики, включаючи прямолінійний рух, круговий рух або комбінацію обох. Це залежить від конкретної конструкції та типу пневмоциліндра.

**Розміри та діапазони:** Пневмоциліндр FESTO доступний в різних розмірах та діапазонах ходу, що дозволяє вибрати підходящу модель для конкретного завдання.

**Управління:** Пневмоциліндр FESTO може бути управляється за допомогою пневматичних клапанів або електромагнітних клапанів, що забезпечує точне і контрольоване переміщення поршня.



Рис.5.27 - Пневмоциліндр FESTO для відкриття заготовітки.



Рис.5.28 - Пневмоциліндр FESTO для відкриття заготовівки.



Рис.5.29 - Накінчик на циліндр



Рис.5.30 - Накінчик на циліндр



рахунок застосування відцентрової сили, що надається робочим колесом. Робоче колесо також виконує функції подрібнювача у випадках попадання цементу, що злежався, у фасувальну машину.

Алгоритм роботи фасовщика цементу в мішки:

- оператор на пульті керуванні (контролері) задає необхідну фасувальну дозу (наприклад 50 кг), "хвости" - інерційність матеріалу враховуються;
- клапанний мішок вручну надівається на патрубок;
- оператор натискає кнопку "Start";
- відбувається пневматичне фіксування мішка, враховується вага мішка, відкривається заслінка, що подає, включається роторний металник і фасований продукт (цемент) надходить через патрубок в мішок;
- мішок починає наповнюватися - змінюється його вага, після досягнення запрограмованої межі відбувається зниження швидкості обертання ротора з наступною зупинкою, закривається заслінка. Вага прописується в пам'ять контролера. Мішок зафасований;
- пневмо-притискання звільняє мішок і поворотне сидло скидає його. Машина готова до наступної фасування;
- цикл фасування починається наново.

Р66 - рівень захисту забезпечує надійну роботу обладнання в умовах вологості тапилу.

## **ВИСНОВОК ЗА РОЗДІЛОМ 5**

В цьому розділі було розглянуто деякі технічні засоби, які будуть використані для реалізації РТК. Було розглянуто принципи дії кожного елемента, пояснення навіщо саме ці технічні засоби потрібні для реалізації РТК.

## РОЗДІЛ 6 ПАРАМЕТРИЗАЦІЯ ТЕХНІЧНИХ ЗАСОБІВ, КОНТРОЛЕРА І МЕРЕЖ, ПРОГРАМУВАННЯ АЛГОРИТМІВ ЛОГІЧНОГО КЕРУВАННЯ ТА РЕГУЛЮВАННЯ САК

6.1 Ознайомлення з мовами програмування FANUC. Демонстрація декількох способів програмування роботів.

Робототехнічний комплекс FANUC складається з прикладного програмного забезпечення інструменту, механічного блоку самого робота (робот серії FANUC) та контролера робота.

Робот FANUC відрізняється видатними робочими характеристиками при виконанні маніпуляцій.

### **ПРИКЛАДНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ІНСТРУМЕНТУ**

Прикладне програмне забезпечення (далі - ПЗ) інструменту - це пакет програм для всіх видів маніпуляцій робота, встановлений у контролері робота. Будь-яку роботу можна виконати, вибравши функцію меню або ввівши команду на пульті навчання. Прикладне програмне забезпечення інструменту містить команди, що управляють роботом, схватами, контролерами дистанційного керування та іншими периферійними пристроями.

Є можливість керування введенням-виводом між додатковою віссю або контролером та іншим периферійним пристроєм. До інших периферійних пристроїв відносяться контролери клітинки або датчики.

### **РОБОТ**

Робот має інтерфейс захвата або іншого робочого органу, що дозволяє системі керування виконати роботу. Робот FANUC чудово підходить для виконання маніпуляцій операцій.

## КОНТРОЛЛЕР 30iB

Прикладне програмне забезпечення інструменту встановлюється на контролер робота для управління пультом навчання, панеллю управління та зовнішніми периферійними пристроями.

Периферійні пристрої, включаючи контролери дистанційного керування, потрібні для налаштування маніпулювання заготовками РТК.

- Контролер дистанційного керування використовується для зовнішнього керування контролером робота.
- Керування схватами, датчиками та іншими пристроями здійснюється через пристрої вводу-виводу та пристрої послідовного зв'язку.

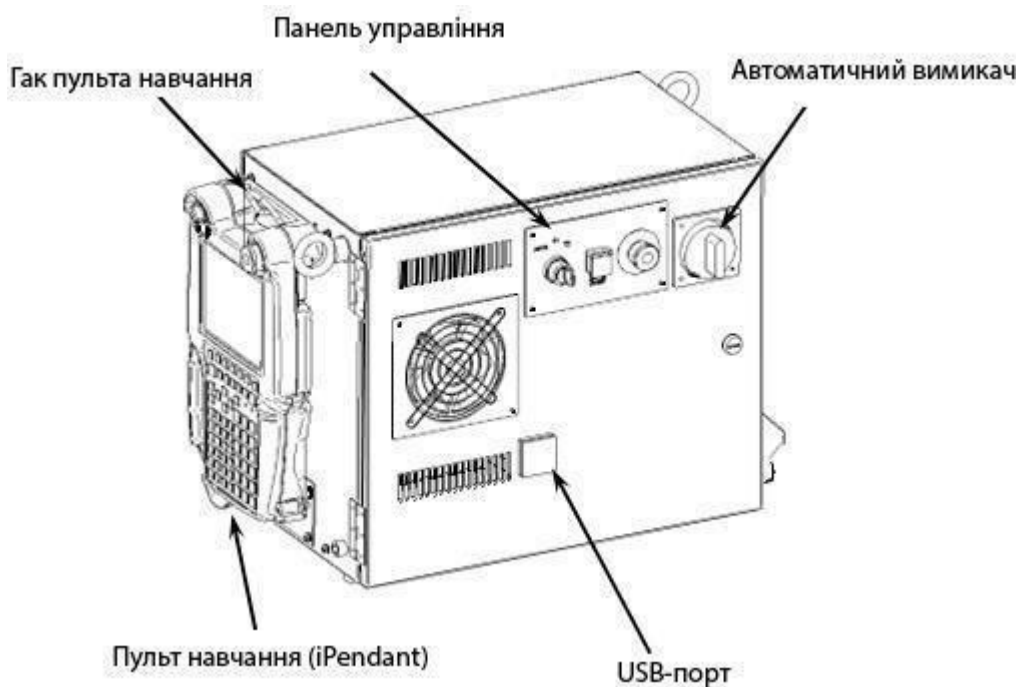


Рис.6.1 – Контролер робота R-30iB

## ПРОГРАМА

Програма містить інструкції з переміщення, введення-виведення, реєстрації та переходу. (Структура програми описана в розділі 4). Кожній команді призначається номер оператора.

Пульт навчання контролера використовується для створення або редагування програми (програмування описано у розділі 5). Програма містить такі інструкції. Нарис. зображено базову програму.

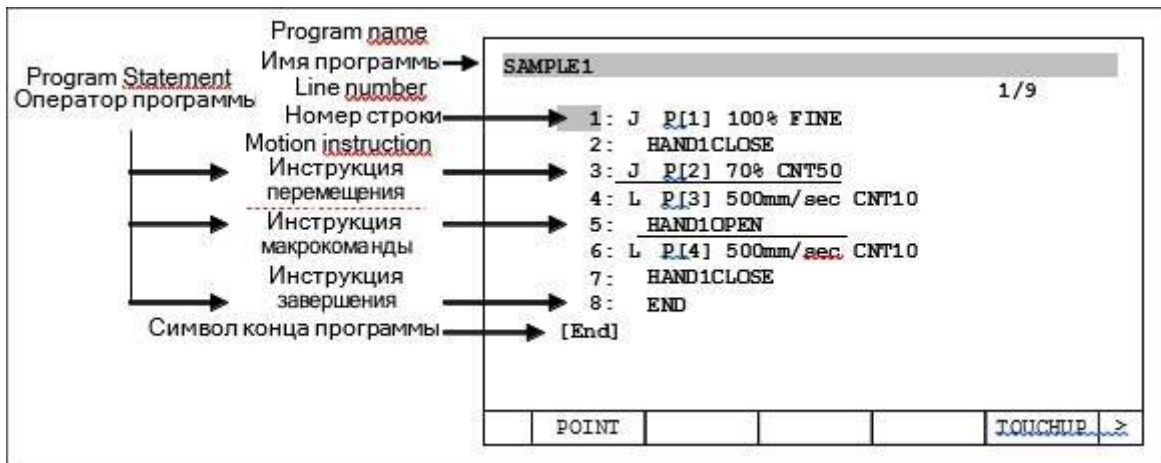


Рис.6.2 - Базова програма

- Інструкція про переміщення: Переміщує робота в задану позицію в межах робочогодіапазону.
- Додаткова інструкція переміщення: Виконує додаткову (спеціальну)операціюпід час переміщення.
- Інструкція реєстрації: Поміщає (завантажує) числові дані до регістру.
- Інструкція реєстрації позиції: Поміщає (завантажує) даніпозиціонування у регістр.
- Інструкція введення-виводу: Надсилає сигнал на периферійний пристрій або отримує сигнал від нього
- Інструкція переходу: Змінює хід програми.
- Інструкція очікування: Зупиняє виконання програми до виконання заданих умов.
- Інструкція виклику підпрограми: Викликає та виконує підпрограму.
- Інструкція макрокоманди: Викликає та виконує задану програму.
- Інструкція штабелювання: Вкладає заготовки на піддони
- Інструкція завершення програми: Завершує виконання програми.
- Інструкція коментаря: Додає коментар до програми.

## 6.2. Розробка програм для реалізації алгоритмів логічного керування

### **Структура програми та команди програми має:**

- Інформація про програму
- Номер рядка, символ кінця програми та аргумент
- Команди переміщення
- Команди палетування
- Команди регістру
- Команди введення-виведення
- Команди переходу
- Команди очікування
- Команда умови пропуску
- Команда вантажопідйомності
- Команда умови корекції
- Команди умови корекції на інструмент
- Команди системи координат
- Команди управління програмою
- Інші команди
- Багатокоординатні команди управління
- Команда групи операцій
- Команда FOR/ENDFOR
- Команда зі змішаною логікою
- Команда діагностики

### **ПРИКЛАДИ КОМАНД ДЛЯ РЕАЛІЗАЦІЇ ПРОГРАМУВАННЯ РТК**

Розглянемо приклади команд для реалізації програмування РТК. У робота дуже багатий функціонал, але я підбрав основні, що потрібні для виконання РТК:

- Команди переміщення

- Команди очікування
- Команда очікування з обмеженням часу WAIT
- Команда траєкторії
- Команди умовного очікування
- Команда умовного очікування на введення-виведення
- Команда аварійного припинення ABORT
- Команда аварійного сигналу користувача UALM[i]
- Команда паузи PAUSE
- Команда повідомлення MESSAGE
- Команда виконання програми
- Команда циклу ENDFOR

## **КОМАНДА ПЕРЕМІЩЕННЯ**

Команда переміщення викликає переміщення інструмента робота в задану точку в межах операційного простору із заданою швидкістю подачі та в заданому режимі переміщення. У команді переміщення мають бути задані наведені нижче елементи.

Формат команди переміщення показано на рис.

- Формат переміщення: Задає керування траєкторією переміщення у задане положення.
- Дані позиціонування: Визначає положення в режимі навчання, в яке має переміститися робот.
- Швидкість подачі: Вказує швидкість подачі робота.
- Траєкторія позиціонування: Вказує на необхідність переміщення робота в задану точку.
- Додаткова інструкція переміщення: Наказує виконання додаткової команди під час переміщення робота

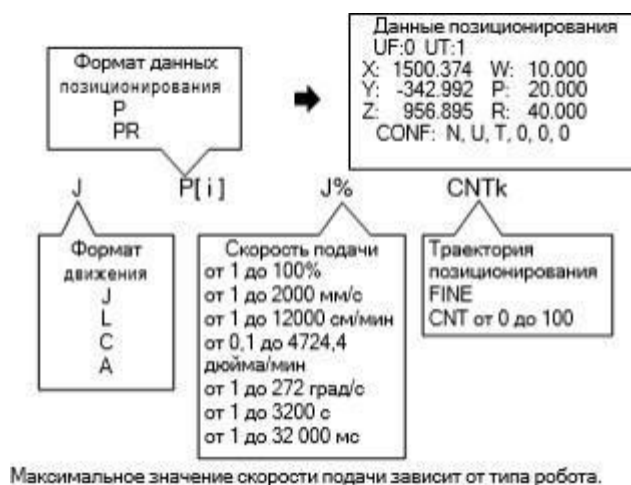


Рис.6.3 - Формат команды перемещения

Під час введення в режимі навчання команди переміщення слід вибрати стандартну команду переміщення за допомогою функціональної клавіші F1 або F5.



Рис.6.4 - Введення в режимі навчання

- Натискаємо F1 POINT, щоб запрограмувати команду виконання операції.
- Натискаємо F5 TOUCHUP, щоб змінити раніше запрограмовані дані позиціонування

### Формат руху

Визначає траєкторію переміщення у задане положення. Можливі чотири варіанти: рух шарніра, при якому не виконується керування траєкторією/орієнтацією, а також поступальний рух, обертальний рух, та поступально-обертальний рух, при якому виконується керування траєкторією/орієнтацією.

- Рух шарніра (J)
- Поступальне переміщення (включаючи поворот) (L)
- Обертальний рух (C)

Для нашого РТК, нам потрібно використовувати : Поступальне переміщення L тарух J

У режимі поступального переміщення контролюється траєкторія руху центральної точки інструменту (ЦТІ) з початкової точки в кінцеву точку; центральна точка інструменту рухається поступально. Формат руху вказується у режимі навчання визначення кінцевої точки. Швидкість подачі при поступальному переміщенні визначається в мм/с, см/хв, дюйм/хв, с і мс.

Орієнтація інструмента, що переміщується, контролюється шляхом віднімання орієнтації в початковій точці з орієнтації в кінцевій точці.



Рис.6.5 - Введення в режимі навчання

Поворот – це метод переміщення, у якому інструмент повертається навколо центральної точки інструмента з вихідного становища кінцеве з допомогою поступального режиму роботи. Орієнтація інструменту під час переміщення контролюється шляхом поділу орієнтації у вихідному положенні .

кінцевому положенні. Швидкість подачі визначається град/с. Фокус контролюється поступально (при переміщенні кінцевої точки інструменту).

## Рух J

Режим руху шарніра – основний режим переміщення робота у задане положення. Робот прискорюється вздовж або навколо всіх ступенів рухливості, переміщається із заданою швидкістю подачі, сповільнюється та зупиняється одночасно по всіх ступенях рухливості. Траєкторія переміщення зазвичай нелінійна. Формат руху вказується у режимі навчання визначення кінцевої точки.

Швидкість подачі руху шарніра вказується у вигляді відсотка максимальної швидкості подачі. Швидкість подачі руху шарніра також можна встановити в секундах або мілісекундах. Орієнтація інструменту, що переміщується, не контролюється.

## Швидкість подачі:

Швидкість подачі визначає швидкість, з якою переміщується робот. Під час виконання програми швидкістю подачі можна коригувати. Для корекції швидкості подачі використовується значення від 1% до 100%.

Одиниця виміру, у якій вказується швидкість подачі, залежить від формату руху, заданого у команді переміщення.

### **J P[1] 50% FINE**

У руху шарніра швидкість подачі вказується так:

- Визначається відсоток максимальної швидкості подачі в межах від 1% до 100%.
- Якщо в якості одиниці вимірювання використовується секунда (sec), встановіть час руху в межах від 0,1 до 3200 секунд. Це введення потрібно, коли важливо задати час, що витрачається на переміщення. Іноді операцію неможливо виконати у вказаний час.
- Якщо в якості одиниці вимірювання використовуються мілісекунди (msec), встановіть час руху від 1 до 32000 мілісекунд.

### **L P[1] 100mm/sec FINE**

У поступального переміщення, обертального руху або поступально-обертального руху швидкість подачі вказується наступним чином:

- Якщо одиниця, що використовується, – міліметри в секунду (mm/sec), задайте швидкість подачі в межах від 1 до 2000 міліметрів в секунду.
- Якщо одиниця, що використовується, – сантиметри за хвилину (cm/min), задайте швидкість подачі в межах від 1 до 12000 сантиметрів за хвилину.
- Якщо одиниця, що використовується, – дюйми в хвилину (inch/min), задайте швидкість подачі в межах від 0,1 до 4724,4 дюйма в хвилину.
- Якщо в якості одиниці вимірювання використовується секунда (sec), встановіть час руху в межах від 0,1 до 3200 секунд.
- Якщо в якості одиниці вимірювання використовуються мілісекунди (msec), встановіть час руху від 1 до 32000 мілісекунд.

Швидкість подачі можна встановити за допомогою реєстра. Завдяки цьому користувач може вказати швидкість подачі команди виконання операції після обчислення швидкості подачі за допомогою реєстра.

Формат, в якому відображається команда виконання операції, якщо швидкість подачі задана в реєстрі

- Joint J P [1] R [i] % FINE
- Linear L P[1] R[i]mm/sec FINE
- Circular C P[1] - P[2] R[i]mm/sec FINE
- Circle Arc A P[1] - R[i]mm/sec FINE
- Команда операції штабелювання
- J PAL\_1[A\_1] R[i]% FINE
- J PAL\_1[BTM] R[i]% FINE
- J PAL\_1[R\_1] R[i]% FINE

Щоб відрегулювати корекцію швидкості для нашого РТК, нам потрібно використати таку команду : J P[1] 50% FINE ACC80 – це зменшує швидкість нашої роботи у два рази.



Рис.6.6 – Команда для корекції швидкості

Ця команда задає процентний коефіцієнт прискорення/уповільнення під час переміщення. Фактично ця функція використовується для уповільнення переміщення. При зменшенні корекції прискорення час прискорення збільшується (прискорення та уповільнення виконуються повільніше). Для виконання потенційно небезпечних операцій, наприклад, зачерпування гарячої води слід використовувати значення менше 100%. У разі збільшення корекції прискорення час прискорення зменшується (прискорення та уповільнення виконуються швидше).

## КОМАНДА ТРАЄКТОРІЇ

J P[1] 50% Cnt10 PTH

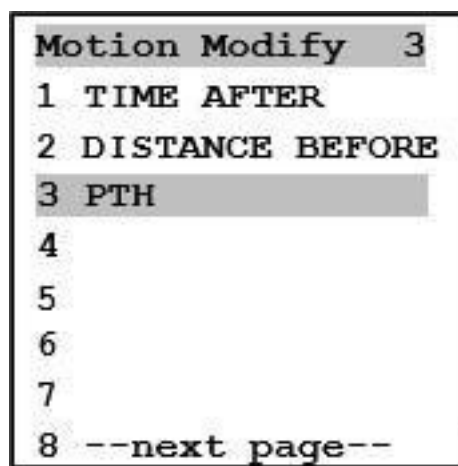


Рис.6.7 – Команда для траєкторії

При переміщенні робота на коротку відстань, швидкість неможливо збільшити до значення, вказаного в операторі переміщення. З цієї причини, плануючи оператор виконання операції, для якого використовується формат позиціонування FINE, слід розраховувати на "Досягну" швидкість, тобто швидкість, якої реально може досягти робот, а не задану швидкість. (Планування переміщення включає обчислення траєкторії, якою буде переміщатися робот, перед фактичним переміщенням.)

З використанням цієї команди планування операції проводиться із застосуванням досяжної швидкості оператора CNT.

Завдяки цій функції в нормальному режимі роботи досягаються такі ефекти:

Ефективність цієї функції підвищується в міру зменшення дистанції переміщення та зменшення значення Cnt (тобто значення n Cnt).

### 6.3. Розробка програм для реалізації алгоритмів регулювання

#### КОМАНДА ОЧІКУВАННЯ

Команда очікування використовується для припинення виконання програми на заданий період часу або до умови. Доступні два типи команд очікування.

Instruction 1	
1	Registers
2	I/O
3	IF/SELECT
4	WAIT
5	JMP/LBL
6	CALL
7	Palletizing
8	--next page--

Рис.6.8 – Команда очікування

- Команда очікування з обмеженням часу: Затримує виконання програми на певний період часу.
- Команда умовного очікування: Затримує виконання програми, доки не буде задоволено задану умову або закінчиться заданий період часу.

Для нашого РТК, нам потрібно оба варіанти, наприклад :

- робот зробить оберт до заготівки тільки , коли буде виконана умова для підняття заготівки, завдяки захвату.
- Світлофор почне світити та подавати звуковий сигнал , тільки якщо всі умови програми виконані
- Конвеєр почне працювати, якщо отримає сигнал від тензо-датчику, що мішок на ньому з потрібною вагою

## КОМАНДА ОЧІКУВАННЯ З ОБМЕЖЕННЯМ ЧАСУ WAIT

Команда очікування з обмеженням часу затримує виконання програми на заданий період часу (в секундах).



Рис.6.9 - Команда очікування з обмеженням часу

Приклад:

```
1: WAIT
2: WAIT 10.5sec
3: WAIT R[1]
```

## КОМАНДИ УМОВНОГО ОЧІКУВАННЯ WAIT (УМОВА) (ОБРОБКА)

Команда умовного очікування затримує виконання програми, доки не буде задоволено задану умову або закінчиться заданий період часу. Доступні два методивказівки обробки тайм-ауту:

- Якщо обробка не задана, виконання програми зупиняється, поки задана умова не буде задоволена.
- Якщо задана умова не задоволена до закінчення часу, заданого в позиції 14 WAIT timeout на екрані конфігурації системи, Timeout LBL[i] переноситься в задану мітку.

Команда умовного очікування регістру порівнює значення регістра з іншим значенням та затримує виконання програми, доки не задоволено умову порівняння



Рис.6.10 - Команда умовного очікування



- Якщо виконується команда операції, програма зупиняється до завершення операції.
- Курсор зупиняється на поточному рядку.
- Під час виконання команди аварійного припинення продовження виконання програми неможливе. Інформація про основну програму, що зберігається командою виклику програми, втрачається.

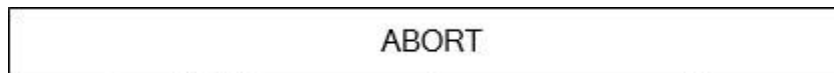


Рис.6.12 - Команда аварійного припинення роботи

### КОМАНДА АВАРІЙНОГО СИГНАЛУ КОРИСТУВАЧА UALM[i]

Команда аварійного сигналу користувача відображає повідомлення аварійного сигналу, що відповідає задалегідь заданому номеру аварійного сигналу в рядку індикації аварійного сигналу. Команда аварійного сигналу користувача припиняє програму, яка виконується зараз. Користувальницький аварійний сигнал задається на екрані налаштування аварійного сигналу користувача і це налаштування записується в системну змінну \$UALM\_MSG.

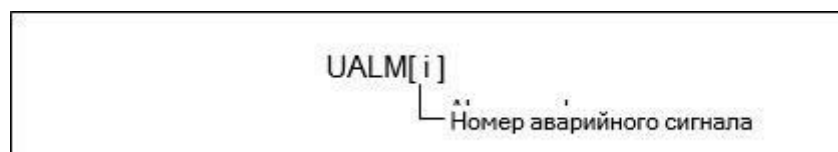


Рис.6.13 - Команда аварійного сигналу користувача

Приклад:

```
UALM[1] ($UALM_MSG[1] = WORK NOT FOUND
```

Якщо подивитись на блок схему, то при виникненні аварійної зупинки, є два варіанти

, коли програма працює:

- Коли система знаходить у цьому сенс , наприклад оператор раптом зайшов до РТК та система захисту , а саме світлові штори , завдяки даичку наявності предмету, відправили сигнал до контролеру робота та одразу РТК вимикається про що свідчить блок-схема.
- Оператор сам вирішує припинити РТК

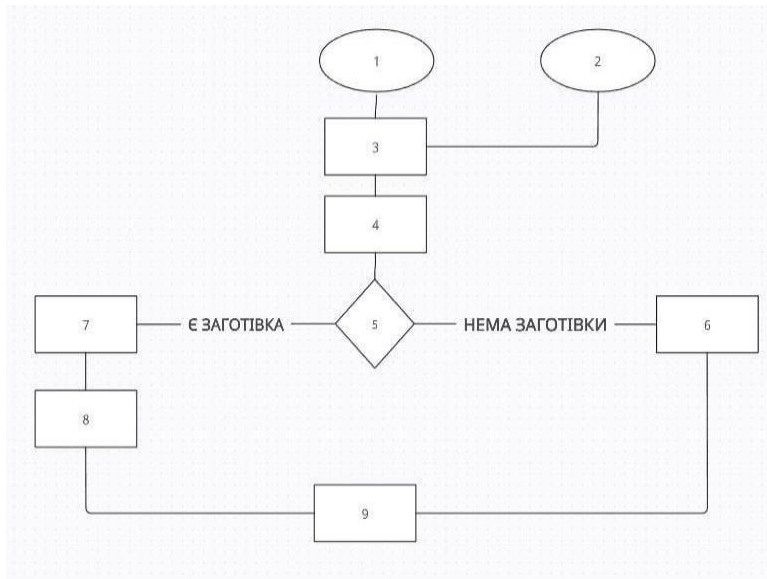


Рис.6.14 - Блок-схема аварійної зупинки **emergency stop** і світлові штори

### Пояснення та опис алгоритму аварійної зупинки :

- 11.Оператор натиснув на кнопку аварійної зупинки .12.Система захисту (світлові штори) помічає об'єкт. 13.Подається сигнал до контролеру.
  - 14.Подається звуковий сигнал світлофору та загорється жовтого кольору15.Система РТК зупиняє роботу.
  16. Робот повертається до вихідної позиції
  - 17.Робот викидає заготовлю у спеціально відведене місце, що передбачає тару для бракованої продукції.
  18. Робот повертається до вихідної позиції19.Оператор запскає основну програму
- У першому випадку, використовуємо команду аварійного припинення ABORT.

У другому випадку , використорвуємо команду аварійного припинення UALM[i]

## КОМАНДА ПАУЗИ PAUSE

- Якщо виконується команда операції, програма зупиняється до завершення операції.
- Курсор переміщується до наступного рядка. Під час перезапуску програма виконується з цього рядка.
- Якщо активний таймер програмного забезпечення, він зупиняється. Під час перезапуску програми програмний таймер запускається.
- Якщо команда подає імпульс, програма зупиняється після виконання цієї команди.
- Під час виконання будь-якої команди, за винятком команди виклику програми, програма зупиняється після виконання цієї команди. Команда виклику програми виконується під час перезапуску програми.



Рис.6.15 - Команда паузи

## КОМАНДА ПОВІДОМЛЕННЯ MESSAGE

Команда повідомлення видає повідомлення на екран користувача. (Інформація про екран користувача міститься в розділі Повідомлення може складатися максимум 324 символів, включаючи буквено-цифрові символи, зірочки (\*), символи підкреслення (\_) та символи @. Щоб додати коментар, натисніть [ENTER].

## MESSAGE [текст сообщения]

└─Текст сообщения может состоять максимум из 24 символов, включая буквенно-цифровые символы, звездочки (\*) знаки подчеркивания (\_), и символы «@», и т. д.

Рис.6.16 - Команда повідомлення

Приклад:

Для нашого РТК, ми використовуємо цю команду у таких випадках, наприклад:

- Якщо весь цикл праці РТК зроблен вірно, тоді оператору висвічує повідомлення, що все вірно.
- Якщо виникла аварійна ситуація, тоді оператору висвічує повідомлення, що проблеми з системою.

## КОМАНДА ВИКОНАННЯ ПРОГРАММИ

Під час виконання програми команда виконання програми починає виконання іншої програми.

- Різниця між нею та командою виклику програми полягає в тому, що при використанні команди виклику програми рядки, що йдуть за командою виклику, виконуються після того, як виконана викликана програма, тоді як при використанні команди виконання програми програма, яка починає виконання іншої програми, продовжує виконуватися одночасно із нею.
- Для синхронізації програм, які виконуються одночасно, використовуйте команду реєстру та команду очікування реєстра.
- Якщо спроба виконати програму, для якої задана та ж група переміщень, видається аварійний сигнал. Якщо це відбувається, вкажіть іншу групу переміщень.

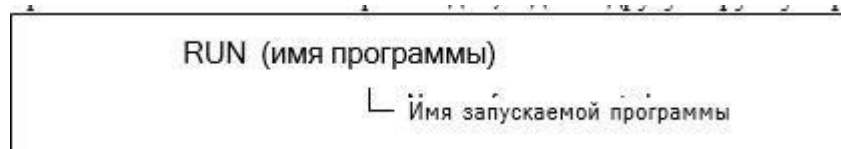


Рис.6.17 - Команда виконання програми

Пример PROG1

```
1: R[1]=0
2: RUN PROG2
3: J P[1] 100% FINE
4: J P[2] 100% FINE
5: WAIT R1[1]=1
```

MOTION GROUP[1,\*,\*,\*,\*,\*,\*]

PROG2

```
1: J P[3] 100% FINE
2: J P[4] 100% FINE
3: J P[5] 100% FINE
4: J P[6] 100% FINE
5: R[1]=1
```

MOTION GROUP[\* ,1,\*,\*,\*,\*,\*]

Для нашого РТК , ми використовуємо цю команду у таких випадках, наприклад:

- Нам потрібно , щоб паралельно працював робот та пневматичний захват.
- Нам потрібно , щоб паралельно працював конвеєр та пневматичний захват.
- Нам потрібно , щоб паралельно працював робот та конвеєр

## КОМАНДА ЦИКЛУ ENDFOR

Нижче наведено формат оператора ENDFOR.



```
ENDFOR
```

Рис.6.18 - Команда циклу Оператор ENDFOR

Цикл FOR/ENDFOR повторюється, поки задовольняються наведені нижче умови.

- Якщо задано TO, то значення лічильника циклу має бути меншим за цільове значення.
- Якщо зазначена умова задовольняється та задано TO, значення лічильника циклу збільшується на 1. Якщо зазначена умова задовольняється та задано DOWNTO, значення лічильника циклу зменшується на 1. Після цього курсор переміщається до наступного рядка об'єднаного оператора FOR.

Якщо вказана умова не задовольняється, курсор переміщається до наступного рядка цикл FOR/ENDFOR не виконується

Для нашого РТК , ми використовуємо цю команду у таких випадках, наприклад:

- Якщо весь цикл праці РТК зроблен вірно, тоді цикл повторюється.

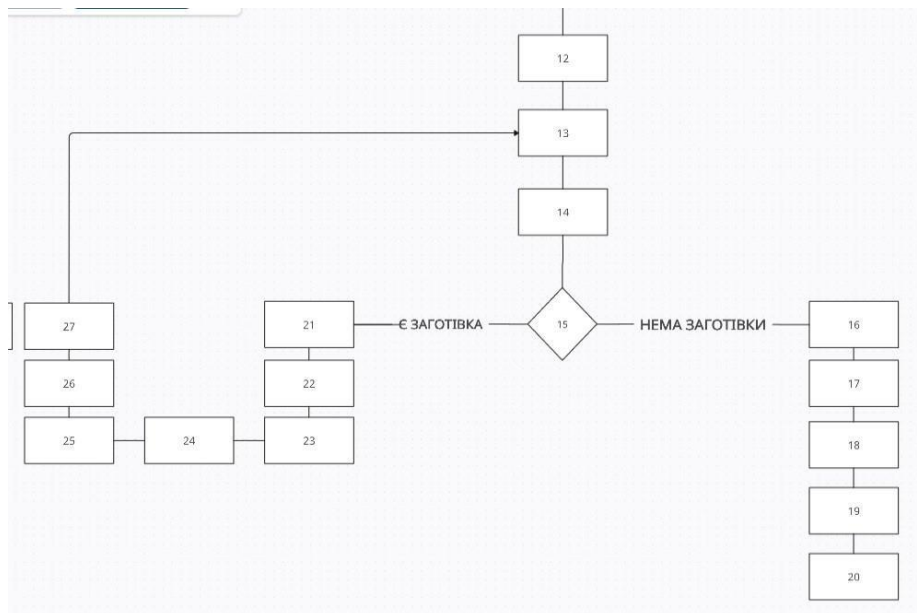


Рис.6.19 – Загальна блок-схема алгоритму праці РТК

Тільки що розібрали основні команди для реалізації нашого РТК та зараз більшдетальніше розглянемо програмування різних програм

## ЗУПИНКА І ВІДНОВЛЕННЯ ПРОГРАМИ

- Зупинка програми означає зупинку програми, що виконується. Зупинка програми викликається: Аварійний сигнал, що випадково стався під час виконання програми.
- Намірними діями оператора для зупинки програми, що виконується. Робот, що працює, зупиняється одним з наступних способів:
- Швидка зупинка: Робот швидко сповільнюється до повної зупинки.
- Повільна зупинка: Робот повільно сповільнюється до повної зупинки. Стану зупинки програми поділяються на два типи:
- Примусове завершення (закінчення): відображається стан завершення програми.

На екрані пульта навчання відображається ABORTED. Якщо основна програма завершена, а підпрограма виконується, інформація про повернення керування основною програмою втрачається.

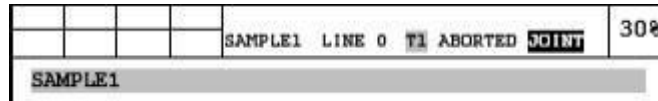


Рис.6.20 - ABORTED

- Зупинка (тимчасова зупинка): Виконання програми призупиняється. На екрані пульта навчання відображається індикація PAUSED. Можна перезапустити програму, що призупинена. Підпрограма, викликана командою виклику програми, повертає керування основною програмою.

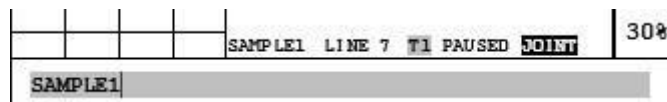


Рис.6.21 - Команда для зупинки

Для запуску цієї програми з іншого рядка або іншої програми необхідно виконати аварійне припинення програми, щоб вимкнути стан призупинення.

### Є три способи навмисного зупинення програми:

1. Натисканням кнопки аварійного зупинки на пультах навчання або на панелі оператора робота або відпуском перемикача аварійного блокування. Вхідний сигнал введення-виведення периферійного пристрою \*IMSTR.
2. Натисканням клавіші призупинення (HOLD) на пультах навчання або використанням вхідного сигналу \*HOLD периферійного вводу-виводу:  
Ці вхідні сигнали зупиняють виконання програми.
3. Обрати 1 ABORT(ALL) у меню функцій. Вхідний сигнал введення-виведення периферійного пристрою \* CSTOP1.

Цей спосіб виконує аварійне припинення програми.

## ЗУПИНКА АВАРІЙНИМ ЗУПИНЕННЯМ ТА ВІДНОВЛЕННЯ

Для негайного зупинення робота натисніть кнопку аварійного зупинки на панелі оператора або щиті управління робота або на пульті навчання (докладні відомості про типи зупинки наведено в розділі «Види зупинки робота» в Правилах техніки безпеки). При цьому видається аварійний сигнал аварійного зупинки.

### ПРОЦЕДУРА АВАРІЙНОЇ ЗУПИНКИ

КРОКИ:

- Натисніть кнопку аварійної зупинки на пульті навчання або на панелі оператора робота. В результаті програма буде зупинена, і на пульті навчання з'явиться PAUSED.
- Кнопка аварійного зупинки блокується в натиснутому положенні (увімкненому стані). На екрані пульта навчання з'являється повідомлення про аварійну зупинку. Світиться індикатор FAULT

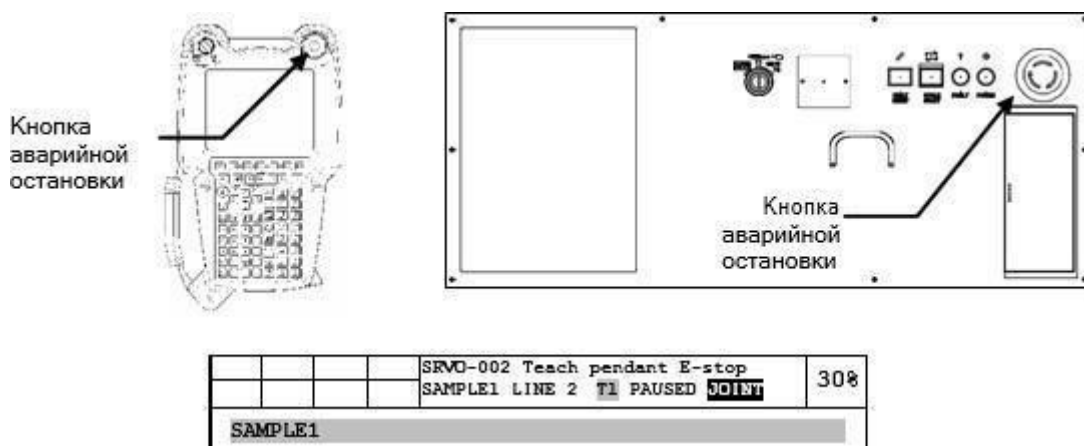
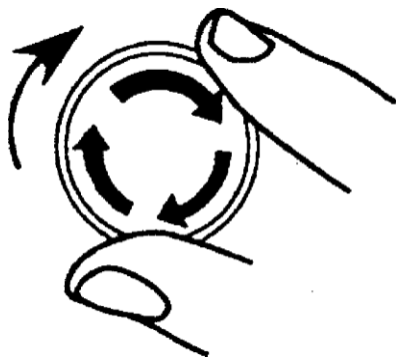


Рис.6.22 - Кнопка аварійного зупинки

- Усуньте причину аварійної зупинки. Наприклад, скоригуйте програму.



- Поверніть кнопку аварійного зупинки за годинниковою стрілкою, щоб розблокувати кнопку .
- Натисніть клавішу RESET на пульті навчання (або на панелі оператора). При цьому повідомлення аварійного сигналу зникне з екрана пульта навчання і світловий індикатор FAULT

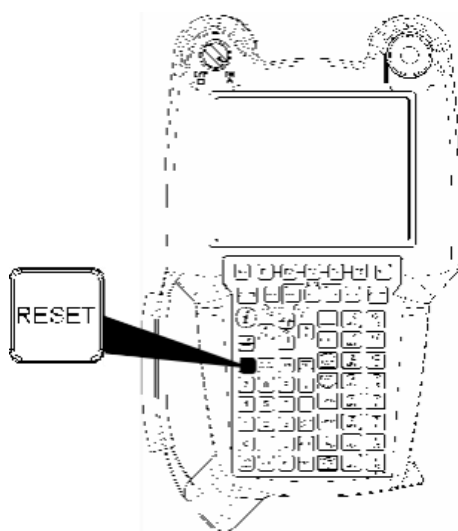


Рис.6.22 - Кнопка RESET

## **ЗУПИНКА З ПРИЗУПИНЕННЯМ ТА ВІДНОВЛЕННЯ**

Щоб повільно сповільнити роботу до повної зупинки, натисніть клавішу HOLD на пульті навчання або на панелі оператора.

При натисканні на кнопку [HOLD]:

- Робот повільно сповільнюється до повної зупинки (припинена програма).

- Під час спрацювання аварійного сигналу можна вимкнути живлення сервоприводу. Для цього виберіть SETUP General на екрані загальних налаштувань. (Див. розділ 3.16 «Загальні налаштування».)

## **ПРИЗУПИНЕННЯ ТА ВІДНОВЛЕННЯ**

### **КРОКИ:**

- Натисніть [HOLD] на пульті навчання. Програма, що виконується, тимчасово зупиниться, а на пульті навчання з'явиться PAUSED. Повідомлення про аварійний сигнал відображається лише, якщо увімкнено аварійний сигнал зупинки.



Рис.6.23 - Кнопка HOLD

- Щоб вимкнути стан зупинки, перезапустіть програму.

## **ПРИМУСОВЕ ЗАВЕРШЕННЯ (АВАРІЙНЕ ПРИПИНЕННЯ) ПРОГРАМИ**

### **КРОКИ:**

- Натисніть функціональну клавішу, щоб відобразити меню функцій, щоб вимкнути стан припинення та аварійне припинення програми.
- Виберіть ABORT (ALL). Буде виконано аварійне припинення програми, після якого стан зупинки відключиться.

## **ЗУПИНКА, ВИКЛИКАНА АВАРІЙНИМ СИГНАЛОМ**

Аварійний сигнал видається, якщо виявлено відмову, якщо подається сигнал аварійного зупинки або якщо інший аварійний сигнал надходить від периферійного

пристрою під час введення в режимі навчання або відтворення програми. Якщо видається аварійний сигнал, на пульті навчання з'являється відповідна індикація, і обробка, наприклад експлуатація робота і виконання програми, зупиняється для забезпечення безпеки.

## **ВІДОБРАЖЕННЯ АВАРІЙНОГО СИГНАЛУ**

Оператор може перевірити наявність аварійного сигналу за світловими індикаторами FAULT на пульті навчання та по першому та другому рядках на екрані панелі оператора. Вид аварійного сигналу визначається за кодом аварійного сигналу. Причина та спосіб усунення причин аварійного сигналу визначаються за кодом аварійного сигналу. У цьому посібнику опис кодів аварійних сигналів не наводиться

Дисплей та індикація аварійного сигналу

## **ЖУРНАЛ АВАРІЙНИХ СИГНАЛІВ**

Щоб відобразити журнал аварійних сигналів, виберіть екран журналу аварійних сигналів.

[4 ALARM].



- Дата спрацювання: Вказує дату спрацювання аварійного сигналу.

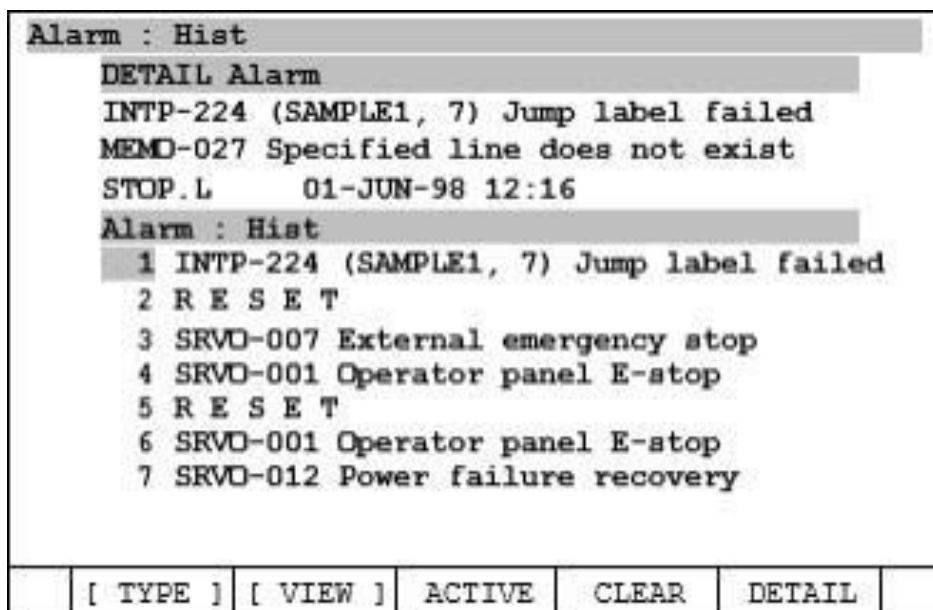


Рис.6.25 - Журнал аварійних сигналів

## СКИДАННЯ АВАРІЙНОГО СИГНАЛУ

Після усунення причини аварійного сигналу натисніть клавішу RESET, щоб скинути аварійний сигнал. Індикація аварійного сигналу в першому та другому рядках пульта навчання вимикається. Якщо вимкнено живлення сервоприводу, воно вмикається. Після скидання аварійного сигналу робот зазвичай вмикається.

## АВАРІЙНИЙ СИГНАЛ ПІД ЧАС ЗУПИНКИ

Функція аварійного сигналу під час зупинки видає аварійний сигнал та вимикає живлення сервосистеми, коли оператор натискає клавішу HOLD, щоб зупинити роботу.

У разі відмови в розділі [6 SETUP General] на екрані загальних установок задайте функцію аварійного сигналу.

## КОЛІРНА ІНДИКЦІЯ СТУПЕНЯ ТЯЖКОСТІ АВАРІЙНОГО СИГНАЛУ

На екрані аварійних сигналів кожному коду аварійних сигналів відповідає колір, який вказує на ступень відповідності аварійного сигналу.

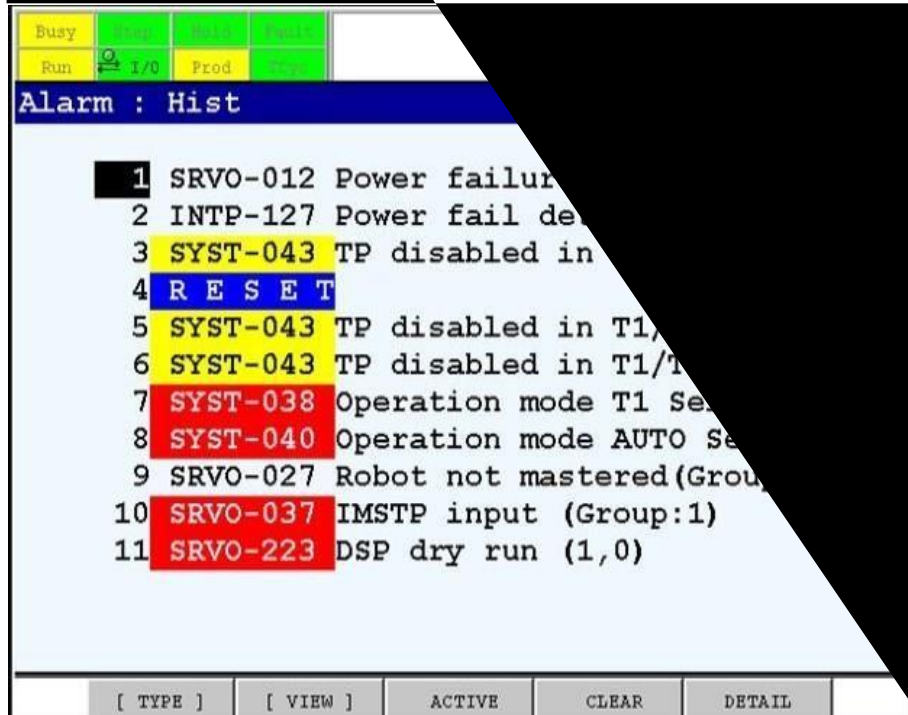


Рис.6.26 - Колірна індикція ступеня тяжкості аварійного сигналу

Колір, присвоєний кожному ступені тяжкості аварійного сигналу, вказаний нижче:

NONE	White
WARN	
PAUSE.L	Yellow
PAUSE.G	
STOP.L	Yellow
STOP.G	
SERVO	Red
SERVO2	
ABORT.L	Red
ABORT.G	
SYSTEM	Red
RESET(*)	Blue
SYST-026 System normal power up(*)	Blue

Рис.6.27 -.- Пояснення аварійних сигналів

## ЗАПУСК ПРОГРАМИ

Програму можна запустити:

- За допомогою пульта навчання. (сполученням клавіш [SHIFT] та [FWD] або [BWD])
- Натисніть кнопку START на панелі оператора.
- За допомогою периферійного пристрою (введення RSR 1 – 8, введення PROD\_START та введення START)

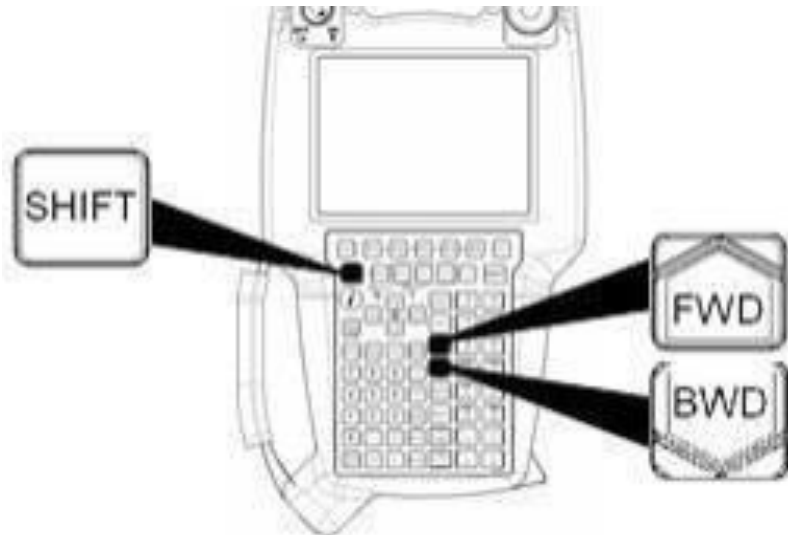


Рис.6.28 - Пояснення запуску

З метою безпеки програму можна запустити лише з пристрою, що має функцію керування рухом. Керування рухом можна змінити за допомогою вимикача пульта навчання та зміни режиму Remote/Local.

### СТРУКТУРА ( КОД) ГОЛОВНОЇ ПРОГРАМИ

```
# This file is used by your post processor (Universal_Robots)# to generate the appropriate file for your robot.
```

```
# You will find your robot program in the same folder where this file is located.# To avoid showing this file after program generation you can select:
```

```
# Tools->Options->tab Other , then, uncheck "Show programs in Notepad"
```

```

import sys
import os

sys.path.append(os.path.abspath(r""C:/RoboDK/Posts/"")) # temporarily add path to
POSTS folder

from Universal_Robots import *

def p(xyzrpw):
    x,y,z,r,p,w = xyzrpw
    a = r*math.pi/180.0
    b = p*math.pi/180.0
    c = w*math.pi/180.0
    ca = math.cos(a)
    sa = math.sin(a)
    cb = math.cos(b)
    sb = math.sin(b)
    cc = math.cos(c)
    sc = math.sin(c)

    return Mat([[cb*ca,ca*sc*sb-
cc*sa,sc*sa+cc*ca*sb,x],[cb*sa,cc*ca+sc*sb*sa,cc*sb*sa-ca*sc,y],[-
sb,cb*sc,cc*cb,z],[0.0,0.0,0.0,1.0]])

def make_program():

```

```

print('Total instructions: 281')

r = RobotPost(r"""Universal_Robots""",r"""UR5e""",6,
axes_type=['R','R','R','R','R','R'],ip_com=r"""127.0.0.1""")

r.ProgStart(r""" _ """)

r.RunMessage(r"""Program generated by RoboDK v4.2.2 for UR5e on 07/06/2023
20:43:51""", True)

r.RunMessage(r"""Using nominal kinematics.""", True)r.RunCode(r"""прога_1""",
True) r.RunCode(r"""рельса_2_линия""", True) r.RunCode(r"""прога_2""", True)
r.RunCode(r"""рельса_3_линия""", True) r.RunCode(r"""прога_3""", True)
r.RunCode(r"""линия_1_уход""", True) r.RunCode(r"""линия_2_уход""", True)
r.RunCode(r"""линия_3_уход""", True) r.ProgFinish(r""" _ """)

r.ProgStart(r"""прога_1""")

r.setFrame(p([0.000000,1000.000000,0.000000,0.000000,0.000000,-
0.000000]),8,r"""Frame 8""")

r.setTool(p([0.000000,0.000000,200.000000,0.000000,0.000000,0.000000]),-
1,r"""RG2_v2""")

```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
print('Done instruction: 10')sys.stdout.flush()
```

```
r.MoveL(p([-413.918807,-615.476836,-29.686094,90.000000,-0.000000,-180.000000]),[150.755698,-114.264853,-111.290861,-44.444286,90.000000,-29.244302],[0.0,1.0,0.0])
```

```
r.MoveL(p([-413.918807,-615.476836,-132.300141,90.000000,-0.000000,-180.000000]),[150.755698,-126.165724,-113.430635,-30.403641,90.000000,-29.244302],[0.0,1.0,0.0])
```

```
r.RunMessage(r"" Attach to Eyes_Mount_Robot_v2 "" , True) r.MoveL(p([-413.918807,-615.476836,-29.686094,90.000000,-0.000000,-180.000000]),[150.755698,-114.264853,-111.290861,-44.444286,90.000000,-29.244302],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([760.178663,-1164.416149,287.900000,-90.000000,-0.000000,-180.000000]),[-2.335622,-133.372153,-29.416777,-107.211070,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.MoveL(p([760.178663,-1164.416149,158.540950,-90.000000,-0.000000,-180.000000]),[-2.335622,-130.374677,-53.738454,-85.886869,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from Eyes_Mount_Robot_v2""", True)
```

```
r.MoveL(p([760.178663,-1164.416149,287.900000,-90.000000,-0.000000,-180.000000]),[-2.335622,-133.372153,-29.416777,-107.211070,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
print('Done instruction: 20')sys.stdout.flush()
```

```
r.MoveL(p([-419.180943,-726.901881,-38.315541,90.000000,0.000000,180.000000]),[162.368287,-108.508762,-122.351480,-39.139758,90.000000,-17.631713],[0.0,1.0,0.0])
```

```
r.MoveL(p([-419.180943,-726.901881,-134.591167,90.000000,0.000000,180.000000]),[162.368287,-121.787864,-124.357921,-23.854215,90.000000,-17.631713],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Attach to RG2_v2""", True)r.MoveL(p([-419.180943,-726.901881,-38.315541,90.000000,0.000000,180.000000]),[162.368287,-108.508762,-122.351480,-39.139758,90.000000,-17.631713],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([760.178663,-1030.677547,287.900000,-90.000000,-0.000000,-180.000000]),[7.779974,-129.267181,-36.571696,-104.161123,90.000000,7.779974],[0.0,1.0,0.0])
```

```
r.MoveL(p([760.178663,-1030.677547,155.940327,-90.000000,-0.000000,-180.000000]),[7.779974,-127.632409,-58.740011,-83.627581,90.000000,7.779974],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([760.178663,-1030.677547,287.900000,-90.000000,-0.000000,-180.000000]),[7.779974,-129.267181,-36.571696,-104.161123,90.000000,7.779974],[0.0,1.0,0.0])
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
print('Done instruction: 30')sys.stdout.flush()
```

```
r.MoveL(p([-415.167669,-847.455528,-38.315541,90.000000,0.000000,180.000000]),[177.365463,-102.272522,-131.520826,-36.206652,90.000000,-2.634537],[0.0,1.0,0.0])
```

r.MoveL(p([-415.167669,-847.455528,-  
129.835887,90.000000,0.000000,180.000000]),[177.365463,-117.179210,-133.788804,-  
19.031986,90.000000,-2.634537],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)r.MoveL(p([-415.167669,-847.455528,-  
38.315541,90.000000,0.000000,180.000000]),[177.365463,-102.272522,-131.520826,-  
36.206652,90.000000,-2.634537],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-  
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-  
21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-  
180.000000]),[0.000000,-90.000000,-90.000000,-  
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([760.178663,-906.852813,287.900000,-90.000000,0.000000,-  
180.000000]),[17.009296,-130.421117,-34.590074,-  
104.988809,90.000000,17.009296],[0.0,1.0,0.0])

r.MoveL(p([760.178663,-906.852813,160.480878,-90.000000,0.000000,-  
180.000000]),[17.009296,-128.375194,-56.832503,-  
84.792303,90.000000,17.009296],[0.0,1.0,0.0])

r.RunMessage(r"" Detach from RG2\_v2"", True)

r.MoveL(p([760.178663,-906.852813,287.900000,-90.000000,0.000000,-  
180.000000]),[17.009296,-130.421117,-34.590074,-  
104.988809,90.000000,17.009296],[0.0,1.0,0.0])

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-  
180.000000]),[0.000000,-90.000000,-90.000000,-  
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
print('Done instruction: 40')sys.stdout.flush()
```

```
r.MoveL(p([-535.179980,-615.476836,-29.686094,90.000000,0.000000,180.000000]),[155.973180,-124.172644,-94.332579,-51.494776,90.000000,-24.026820],[0.0,1.0,0.0])
```

```
r.MoveL(p([-535.179980,-624.329779,-127.541240,90.000000,0.000000,180.000000]),[156.695979,-132.947352,-97.292270,-39.760378,90.000000,-23.304021],[0.0,1.0,0.0])
```

```
r.RunMessage(r"" Attach to RG2_v2"" , True)r.MoveL(p([-535.179980,-615.476836,-29.686094,90.000000,0.000000,180.000000]),[155.973180,-124.172644,-94.332579,-51.494776,90.000000,-24.026820],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-906.852813,287.900000,-90.000000,0.000000,-180.000000]),[19.692662,-111.839366,-63.700709,-94.459924,90.000000,19.692662],[0.0,1.0,0.0])
```

```

r.MoveL(p([655.833708,-906.852813,162.639108,-90.000000,0.000000,-
180.000000]),[19.692662,-113.222055,-80.166806,-
76.611139,90.000000,19.692662],[0.0,1.0,0.0])

r.RunMessage(r""""Detach from RG2_v2""", True)

r.MoveL(p([655.833708,-906.852813,287.900000,-90.000000,0.000000,-
180.000000]),[19.692662,-111.839366,-63.700709,-
94.459924,90.000000,19.692662],[0.0,1.0,0.0])

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

print('Done instruction: 50')sys.stdout.flush()

r.MoveL(p([-535.179980,-738.862340,-29.686094,90.000000,-0.000000,-
180.000000]),[166.925413,-117.448180,-105.998672,-46.553148,90.000000,-
13.074587],[0.0,1.0,0.0])

r.MoveL(p([-535.179980,-734.405721,-135.475556,90.000000,-0.000000,-
180.000000]),[166.497973,-129.087717,-107.720801,-33.191482,90.000000,-
13.502027],[0.0,1.0,0.0])

r.RunMessage(r""""Attach to RG2_v2""", True)

r.MoveL(p([-535.179980,-738.862340,-29.686094,90.000000,-0.000000,-
180.000000]),[166.925413,-117.448180,-105.998672,-46.553148,90.000000,-
13.074587],[0.0,1.0,0.0])

```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-1032.968117,287.900000,-90.000000,0.000000,-180.000000]),[8.834456,-110.962945,-64.924604,-94.112452,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-1032.968117,155.282864,-90.000000,0.000000,-180.000000]),[8.834456,-112.688251,-82.040881,-75.270869,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([655.833708,-1032.968117,287.900000,-90.000000,0.000000,-180.000000]),[8.834456,-110.962945,-64.924604,-94.112452,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
print('Done instruction: 60')sys.stdout.flush()
```

```
r.MoveL(p([-535.179980,-841.735050,-29.686094,90.000000,-0.000000,-180.000000]),[177.344524,-113.547783,-112.459393,-43.992824,90.000000,-2.655476],[0.0,1.0,0.0])
```

r.MoveL(p([-535.179980,-844.277022,-135.217132,90.000000,-0.000000,-180.000000]),[177.613194,-125.965018,-114.744725,-29.290257,90.000000,-2.386806],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)

r.MoveL(p([-535.179980,-841.735050,-29.686094,90.000000,-0.000000,-180.000000]),[177.344524,-113.547783,-112.459393,-43.992824,90.000000,-2.655476],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([655.833708,-1159.173644,287.900000,-90.000000,-0.000000,-180.000000]),[-2.250261,-113.767702,-60.959707,-95.272592,90.000000,-2.250261],[0.0,1.0,0.0])

r.MoveL(p([655.833708,-1159.173644,153.902349,-90.000000,-0.000000,-180.000000]),[-2.250261,-115.198714,-78.599551,-76.201735,90.000000,-2.250261],[0.0,1.0,0.0])

r.RunMessage(r"" Detach from RG2\_v2"", True)

r.MoveL(p([655.833708,-1159.173644,287.900000,-90.000000,-0.000000,-180.000000]),[-2.250261,-113.767702,-60.959707,-95.272592,90.000000,-2.250261],[0.0,1.0,0.0])

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
print('Done instruction: 70')sys.stdout.flush()
```

```
r.MoveL(p([-644.620589,-621.733168,-29.686094,90.000000,-0.000000,-180.000000]),[159.868984,-134.247893,-75.904078,-59.848029,90.000000,-20.131016],[0.0,1.0,0.0])
```

```
r.MoveL(p([-644.620589,-621.733168,-126.649245,90.000000,-0.000000,-180.000000]),[159.868984,-141.953950,-77.920510,-50.125540,90.000000,-20.131016],[0.0,1.0,0.0])
```

```
r.RunMessage(r"" Attach to RG2_v2"" , True)
```

```
r.MoveL(p([-644.620589,-621.733168,-29.686094,90.000000,-0.000000,-180.000000]),[159.868984,-134.247893,-75.904078,-59.848029,90.000000,-20.131016],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-906.852813,287.900000,-90.000000,-0.000000,-180.000000]),[24.280089,-94.058391,-85.785941,-90.155668,90.000000,24.280089],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-906.852813,165.171094,-90.000000,-0.000000,-180.000000]),[24.280089,-96.600934,-101.199458,-72.199608,90.000000,24.280089],[0.0,1.0,0.0])
```

```

r.RunMessage(r""""Detach from RG2_v2""", True)

    r.MoveL(p([530.664479,-906.852813,287.900000,-90.000000,-0.000000,-
180.000000]),[24.280089,-94.058391,-85.785941,-
90.155668,90.000000,24.280089],[0.0,1.0,0.0])

    r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

    r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

print('Done instruction: 80')sys.stdout.flush()

    r.MoveL(p([-644.620589,-736.638137,-29.686094,90.000000,-0.000000,-
180.000000]),[168.813520,-128.282636,-86.933505,-54.783859,90.000000,-
11.186480],[0.0,1.0,0.0])

    r.MoveL(p([-644.620589,-734.148046,-127.850924,90.000000,-0.000000,-
180.000000]),[168.608956,-136.982818,-88.709434,-44.307748,90.000000,-
11.391044],[0.0,1.0,0.0])

r.RunMessage(r""""Attach to RG2_v2""", True)

    r.MoveL(p([-644.620589,-736.638137,-29.686094,90.000000,-0.000000,-
180.000000]),[168.813520,-128.282636,-86.933505,-54.783859,90.000000,-
11.186480],[0.0,1.0,0.0])

    r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-1039.203895,287.900000,-90.000000,-0.000000,180.000000]),[10.282636,-93.128425,-86.779059,-90.092516,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-1039.203895,163.143298,-90.000000,-0.000000,180.000000]),[10.282636,-95.785615,-102.426441,-71.787944,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([530.664479,-1039.203895,287.900000,-90.000000,-0.000000,180.000000]),[10.282636,-93.128425,-86.779059,-90.092516,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
print('Done instruction: 90')sys.stdout.flush()
```

```
r.MoveL(p([-644.620589,-844.606627,-29.686094,90.000000,-0.000000,-180.000000]),[178.043944,-124.615269,-93.544599,-51.840132,90.000000,-1.956056],[0.0,1.0,0.0])
```

r.MoveL(p([-644.620589,-844.606627,-128.570202,90.000000,-0.000000,-180.000000]),[178.043944,-133.877730,-95.521097,-40.601173,90.000000,-1.956056],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)

r.MoveL(p([-644.620589,-844.606627,-29.686094,90.000000,-0.000000,-180.000000]),[178.043944,-124.615269,-93.544599,-51.840132,90.000000,-1.956056],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([530.664479,-1155.884259,287.900000,-90.000000,-0.000000,180.000000]),[-2.424082,-96.056313,-83.597199,-90.346488,90.000000,-2.424082],[0.0,1.0,0.0])

r.MoveL(p([530.664479,-1155.884259,160.125815,-90.000000,-0.000000,180.000000]),[-2.424082,-98.737977,-99.512466,-71.749557,90.000000,-2.424082],[0.0,1.0,0.0])

r.RunMessage(r""Detach from RG2\_v2"", True) r.MoveL(p([530.664479,-1155.884259,287.900000,-90.000000,-0.000000,180.000000]),[-2.424082,-96.056313,-83.597199,-90.346488,90.000000,-2.424082],[0.0,1.0,0.0])

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

```

r.ProgStart(r""""прога_2""")

    r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

print('Done instruction: 100')sys.stdout.flush()

    r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

    r.MoveL(p([-413.918807,-615.476836,-29.686094,90.000000,-0.000000,-
180.000000]),[150.755698,-114.264853,-111.290861,-44.444286,90.000000,-
29.244302],[0.0,1.0,0.0])

    r.MoveL(p([-413.918807,-615.476836,-132.300141,90.000000,-0.000000,-
180.000000]),[150.755698,-126.165724,-113.430635,-30.403641,90.000000,-
29.244302],[0.0,1.0,0.0])

r.RunMessage(r""""Attach to Eyes_Mount_Robot_v2""", True) r.MoveL(p([-
413.918807,-615.476836,-29.686094,90.000000,-0.000000,-
180.000000]),[150.755698,-114.264853,-111.290861,-44.444286,90.000000,-
29.244302],[0.0,1.0,0.0])

    r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

    r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

```

```
r.MoveJ(p([760.178663,-1164.416149,287.900000,-90.000000,-0.000000,-180.000000]),[-2.335622,-133.372153,-29.416777,-107.211070,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.MoveL(p([760.178663,-1164.416149,158.540950,-90.000000,-0.000000,-180.000000]),[-2.335622,-130.374677,-53.738454,-85.886869,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.RunMessage(r""Detach from Eyes_Mount_Robot_v2"", True)
```

```
r.MoveL(p([760.178663,-1164.416149,287.900000,-90.000000,-0.000000,-180.000000]),[-2.335622,-133.372153,-29.416777,-107.211070,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
print('Done instruction: 110')sys.stdout.flush()
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveL(p([-419.180943,-726.901881,-38.315541,90.000000,0.000000,180.000000]),[162.368287,-108.508762,-122.351480,-39.139758,90.000000,-17.631713],[0.0,1.0,0.0])
```

```
r.MoveL(p([-419.180943,-726.901881,-134.591167,90.000000,0.000000,180.000000]),[162.368287,-121.787864,-124.357921,-23.854215,90.000000,-17.631713],[0.0,1.0,0.0])
```

```
r.RunMessage(r""Attach to RG2_v2"", True)
```

```
r.MoveL(p([-419.180943,-726.901881,-  
38.315541,90.000000,0.000000,180.000000]),[162.368287,-108.508762,-122.351480,-  
39.139758,90.000000,-17.631713],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-  
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-  
21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-  
180.000000]),[0.000000,-90.000000,-90.000000,-  
90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([760.178663,-1030.677547,287.900000,-90.000000,-0.000000,-  
180.000000]),[7.779974,-129.267181,-36.571696,-  
104.161123,90.000000,7.779974],[0.0,1.0,0.0])
```

```
r.MoveL(p([760.178663,-1030.677547,155.940327,-90.000000,-0.000000,-  
180.000000]),[7.779974,-127.632409,-58.740011,-  
83.627581,90.000000,7.779974],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([760.178663,-1030.677547,287.900000,-90.000000,-0.000000,-  
180.000000]),[7.779974,-129.267181,-36.571696,-  
104.161123,90.000000,7.779974],[0.0,1.0,0.0])
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-  
180.000000]),[0.000000,-90.000000,-90.000000,-  
90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
print('Done instruction: 120')sys.stdout.flush()
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-  
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-  
21.160989],[0.0,1.0,0.0])
```

r.MoveL(p([-415.167669,-847.455528,-  
38.315541,90.000000,0.000000,180.000000]),[177.365463,-102.272522,-131.520826,-  
36.206652,90.000000,-2.634537],[0.0,1.0,0.0])

r.MoveL(p([-415.167669,-847.455528,-  
129.835887,90.000000,0.000000,180.000000]),[177.365463,-117.179210,-133.788804,-  
19.031986,90.000000,-2.634537],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)r.MoveL(p([-415.167669,-847.455528,-  
38.315541,90.000000,0.000000,180.000000]),[177.365463,-102.272522,-131.520826,-  
36.206652,90.000000,-2.634537],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-  
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-  
21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-  
180.000000]),[0.000000,-90.000000,-90.000000,-  
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([760.178663,-906.852813,287.900000,-90.000000,0.000000,-  
180.000000]),[17.009296,-130.421117,-34.590074,-  
104.988809,90.000000,17.009296],[0.0,1.0,0.0])

r.MoveL(p([760.178663,-906.852813,160.480878,-90.000000,0.000000,-  
180.000000]),[17.009296,-128.375194,-56.832503,-  
84.792303,90.000000,17.009296],[0.0,1.0,0.0])

r.RunMessage(r"" Detach from RG2\_v2"", True)

r.MoveL(p([760.178663,-906.852813,287.900000,-90.000000,0.000000,-  
180.000000]),[17.009296,-130.421117,-34.590074,-  
104.988809,90.000000,17.009296],[0.0,1.0,0.0])

```

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

print('Done instruction: 130')sys.stdout.flush()

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

r.MoveL(p([-535.179980,-615.476836,-
29.686094,90.000000,0.000000,180.000000]),[155.973180,-124.172644,-94.332579,-
51.494776,90.000000,-24.026820],[0.0,1.0,0.0])

r.MoveL(p([-535.179980,-624.329779,-
127.541240,90.000000,0.000000,180.000000]),[156.695979,-132.947352,-97.292270,-
39.760378,90.000000,-23.304021],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2_v2"" , True)r.MoveL(p([-535.179980,-615.476836,-
29.686094,90.000000,0.000000,180.000000]),[155.973180,-124.172644,-94.332579,-
51.494776,90.000000,-24.026820],[0.0,1.0,0.0])

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

```

```
r.MoveL(p([655.833708,-906.852813,287.900000,-90.000000,0.000000,-180.000000]),[19.692662,-111.839366,-63.700709,-94.459924,90.000000,19.692662],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-906.852813,162.639108,-90.000000,0.000000,-180.000000]),[19.692662,-113.222055,-80.166806,-76.611139,90.000000,19.692662],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([655.833708,-906.852813,287.900000,-90.000000,0.000000,-180.000000]),[19.692662,-111.839366,-63.700709,-94.459924,90.000000,19.692662],[0.0,1.0,0.0])
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
print('Done instruction: 140')sys.stdout.flush()
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveL(p([-535.179980,-738.862340,-29.686094,90.000000,-0.000000,-180.000000]),[166.925413,-117.448180,-105.998672,-46.553148,90.000000,-13.074587],[0.0,1.0,0.0])
```

```
r.MoveL(p([-535.179980,-734.405721,-135.475556,90.000000,-0.000000,-180.000000]),[166.497973,-129.087717,-107.720801,-33.191482,90.000000,-13.502027],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Attach to RG2_v2""", True)
```

```
r.MoveL(p([-535.179980,-738.862340,-29.686094,90.000000,-0.000000,-180.000000]),[166.925413,-117.448180,-105.998672,-46.553148,90.000000,-13.074587],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-1032.968117,287.900000,-90.000000,0.000000,-180.000000]),[8.834456,-110.962945,-64.924604,-94.112452,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-1032.968117,155.282864,-90.000000,0.000000,-180.000000]),[8.834456,-112.688251,-82.040881,-75.270869,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([655.833708,-1032.968117,287.900000,-90.000000,0.000000,-180.000000]),[8.834456,-110.962945,-64.924604,-94.112452,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
print('Done instruction: 150')sys.stdout.flush()
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

r.MoveL(p([-535.179980,-841.735050,-29.686094,90.000000,-0.000000,-180.000000]),[177.344524,-113.547783,-112.459393,-43.992824,90.000000,-2.655476],[0.0,1.0,0.0])

r.MoveL(p([-535.179980,-844.277022,-135.217132,90.000000,-0.000000,-180.000000]),[177.613194,-125.965018,-114.744725,-29.290257,90.000000,-2.386806],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)

r.MoveL(p([-535.179980,-841.735050,-29.686094,90.000000,-0.000000,-180.000000]),[177.344524,-113.547783,-112.459393,-43.992824,90.000000,-2.655476],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([655.833708,-1159.173644,287.900000,-90.000000,-0.000000,-180.000000]),[-2.250261,-113.767702,-60.959707,-95.272592,90.000000,-2.250261],[0.0,1.0,0.0])

r.MoveL(p([655.833708,-1159.173644,153.902349,-90.000000,-0.000000,-180.000000]),[-2.250261,-115.198714,-78.599551,-76.201735,90.000000,-2.250261],[0.0,1.0,0.0])

r.RunMessage(r"" Detach from RG2\_v2"", True)

r.MoveL(p([655.833708,-1159.173644,287.900000,-90.000000,-0.000000,-180.000000]),[-2.250261,-113.767702,-60.959707,-95.272592,90.000000,-2.250261],[0.0,1.0,0.0])

```

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

print('Done instruction: 160')sys.stdout.flush()

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

r.MoveL(p([-644.620589,-621.733168,-29.686094,90.000000,-0.000000,-
180.000000]),[159.868984,-134.247893,-75.904078,-59.848029,90.000000,-
20.131016],[0.0,1.0,0.0])

r.MoveL(p([-644.620589,-621.733168,-126.649245,90.000000,-0.000000,-
180.000000]),[159.868984,-141.953950,-77.920510,-50.125540,90.000000,-
20.131016],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2_v2"", True)

r.MoveL(p([-644.620589,-621.733168,-29.686094,90.000000,-0.000000,-
180.000000]),[159.868984,-134.247893,-75.904078,-59.848029,90.000000,-
20.131016],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([530.664479,-906.852813,287.900000,-90.000000,-0.000000,-
180.000000]),[24.280089,-94.058391,-85.785941,-
90.155668,90.000000,24.280089],[0.0,1.0,0.0])

```

```

r.MoveL(p([530.664479,-906.852813,165.171094,-90.000000,-0.000000,-
180.000000]),[24.280089,-96.600934,-101.199458,-
72.199608,90.000000,24.280089],[0.0,1.0,0.0])

r.RunMessage(r""""Detach from RG2_v2""", True)

r.MoveL(p([530.664479,-906.852813,287.900000,-90.000000,-0.000000,-
180.000000]),[24.280089,-94.058391,-85.785941,-
90.155668,90.000000,24.280089],[0.0,1.0,0.0])

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

print('Done instruction: 170')sys.stdout.flush()

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

r.MoveL(p([-644.620589,-736.638137,-29.686094,90.000000,-0.000000,-
180.000000]),[168.813520,-128.282636,-86.933505,-54.783859,90.000000,-
11.186480],[0.0,1.0,0.0])

r.MoveL(p([-644.620589,-734.148046,-127.850924,90.000000,-0.000000,-
180.000000]),[168.608956,-136.982818,-88.709434,-44.307748,90.000000,-
11.391044],[0.0,1.0,0.0])

r.RunMessage(r""""Attach to RG2_v2""", True)

r.MoveL(p([-644.620589,-736.638137,-29.686094,90.000000,-0.000000,-
180.000000]),[168.813520,-128.282636,-86.933505,-54.783859,90.000000,-
11.186480],[0.0,1.0,0.0])

```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-1039.203895,287.900000,-90.000000,-0.000000,180.000000]),[10.282636,-93.128425,-86.779059,-90.092516,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-1039.203895,163.143298,-90.000000,-0.000000,180.000000]),[10.282636,-95.785615,-102.426441,-71.787944,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([530.664479,-1039.203895,287.900000,-90.000000,-0.000000,180.000000]),[10.282636,-93.128425,-86.779059,-90.092516,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
print('Done instruction: 180')sys.stdout.flush()
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

r.MoveL(p([-644.620589,-844.606627,-29.686094,90.000000,-0.000000,-180.000000]),[178.043944,-124.615269,-93.544599,-51.840132,90.000000,-1.956056],[0.0,1.0,0.0])

r.MoveL(p([-644.620589,-844.606627,-128.570202,90.000000,-0.000000,-180.000000]),[178.043944,-133.877730,-95.521097,-40.601173,90.000000,-1.956056],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)

r.MoveL(p([-644.620589,-844.606627,-29.686094,90.000000,-0.000000,-180.000000]),[178.043944,-124.615269,-93.544599,-51.840132,90.000000,-1.956056],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([530.664479,-1155.884259,287.900000,-90.000000,-0.000000,180.000000]),[-2.424082,-96.056313,-83.597199,-90.346488,90.000000,-2.424082],[0.0,1.0,0.0])

r.MoveL(p([530.664479,-1155.884259,160.125815,-90.000000,-0.000000,180.000000]),[-2.424082,-98.737977,-99.512466,-71.749557,90.000000,-2.424082],[0.0,1.0,0.0])

r.RunMessage(r""Detach from RG2\_v2"", True) r.MoveL(p([530.664479,-1155.884259,287.900000,-90.000000,-0.000000,180.000000]),[-2.424082,-96.056313,-83.597199,-90.346488,90.000000,-2.424082],[0.0,1.0,0.0])

```

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

print('Done instruction: 190')sys.stdout.flush() r.ProgFinish(r""""прога_2""")

r.ProgStart(r""""прога_3""")

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

r.MoveL(p([-413.918807,-615.476836,-29.686094,90.000000,-0.000000,-
180.000000]),[150.755698,-114.264853,-111.290861,-44.444286,90.000000,-
29.244302],[0.0,1.0,0.0])

r.MoveL(p([-413.918807,-615.476836,-132.300141,90.000000,-0.000000,-
180.000000]),[150.755698,-126.165724,-113.430635,-30.403641,90.000000,-
29.244302],[0.0,1.0,0.0])

r.RunMessage(r""""Attach to Eyes_Mount_Robot_v2""", True) r.MoveL(p([-
413.918807,-615.476836,-29.686094,90.000000,-0.000000,-
180.000000]),[150.755698,-114.264853,-111.290861,-44.444286,90.000000,-
29.244302],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([760.178663,-1164.416149,287.900000,-90.000000,-0.000000,-180.000000]),[-2.335622,-133.372153,-29.416777,-107.211070,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.MoveL(p([760.178663,-1164.416149,158.540950,-90.000000,-0.000000,-180.000000]),[-2.335622,-130.374677,-53.738454,-85.886869,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
r.RunMessage(r""Detach from Eyes_Mount_Robot_v2"", True)
```

```
r.MoveL(p([760.178663,-1164.416149,287.900000,-90.000000,-0.000000,-180.000000]),[-2.335622,-133.372153,-29.416777,-107.211070,90.000000,-2.335622],[0.0,1.0,0.0])
```

```
print('Done instruction: 200')sys.stdout.flush()
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveL(p([-419.180943,-726.901881,-38.315541,90.000000,0.000000,180.000000]),[162.368287,-108.508762,-122.351480,-39.139758,90.000000,-17.631713],[0.0,1.0,0.0])
```

```

r.MoveL(p([-419.180943,-726.901881,-
134.591167,90.000000,0.000000,180.000000]),[162.368287,-121.787864,-124.357921,-
23.854215,90.000000,-17.631713],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2_v2"" , True)r.MoveL(p([-419.180943,-726.901881,-
38.315541,90.000000,0.000000,180.000000]),[162.368287,-108.508762,-122.351480,-
39.139758,90.000000,-17.631713],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-
21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveJ(p([760.178663,-1030.677547,287.900000,-90.000000,-0.000000,-
180.000000]),[7.779974,-129.267181,-36.571696,-
104.161123,90.000000,7.779974],[0.0,1.0,0.0])

r.MoveL(p([760.178663,-1030.677547,155.940327,-90.000000,-0.000000,-
180.000000]),[7.779974,-127.632409,-58.740011,-
83.627581,90.000000,7.779974],[0.0,1.0,0.0])

r.RunMessage(r""Detach from RG2_v2"" , True)

r.MoveL(p([760.178663,-1030.677547,287.900000,-90.000000,-0.000000,-
180.000000]),[7.779974,-129.267181,-36.571696,-
104.161123,90.000000,7.779974],[0.0,1.0,0.0])

print('Done instruction: 210')sys.stdout.flush()

```

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveL(p([-415.167669,-847.455528,-38.315541,90.000000,0.000000,180.000000]),[177.365463,-102.272522,-131.520826,-36.206652,90.000000,-2.634537],[0.0,1.0,0.0])

r.MoveL(p([-415.167669,-847.455528,-129.835887,90.000000,0.000000,180.000000]),[177.365463,-117.179210,-133.788804,-19.031986,90.000000,-2.634537],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)r.MoveL(p([-415.167669,-847.455528,-38.315541,90.000000,0.000000,180.000000]),[177.365463,-102.272522,-131.520826,-36.206652,90.000000,-2.634537],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([760.178663,-906.852813,287.900000,-90.000000,0.000000,-180.000000]),[17.009296,-130.421117,-34.590074,-104.988809,90.000000,17.009296],[0.0,1.0,0.0])

```
r.MoveL(p([760.178663,-906.852813,160.480878,-90.000000,0.000000,-  
180.000000]),[17.009296,-128.375194,-56.832503,-  
84.792303,90.000000,17.009296],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([760.178663,-906.852813,287.900000,-90.000000,0.000000,-  
180.000000]),[17.009296,-130.421117,-34.590074,-  
104.988809,90.000000,17.009296],[0.0,1.0,0.0])
```

```
print('Done instruction: 220')sys.stdout.flush()
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-  
180.000000]),[0.000000,-90.000000,-90.000000,-  
90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-  
180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-  
21.160989],[0.0,1.0,0.0])
```

```
r.MoveL(p([-535.179980,-615.476836,-  
29.686094,90.000000,0.000000,180.000000]),[155.973180,-124.172644,-94.332579,-  
51.494776,90.000000,-24.026820],[0.0,1.0,0.0])
```

```
r.MoveL(p([-535.179980,-624.329779,-  
127.541240,90.000000,0.000000,180.000000]),[156.695979,-132.947352,-97.292270,-  
39.760378,90.000000,-23.304021],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Attach to RG2_v2""", True)r.MoveL(p([-535.179980,-615.476836,-  
29.686094,90.000000,0.000000,180.000000]),[155.973180,-124.172644,-94.332579,-  
51.494776,90.000000,-24.026820],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-906.852813,287.900000,-90.000000,0.000000,-180.000000]),[19.692662,-111.839366,-63.700709,-94.459924,90.000000,19.692662],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-906.852813,162.639108,-90.000000,0.000000,-180.000000]),[19.692662,-113.222055,-80.166806,-76.611139,90.000000,19.692662],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([655.833708,-906.852813,287.900000,-90.000000,0.000000,-180.000000]),[19.692662,-111.839366,-63.700709,-94.459924,90.000000,19.692662],[0.0,1.0,0.0])
```

```
print('Done instruction: 230')sys.stdout.flush()
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveL(p([-535.179980,-738.862340,-29.686094,90.000000,-0.000000,-180.000000]),[166.925413,-117.448180,-105.998672,-46.553148,90.000000,-13.074587],[0.0,1.0,0.0])
```

```
r.MoveL(p([-535.179980,-734.405721,-135.475556,90.000000,-0.000000,-180.000000]),[166.497973,-129.087717,-107.720801,-33.191482,90.000000,-13.502027],[0.0,1.0,0.0])
```

```
r.RunMessage(r"" Attach to RG2_v2"", True)
```

```
r.MoveL(p([-535.179980,-738.862340,-29.686094,90.000000,-0.000000,-180.000000]),[166.925413,-117.448180,-105.998672,-46.553148,90.000000,-13.074587],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-1032.968117,287.900000,-90.000000,0.000000,-180.000000]),[8.834456,-110.962945,-64.924604,-94.112452,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.MoveL(p([655.833708,-1032.968117,155.282864,-90.000000,0.000000,-180.000000]),[8.834456,-112.688251,-82.040881,-75.270869,90.000000,8.834456],[0.0,1.0,0.0])
```

```
r.RunMessage(r"" Detach from RG2_v2"", True)
```

```
r.MoveL(p([655.833708,-1032.968117,287.900000,-90.000000,0.000000,-180.000000]),[8.834456,-110.962945,-64.924604,-94.112452,90.000000,8.834456],[0.0,1.0,0.0])
```

```
print('Done instruction: 240')sys.stdout.flush()
```

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveL(p([-535.179980,-841.735050,-29.686094,90.000000,-0.000000,-180.000000]),[177.344524,-113.547783,-112.459393,-43.992824,90.000000,-2.655476],[0.0,1.0,0.0])

r.MoveL(p([-535.179980,-844.277022,-135.217132,90.000000,-0.000000,-180.000000]),[177.613194,-125.965018,-114.744725,-29.290257,90.000000,-2.386806],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"", True)

r.MoveL(p([-535.179980,-841.735050,-29.686094,90.000000,-0.000000,-180.000000]),[177.344524,-113.547783,-112.459393,-43.992824,90.000000,-2.655476],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([655.833708,-1159.173644,287.900000,-90.000000,-0.000000,-180.000000]),[-2.250261,-113.767702,-60.959707,-95.272592,90.000000,-2.250261],[0.0,1.0,0.0])

```
r.MoveL(p([655.833708,-1159.173644,153.902349,-90.000000,-0.000000,-180.000000]),[-2.250261,-115.198714,-78.599551,-76.201735,90.000000,-2.250261],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([655.833708,-1159.173644,287.900000,-90.000000,-0.000000,-180.000000]),[-2.250261,-113.767702,-60.959707,-95.272592,90.000000,-2.250261],[0.0,1.0,0.0])
```

```
print('Done instruction: 250')sys.stdout.flush()
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveL(p([-644.620589,-621.733168,-29.686094,90.000000,-0.000000,-180.000000]),[159.868984,-134.247893,-75.904078,-59.848029,90.000000,-20.131016],[0.0,1.0,0.0])
```

```
r.MoveL(p([-644.620589,-621.733168,-126.649245,90.000000,-0.000000,-180.000000]),[159.868984,-141.953950,-77.920510,-50.125540,90.000000,-20.131016],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Attach to RG2_v2""", True)
```

```
r.MoveL(p([-644.620589,-621.733168,-29.686094,90.000000,-0.000000,-180.000000]),[159.868984,-134.247893,-75.904078,-59.848029,90.000000,-20.131016],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-906.852813,287.900000,-90.000000,-0.000000,-180.000000]),[24.280089,-94.058391,-85.785941,-90.155668,90.000000,24.280089],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-906.852813,165.171094,-90.000000,-0.000000,-180.000000]),[24.280089,-96.600934,-101.199458,-72.199608,90.000000,24.280089],[0.0,1.0,0.0])
```

```
r.RunMessage(r""""Detach from RG2_v2""", True)
```

```
r.MoveL(p([530.664479,-906.852813,287.900000,-90.000000,-0.000000,-180.000000]),[24.280089,-94.058391,-85.785941,-90.155668,90.000000,24.280089],[0.0,1.0,0.0])
```

```
print('Done instruction: 260')sys.stdout.flush()
```

```
r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveL(p([-644.620589,-736.638137,-29.686094,90.000000,-0.000000,-180.000000]),[168.813520,-128.282636,-86.933505,-54.783859,90.000000,-11.186480],[0.0,1.0,0.0])
```

```
r.MoveL(p([-644.620589,-734.148046,-127.850924,90.000000,-0.000000,-180.000000]),[168.608956,-136.982818,-88.709434,-44.307748,90.000000,-11.391044],[0.0,1.0,0.0])
```

```
r.RunMessage(r"" Attach to RG2_v2"", True)
```

```
r.MoveL(p([-644.620589,-736.638137,-29.686094,90.000000,-0.000000,-180.000000]),[168.813520,-128.282636,-86.933505,-54.783859,90.000000,-11.186480],[0.0,1.0,0.0])
```

```
r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])
```

```
r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-1039.203895,287.900000,-90.000000,-0.000000,180.000000]),[10.282636,-93.128425,-86.779059,-90.092516,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.MoveL(p([530.664479,-1039.203895,163.143298,-90.000000,-0.000000,180.000000]),[10.282636,-95.785615,-102.426441,-71.787944,90.000000,10.282636],[0.0,1.0,0.0])
```

```
r.RunMessage(r"" Detach from RG2_v2"", True)
```

```
r.MoveL(p([530.664479,-1039.203895,287.900000,-90.000000,-0.000000,180.000000]),[10.282636,-93.128425,-86.779059,-90.092516,90.000000,10.282636],[0.0,1.0,0.0])
```

```
print('Done instruction: 270')sys.stdout.flush()
```

r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveJ(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveL(p([-644.620589,-844.606627,-29.686094,90.000000,-0.000000,-180.000000]),[178.043944,-124.615269,-93.544599,-51.840132,90.000000,-1.956056],[0.0,1.0,0.0])

r.MoveL(p([-644.620589,-844.606627,-128.570202,90.000000,-0.000000,-180.000000]),[178.043944,-133.877730,-95.521097,-40.601173,90.000000,-1.956056],[0.0,1.0,0.0])

r.RunMessage(r"" Attach to RG2\_v2"" , True)

r.MoveL(p([-644.620589,-844.606627,-29.686094,90.000000,-0.000000,-180.000000]),[178.043944,-124.615269,-93.544599,-51.840132,90.000000,-1.956056],[0.0,1.0,0.0])

r.MoveL(p([-491.950000,-666.632053,287.900000,90.000000,-0.000000,-180.000000]),[158.839011,-101.850119,-76.826732,-91.323149,90.000000,-21.160989],[0.0,1.0,0.0])

r.MoveJ(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-180.000000]),[0.000000,-90.000000,-90.000000,-90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.MoveL(p([530.664479,-1155.884259,287.900000,-90.000000,-0.000000,180.000000]),[-2.424082,-96.056313,-83.597199,-90.346488,90.000000,-2.424082],[0.0,1.0,0.0])

```

    r.MoveL(p([530.664479,-1155.884259,160.125815,-90.000000,-
0.000000,180.000000]),[-2.424082,-98.737977,-99.512466,-71.749557,90.000000,-
2.424082],[0.0,1.0,0.0])

r.RunMessage(r""""Detach from RG2_v2""", True) r.MoveL(p([530.664479,-
1155.884259,287.900000,-90.000000,-
0.000000,180.000000]),[-2.424082,-96.056313,-83.597199,-90.346488,90.000000,-
2.424082],[0.0,1.0,0.0])

print('Done instruction: 280')sys.stdout.flush()

    r.MoveL(p([491.950000,-1133.300000,287.900000,-90.000000,-0.000000,-
180.000000]),[0.000000,-90.000000,-90.000000,-
90.000000,90.000000,0.000000],[0.0,1.0,0.0])

r.ProgFinish(r""""Ипора_3""")

r.ProgSave(r""""C:/Users/Work/Documents/RoboDK/""",r""""_ """,False,r""""C:/RoboDK/
Other/VSCodium/VSCodium.exe""")

if name == "main":make_program()

```

## ВИСНОВОК ЗА РОЗДІЛОМ 6

У даному розділі було розглянуто основні функції для програмування РТК. Розробив програму реалізації алгоритму логічного керування на одній з мов програмування відповідно до міжнародного стандарту . Провів тестування програми логічного керування із використанням засобів емуляції контролера.

Склад технічних засобів та структура контролерно-комп'ютерної мережі повністю відповідають структурі, наведеній у розділі 5. Підпрограма автоматичного пуску ізупинки добре прокоментована і передбачає реакцію на виникнення позаштатних ситуацій в системі керування

При тестуванні керуючі дії в цій підпрограмі не протирічать алгоритмам логічного керування, наведеним в розділі 4.

## РОЗДІЛ 7 ВІЗУАЛІЗАЦІЯ РТК В ПРОГРАМНОМУ СЕРЕДОВИЩІ ROBO DKТА СТВОРЕННЯ СИМУЛЯЦІЇ ЦИФРОВОГО ДВІЙНИКА

### 7.1 Створення цифрового двійника

Для початку реалізації симуляції РТК та створення цифрового двійника ,  
нашpotрiбно

1. Створення 3D об'єктів ділянки в програмному середовищі FUSION 360
  - Скетч планування РТК
  - Створення 3D-моделей та їх розташування
  - Креслення РТК (Drawing)
2. Створення симуляції у Robo DK
  - Розташування створених об'єктів в програмному середовищі RoboDK
  - Створення візуалізації роботи ділянки

### 1. СТВОРЕННЯ 3D ОБ'ЄКТІВ ДІЛЯНКИ В ПРОГРАМНОМУ СЕРЕДОВИЩІ FUSION 360

Fusion 360 — це хмарний інструмент CAD/CAM для спільної розробки продуктів. Fusion 360 дає змогу досліджувати й повторювати ідеї продукту та співпрацювати з командою розробників розподіленого продукту. Fusion 360 поєднує моделювання органічних форм, механічне проектування та виготовлення в одному комплексному пакеті.



Рис. 7.1 - Головне меню програми Fusion 360

### 7.1.1 Скетч планування РТК

Програма Fusion 360 створена на основі AutoCAD , отримав навички роботи нанавчальному курсі «САПР».

У 3D-моделюванні скетч (або скетч-моделювання) відноситься до процесу створення грубих, але швидких ескізів або наближених моделей об'єктів у тривимірному просторі. Скетчі допомагають швидко візуалізувати форму, структуру та основні деталі об'єкта перед переходом до більш деталізованого моделювання.

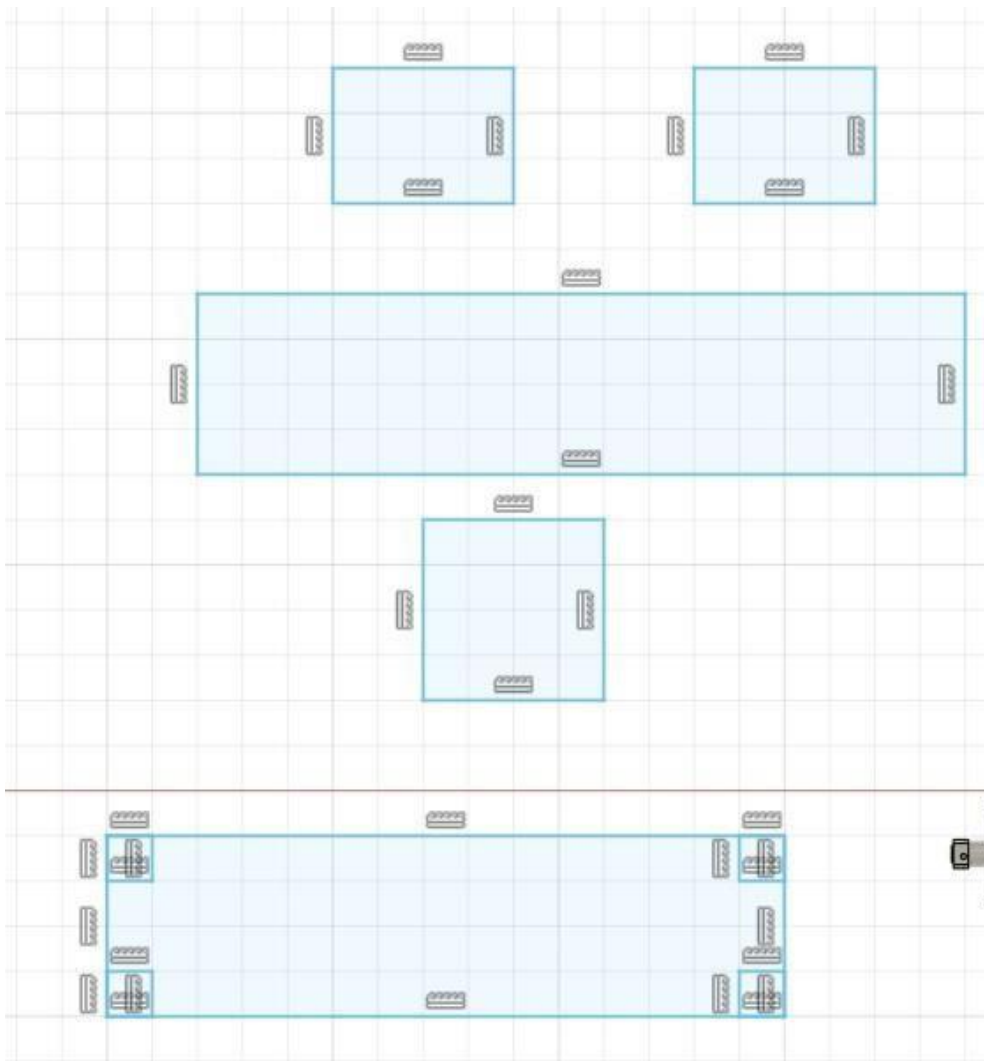


Рис. 7.2 - Створення скетч ділянки

## 7.1.2 Створення 3-D моделей та розташування

Після скетча нам потрібно це витягнути та для цього нам допомагає функція Extrude. У Fusion 360 дозволяє видавлювати (продовжувати у висоту) об'єкти, створюючи 3D-моделі з 2D-фігур або площин.

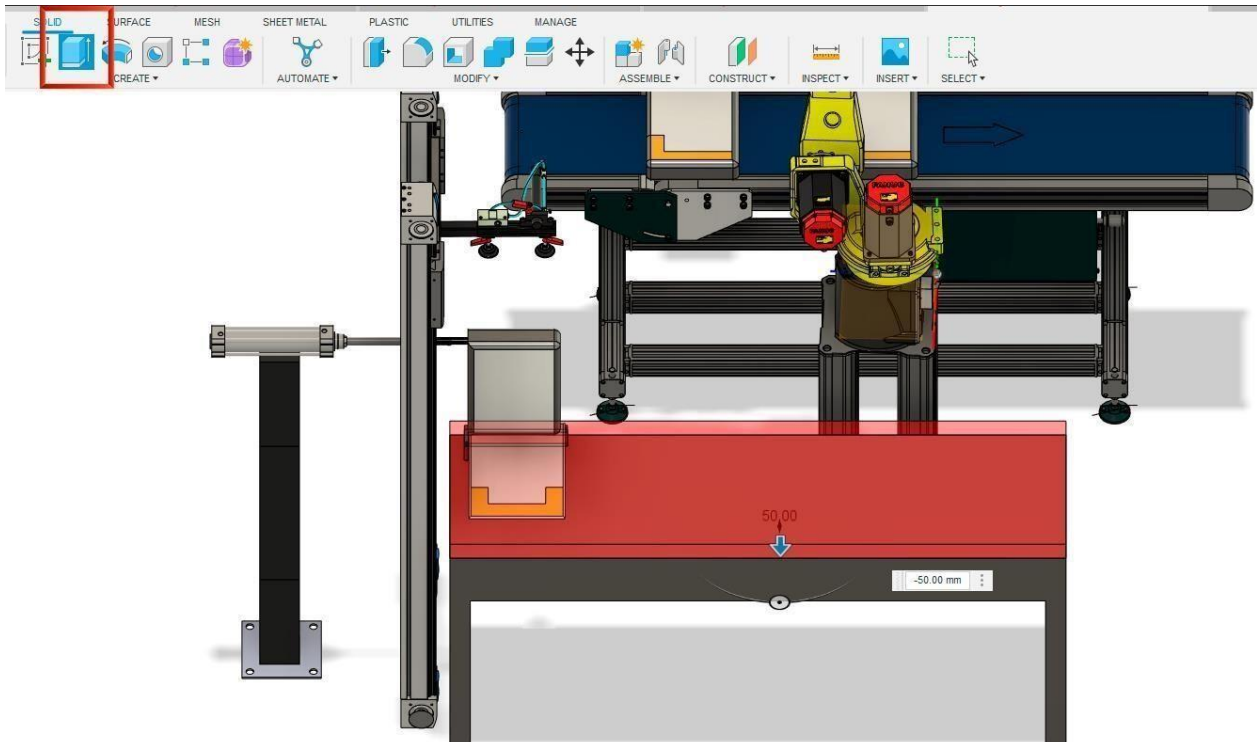


Рис. 7.3 - функція Extrude

Ось кілька кроків для використання функції Extrude у Fusion 360, що я використовував :

- Виберіть об'єкт: Виберіть 2D-фігуру або площину, яку ви бажаєте видавити. Це може бути будь-яка замкнута фігура, така як круг, прямокутник або складніші контури.

- Застосуйте команду Extrude: У меню "Create" (Створити) виберіть опцію "Extrude". З'явиться панель управління Extrude, де ви зможете встановити параметри для видавлювання.

Встановлення параметрів Extrude: У панелі управління Extrude ви можете встановити наступні параметри:

- Direction (Напрямок): Виберіть напрямок видавлювання, це може бути вгору, вниз або за будь-яким іншим визначеним напрямком.
- Distance (Відстань): Вкажіть відстань видавлювання, тобто висоту, на яку потрібно продовжити об'єкт.
- Operation (Операція): Виберіть тип операції видавлювання. Це може бути Solid (тверде тіло), Join (з'єднання), Cut (вирізання) або Intersect (перетин).
- Taper Angle (Кут нахилу): Встановіть кут нахилу, якщо бажаєте надати видавлюванню нахил або похилу форму.

Я зробив так з кожною моделлю та отримав ось таку 3D модель РТК:

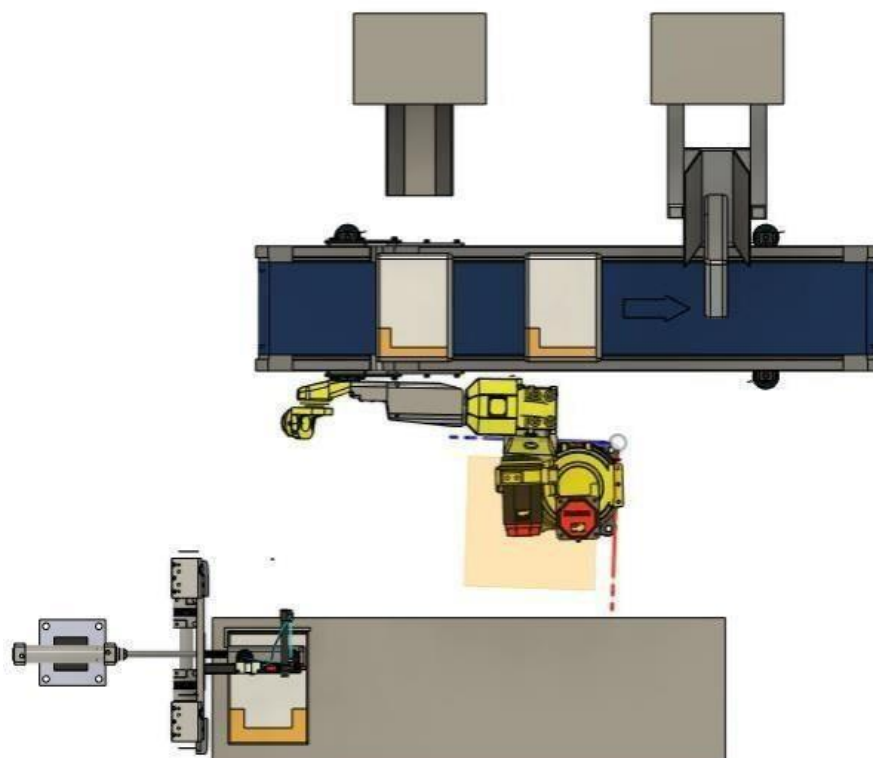


Рис. 7.4 - 3Д РТК

### 7.1.3 Креслення РТК

Drawing у програмі Fusion 360 є інструментом для створення технічних креслень та креслених документів на основі ваших 3D-моделей. Drawing в Fusion 360 надає ряд можливостей, що сприяють створенню докладних інженерних креслень для виробництва або документації проекту.

Основні переваги Drawing у Fusion 360 включають:

- Створення вимірювальних креслень: Drawing дозволяє додавати розміри, анотації та інші виміри до вашої 3D-моделі. Ви можете вказати розміри деталей, відстані між елементами, радіуси, кути та інші характеристики.
- Технічні анотації: Ви можете додавати текстові анотації, специфікації, таблиці, позначення матеріалів та іншу інженерну документацію до своїх креслень. Це допомагає зрозуміти, якісно описати і документувати ваші проекти.
- Додавання символів та стандартних елементів: Fusion 360 має багатий набір символів, штампів, логотипів, геометричних форм та інших стандартних елементів, які можна легко додати до креслення. Це допомагає стандартизувати ваші креслення та забезпечити консистентність у документації.

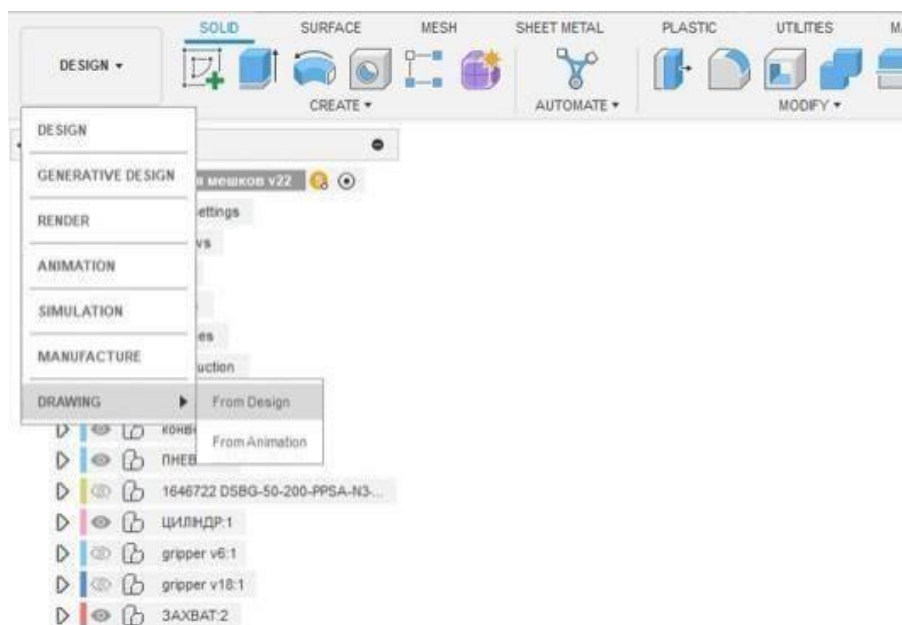


Рис. 7.5 - функція Drawing

Креслення , де зображено відстань між об'єктам на рис

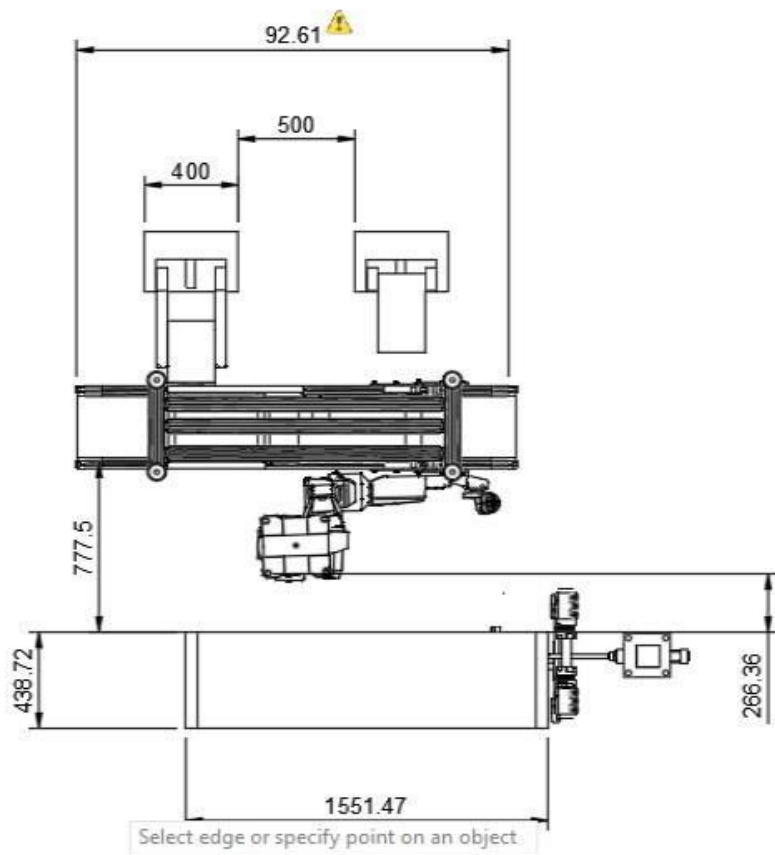


Рис. 7.6 - Креслення

## 7.2 Створення симуляції у ROBO DK

RoboDK - це програмне забезпечення для симуляції та оффлайн програмування роботів. Воно використовується для планування, моделювання, оптимізації та симуляції робототехнічних процесів.

Основні переваги використання RoboDK включають:

**Симуляція та верифікація:** RoboDK дозволяє створювати віртуальну модель робота та робочого середовища. Можливість перевірити та валідувати програми робота безпосередньо на комп'ютері, уникнувши ризику пошкодження обладнання або простої у виробництві. Це допомагає знизити часта витрати на впровадження.

**Оффлайн-програмування:** За допомогою RoboDK можливість створювати програми для роботів безпосередньо на комп'ютері, незалежно від роботичної системи. Ви можете програмувати робота за допомогою графічного інтерфейсу або писати скрипти на мовах програмування, таких як Python. Це дозволяє прискорити процес програмування та знизити час налагодження.

**Оптимізація шляхів руху:** RoboDK має вбудовані алгоритми для оптимізації шляхів руху робота. Ви можете автоматично згладити траєкторії, уникнути перешкод та мінімізувати час переміщення робота. Це допомагає забезпечити ефективніше та швидше виконання завдань робота.

**Інтеграція з CAD/CAM:** RoboDK може інтегруватися з різними системами CAD/CAM, такими як SolidWorks, AutoCAD, Mastercam тощо. Це дозволяє імпортувати моделі та шляхи руху з цих систем безпосередньо до RoboDK для подальшої праці.

## 7.2.1 Розташування створених об'єктів у ROBO DK



Рис. 7.7 - Головні функції RoboDK

Опис головних функцій (рис 7.7.):

- Створення структури (фрейм) – визначає положення об'єкта відносно робота (положення та орієнтація). Рекомендується навчати цілі відносно еталону, яку ви можете визначити за допомогою свого робота.
- Вільний рух (MoveL) – переміщення роботу до точки з оптимальною траєкторією та швидкістю, не використовується при великій кількості об'єктів з якими робот може зіштовхнутись.
- Лінійний рух (MoveL) – переміщення роботу по лінійній траєкторії, використовується для точних рухів. Наприклад, захват деталі.
- Рух по радіусу (MoveR) – використовується для опису рухів в вигляді радіусу. Наприклад, на півколо та коло.

Додавання об'єктів в середовище RoboDK:

- Додавання об'єктів з комп'ютера – виберіть команду «Завантаження об'єктів з комп'ютера» та вкажіть шлях до експортованого файлу.
- Додавання об'єктів з онлайн бібліотеки (рис.2.1.) – виберіть команду онлайн бібліотеку, виберіть за назвою «Universal Robots» та додайте модель роботу «UR3e» натиснувши «Download».

Fanuc M-10iA/6L		Fanuc M-10iA/7L		Fanuc M-10iA/8L		Fanuc M-10iA/10S		Fanuc M-10iA		Fanuc M-10iD/10L	
Brand	Fanuc	Brand	Fanuc	Brand	Fanuc	Brand	Fanuc	Brand	Fanuc	Brand	Fanuc
Model	M-10iA/6L	Model	M-10iA/7L	Model	M-10iA/8L	Model	M-10iA/10S	Model	M-10iA	Model	M-10iD/10L
Axes	6	Axes	6	Axes	6	Axes	6	Axes	6	Axes	6
Reach	1633 mm	Reach	1633 mm	Reach	2028 mm	Reach	1098 mm	Reach	1420 mm	Reach	1636 mm
Payload	6 kg	Payload	7 kg	Payload	8 kg	Payload	10 kg	Payload	10 kg	Payload	10 kg
Weight	135 kg	Weight	135 kg	Weight	150 kg	Weight	130 kg	Weight	130 kg	Weight	150 kg
Repeatability	0.080 mm	Repeatability	0.080 mm	Repeatability	0.040 mm	Repeatability	0.080 mm	Repeatability	0.080 mm	Repeatability	0.030 mm

Рис. 7.8 - Додавання роботу з онлайн бібліотеки

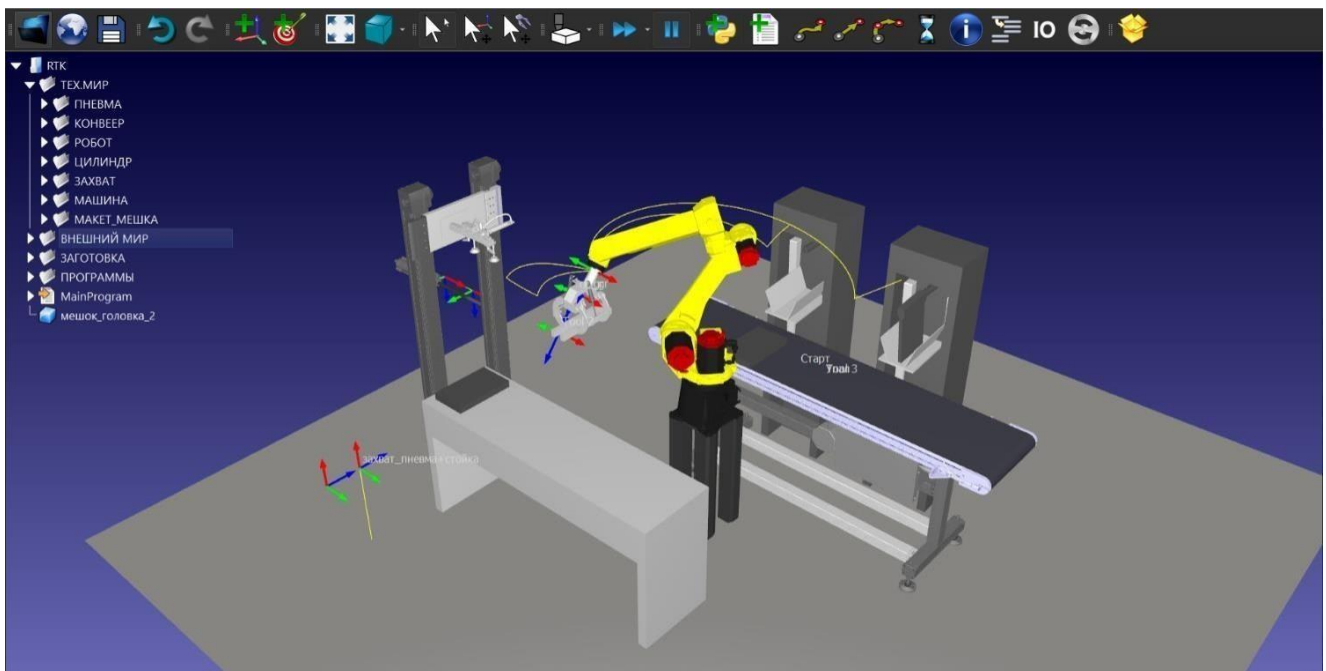


Рис. 7.9 - Додавання 3D моделей

## 7.2.2 Створення візуалізації роботи ділянки

Після розміщення об'єктів та налаштування інструменту на фланці роботи, створив програму, механізми, програмування для симуляції.

RoboDK може автоматично генерувати шляхи руху для робота з урахуванням фізичних обмежень, таких як обмеження швидкості, обмеження поворотів і мінімізація перекриття між сегментами шляху. Це допомагає оптимізувати рухи робота та отримати найкращий результат.

Для створення, я спочатку :

1. Створив новий фрейм вибравши команду «Add Frame».
2. Обрав інструмент роботи до потрібного місця шляхом налаштування координат

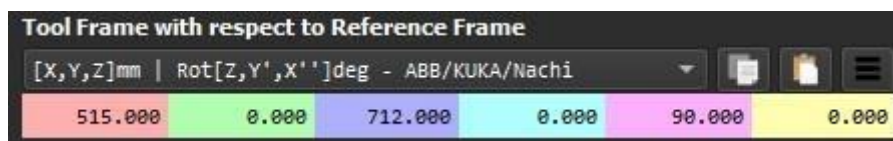


Рис. 7.10 - Додавання роботу з онлайн бібліотеки

3. Задав точки руху робота: HP – початкова позиція . PRE-TAKE – позиція перед тим як робот візьме заготовку. TAKE – позиція для того , щоб взяти заготовку. TRUBA

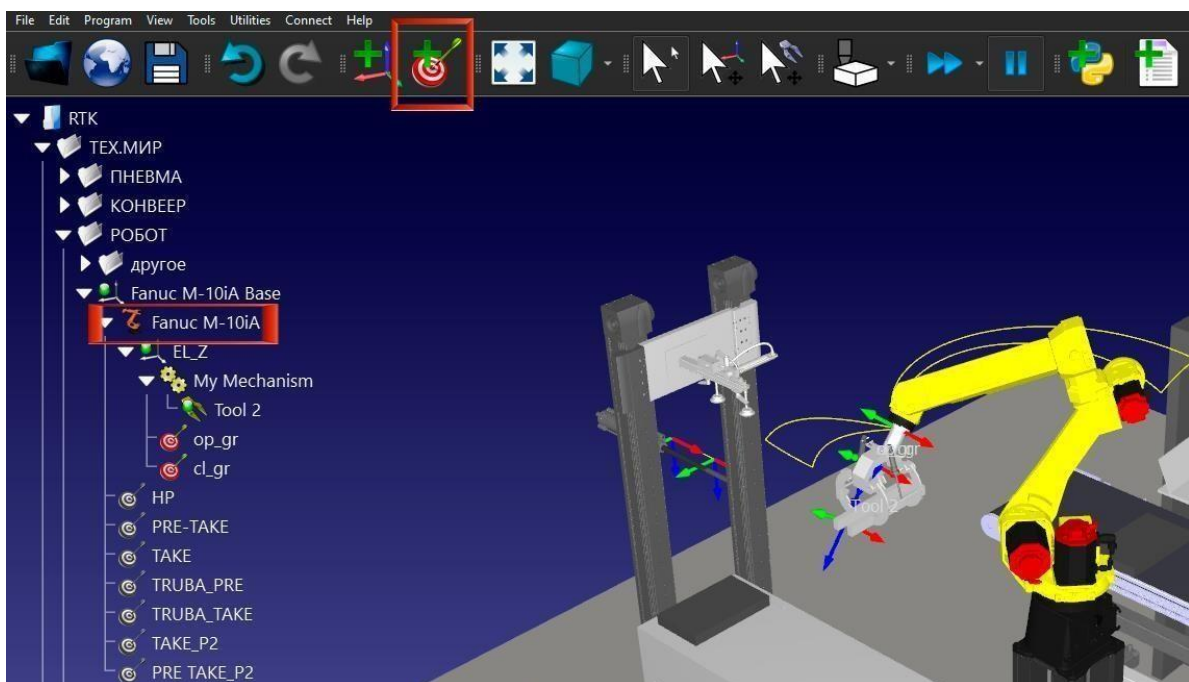


Рис. 7.11 - Додавання точки

PRE - позиція перед тим як робот закріпе заготовку до машини фасувальної.  
TRUBA TAKE – позиція для того , щоб закріпити заготовку до машини фасувальної.

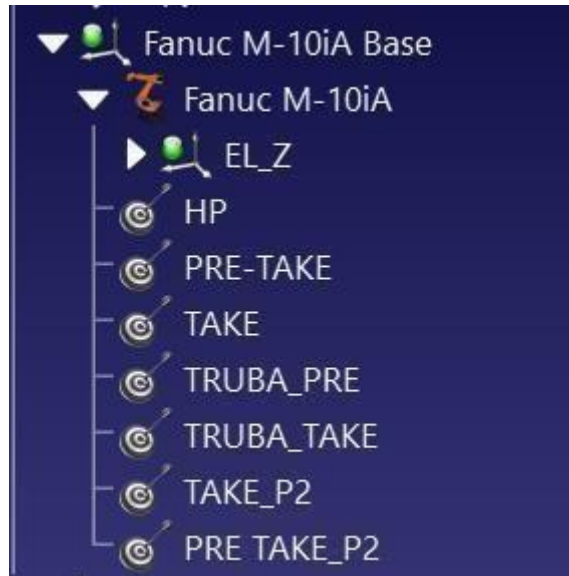


Рис. 7.12 - Додавання роботи з онлайн бібліотеки

4. Створив механізм, а саме за допомогою функції "Add" (Додати) у панелі інструментів RoboDK додав необхідні компоненти механізму, такі як базову платформу, ланки, суглоби, ефектори тощо.
5. Зв'язав компоненти, завдяки функції "Link". Встановив відповідні параметри, такі як тип суглобу, обмеження руху та кутові обмеження.
6. Налаштував кінематику, а саме обрав кінематичний тип механізму, серійну ланку та механізм.
7. Налаштував параметри кінематики, такі як кількість ступенів свободи, довжини ланок, перетини суглобів тощо.

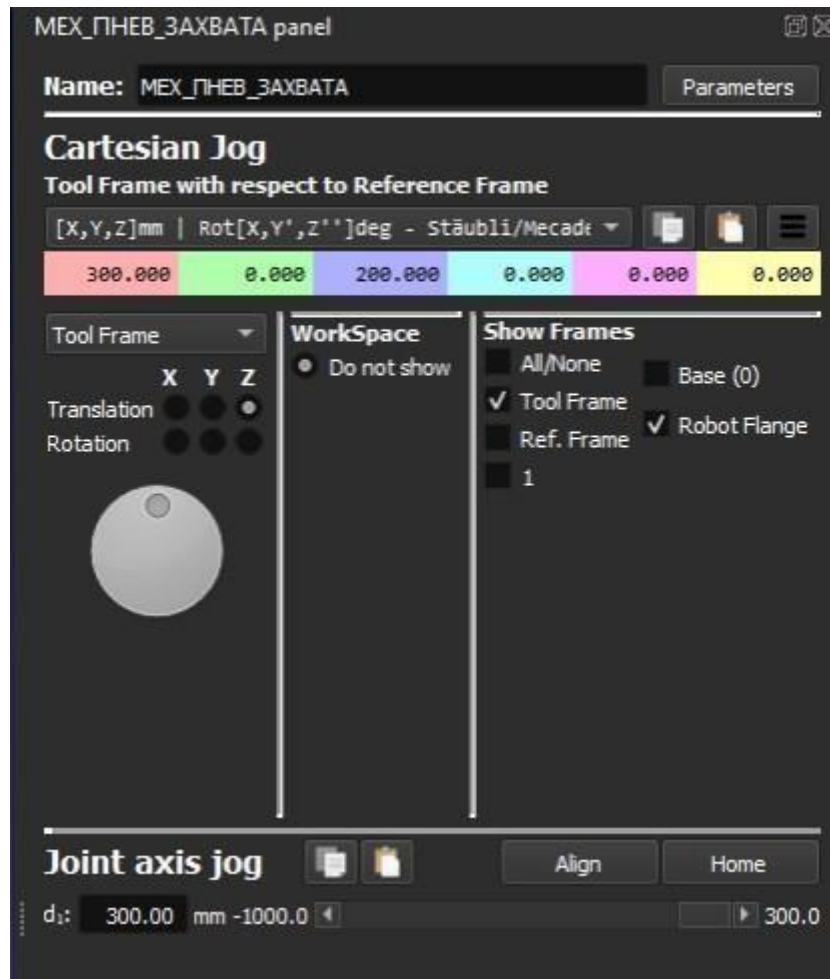


Рис. 7.13 - Вікно налаштування механізму.

Оффлайн-програмування: RoboDK дозволяє оффлайн-програмування роботів, що означає, що є можливість створювати та редагувати програми для робота безпосередньо на комп'ютері, незалежно від роботичної системи.

Для цього потрібно використовувати графічний інтерфейс для створення послідовності рухів робота, задавати точки, додавати логічні умови та цикли. Оффлайн-програмування дозволяє прискорити процес програмування та зменшити час налагодження.

Коли зробив усі механізми для моделей, а саме для циліндру, пневмозахвату, електрозахвату, фасувальної машини, я почав програмувати РТК:

Я розділив етапи на під-програмування РТК на такі моменти , як: програмування пневмо-захвату , програмування циліндру, програмування робота та його захвату, програмування фасувальної машини.

## Програмування пневмо-захвату

Піднімається проміжний мішок , завдяки пневматичного захвату, який використовує стиснене повітря для захоплення та утримання об'єкту під час переміщення на виробничий лінії.

Алгоритм дій:

- Пневма-захват опускається
- Бере заготовку
- Піднімає
- Очікує на циліндр
- Піднімається, щоб робот міг отримати

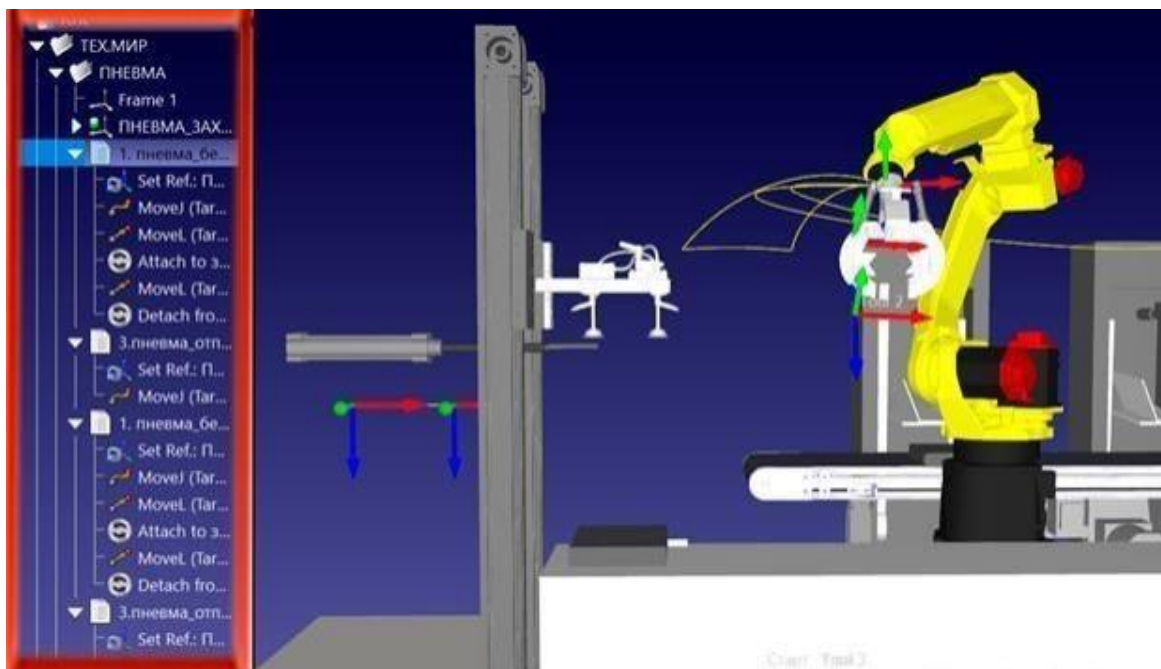


Рис. 7.15 - Програмування пневмо-захвату

## Програмування циліндру

Алгоритм дій:

- Виїзжає
- Заїджає

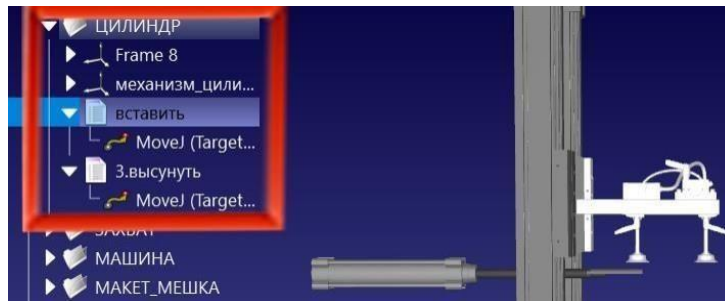


Рис. 7.14 - Програмування циліндру

### Програмування робота та його електро-захвату.

Задав точки руху робота: HP – початкова позиція . PRE-TAKE – позиція перед тим як робот візьме заготовку. TAKE – позиція для того , щоб взяти заготовку. TRUBAPRE - позиція перед тим як робот закріпе заготовку до машини фасувальної. TRUBA TAKE – позиція для того , щоб закріпити заготовку до машини фасувальної.

### Програмування фасувальної машини

Алгоритм дій:

- Пневма-захват опускається
- Бере заготовку

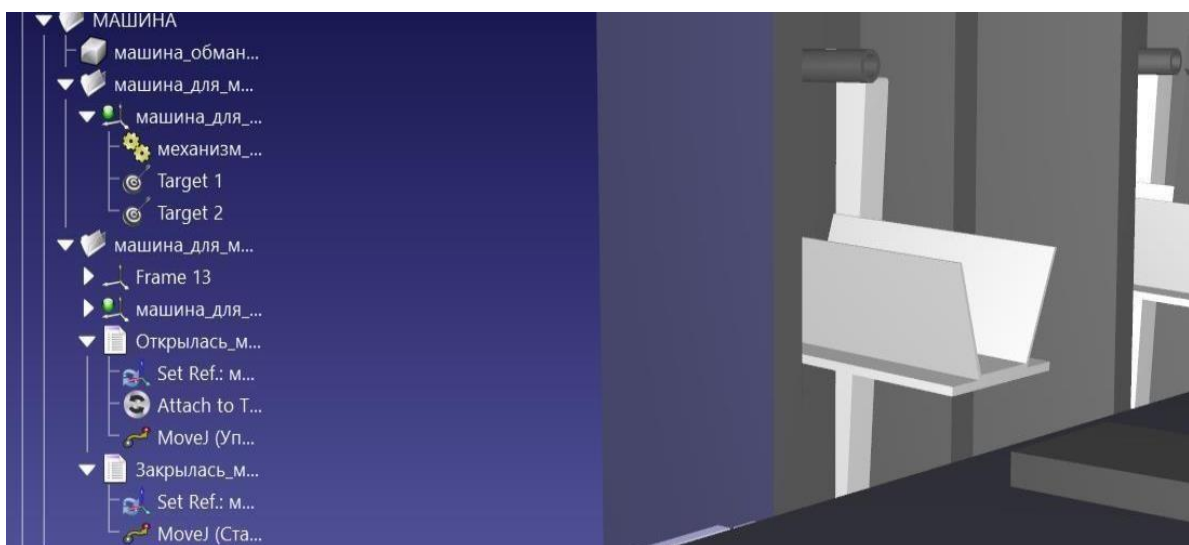


Рис. 7.16 - Програмування фасувальної машини

Перед створенням основної потрібно створити підпрограму для закріплення об'єктів. Вона потрібна для того щоб повернути об'єкти до моменту фіксації, якщо об'єкти

були здвинуті с потрібного місця, наприклад переміщенням кубика, але немає можливості його повернути на місце, окрім ручного налаштування.

Створення підпрограми:

1. Вибрав команду «Створити пусту програму».
2. Клацнув на неї лівої кнопкою миші та виберіть пункт «Add instruction».
3. В випадаючому меню виберіть пункт «Simulation Event» та виберіть підпункт «Set Object Relative».
4. Вибрав всі об'єкти, фрейми.Створення основної програми:
  1. Вибрав створену пусту програму «Main program»
  2. Вибрав команду «Call Program», в випадаючому меню та розмістив по ланцюгу дій підпрограми .
  3. Запустив, перевірів.

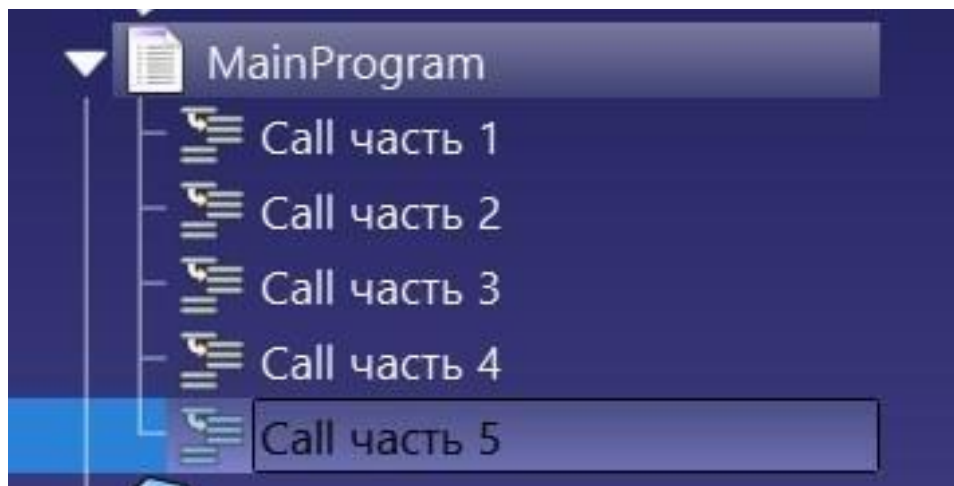


Рис. 7.17 - Програмування основної програми

## **ВИСНОВОК ЗА РОЗДІЛОМ 7**

У цьому розділі розглянули основні функції програми RoboDK та зробили 3D моделі у Fusion 360. Програма дозволяє створювати віртуальні моделі роботів, симулювати їх рухи та виконувати оффлайн-програмування. Створили складні механізми, зв'язували компоненти, налаштовували кінематику та встановлювали параметри робота. Симуляція дозволила перевірити, як буде працювати РТК. Це допомогло знизити ризик пошкодження обладнання та зменшити час налагодження.

RoboDK є потужним інструментом для інженерів та програмістів, що працюють з роботами, і допомагає автоматизувати процеси, збільшувати ефективність та знижувати витрати..

## РОЗДІЛ 8 РОЗРОБКА ФРАГМЕНТІВ ДОКУМЕНТАЦІЇ ТЕХНІЧНОГО ЗАБЕСПЕЧЕННЯ САК, ВКЛЮЧАЮЧИ ПИТАННЯ ОХОРОНИ ПРАЦІ

8.1. Коротка характеристика технологічного об'єкту керування, склад документації, що розроблена, та опис основних технічних рішень із побудови САК.

За результатами вибору комплексу технічних засобів й відповідно до складеної технічної структури системи керування розроблено комплект її проектної документації. Комплект включає наступні документи:

### **Основні технічні рішення**

Комплект графічної і текстової документації. Графічні:

- 1 – схему автоматизації та технологічну;
- 2 – принципову електричну схему живлення;
- 3 – принципову електричну схему підключення до блоку дискретних входів/виходів ПЛК;
- 4 – принципову електричну схему підключення до блоку аналогових входів/виходів ПЛК.

Текстові:

- попередній перелік обладнання до всіх графічних документів;
- пояснювальна записка до проекту.

8.2. Розробка схеми автоматизації та специфікацій на технічні засоби системи

Схема автоматизації технологічного процесу є основним документом, що визначає функціональну структуру, обсяг автоматизації й що роз'ясняє певні процеси, що протікають у функціональних ланцюгах Автоматизація технологічних процесів. Зображення умовні приладів і засобів автоматизації в схемах".

Цемент займає перше місце серед усіх в'язких речовин за обсягами виробництва і споживання. Його одержують випалом до спікання

(максимальна температура випалу 1400–1450 °C) точно дозованої суміші глини та карбонату кальцію з наступним тонким подрібнюванням отриманої спеченої маси (клінкеру).

Виробництво цементу складається з двох самостійних процесів: а) виробництва напівфабрикату-клінкеру, що включає підготовку сировинної суміші та її випал, і б) помелу клінкеру з домішками, складування й упакування цементу. Схему виробництва цементу мокрим способом показано на **рис.1**. Подріблений у дробарках 2 вапняк подається в трубні сировинні млини, де відбувається загальне (тонке) здрібнювання вапняку і глини, що подається у вигляді рідкої маси глини-бовтанки. Із млинів сировинний шлак подається в залізобетонні резервуари з мішалками або у шламбасейни 4 та 5 з пневматичним перемішуванням суміші. У шламових басейнах зберігається і коригується сировинна суміш, тобто змішуються шлами різного хімічного складу для отримання необхідного співвідношення компонентів.

Сировинна суміш обпікається в барабанах обертових печах 6, які опалюються вугільним пилом, газоподібним паливом або мазутом, причому паливо спалюється всередині печі. Сировинний шлам, подаваний у піч з горизонтального шламбасейну 5, рухається протитечією до гарячих топкових газів. При цьому послідовно відбуваються процеси випаровування води, дегідратації мінералів, дисоціації вапняку та реакції між утворюваними оксидом CaO і складовими глиняного компонента SiO<sub>2</sub>, Al<sub>2</sub>O<sub>3</sub>, Fe<sub>2</sub>O<sub>3</sub>.

У зоні спікання за температури 1450 °C остаточно формується цементний клінкер. Одержаний клінкер охолоджується холодним повітрям у колосникових холодильниках до температури 50–60 °C.

У цих холодильниках, розташованих під печами, зерна клінкеру рівномірним шаром розподіляються на колосникових ґратах, під які подається повітря, що далі спрямовується в піч для горіння палива.

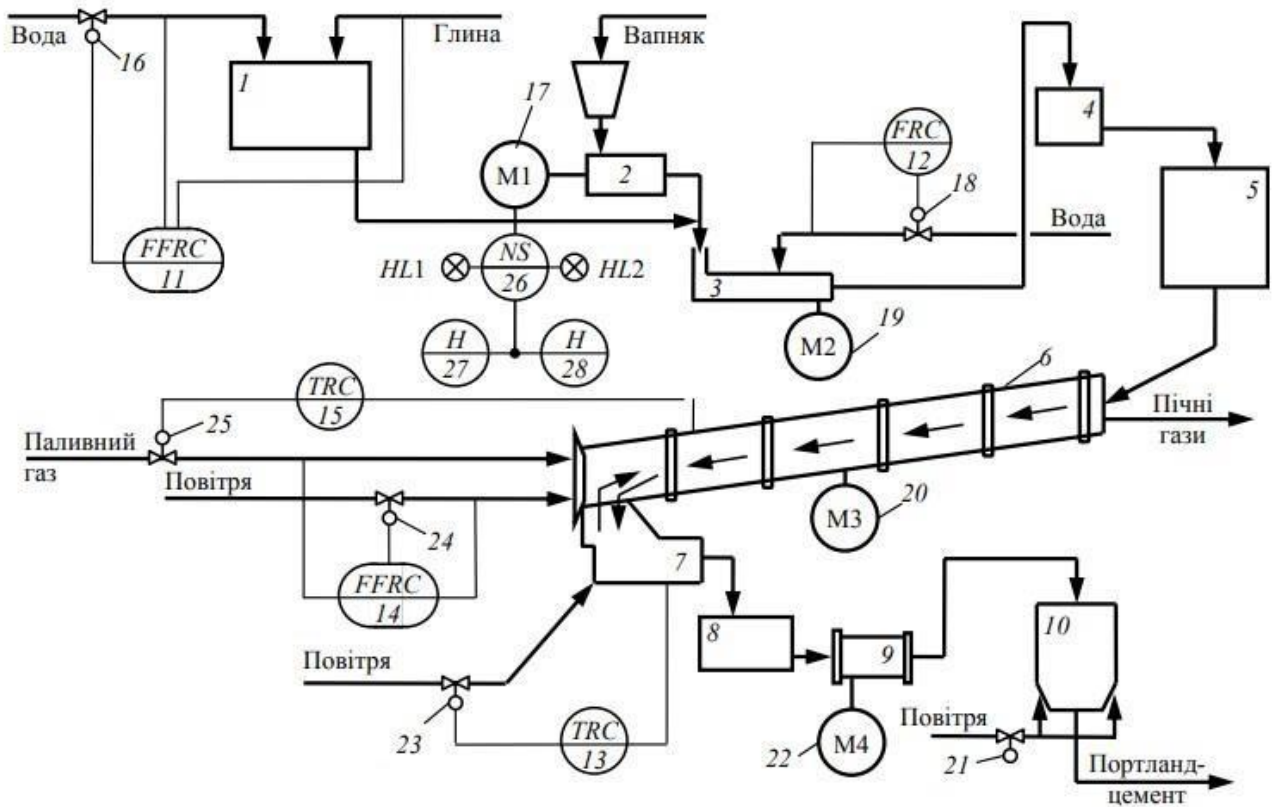


Рис.8.1 - Технологічна схема та схема автоматизації виробництва цементу

Пояснення схеми автоматизації виробництва:

- 1 – глинобовтанка;
- 2 – молоткова дробарка;
- 3 – сировинний млин;
- 4 – коригувальні шлам-басейни;
- 5 – горизонтальні шлам-басейни;
- 6 – барабанна обертова піч;
- 7 – колосниковий холодильник
- 8 – склад;
- 9 – кульовий цементний млин;
- 10 – цементні силоси;
- 11, 14 – регулятори співвідношення;
- 12 – регулятор витрати;
- 13, 15 – регулятори температури;
- 16, 18, 21, 23, 24, 25 – регулювальні клапани;
- 17, 19, 20, 22 – електроприводи;
- 26 – магнітний пускач;

27, 28 – кнопочві пости керування

Із холодильників клінкер подається на склад, де якийсь час вилежується для гашення (гідратації) вільного вапна і зв'язування її гідравлічними домішками. Вилежаний клінкер разом з гідравлічними або інертними домішками та домішкою гіпсу, що регулює терміни схоплювання, розмелюють у кульових цементних млинах 9. Цемент зберігається в залізобетонних силосах 10 з аерувальними днищами, через які під час розвантаження силосу нагнітається повітря, що розпушує цемент. Цемент споживачеві постачають навалом в авто- чи залізничних цементовозах або у багатошарових паперових мішках. Для надання цементам певних властивостей, а також для їх здешевлення застосовують різні домішки до в'язучих речовин: гідравлічні, що містять активний кремнезем, який підвищує водостійкість цементів і сприяє їх твердненню під водою; пластифікувальні поверхнево-активні речовини, що підвищують еластичність і зв'язність цементного тіста; інертні (пісок, вапняк, доломит), кислотостійкі (андезит, граніт), а також такі, що регулюють терміни схоплювання (гіпс).

Цемент виготовляють з порівняно доступної і дешевої сировини; вироби з нього мають високі механічну міцність і морозостійкість. Виробництво цементу мокрим способом є складним процесом і тому треба постійно контролювати та регулювати параметри технологічного процесу. Автоматизація процесу виробництва цементу мокрим способом.

Приготування водоглиняної суспензії заданої концентрації забезпечується регулятором співвідношення 11 через зміну витрати води (відповідно до зміни подавання глини) за допомогою регулювального клапана 16.

Витрата води у трубний сировинний млин 3, де відбувається спільне тонке розмелювання подрібненого вапняку та глини, регулюється за допомогою регулятора витрати 12 і регулювального клапана 18 на трубопроводі подавання води. Температура в обертовій барабанній печі 6 регулюється через зміну подавання палива за допомогою регулятора 15 і

регулювального клапана 25. Регулятор співвідношення 14 підтримує на заданому рівні співвідношення паливо – повітря через зміну подавання в піч повітря клапаном 24.

Стабілізація температури цементного клінкеру на виході колосникового холодильника 7 у межах 60–70 о С забезпечується регулятором 13 зміною ступеня відкриття регулювального клапана 23 на трубопроводі подавання охолоджувального повітря у холодильник. Електродвигун М1, що приводить у дію молоткову дробарку 2, дистанційно керується за допомогою місцевого магнітного пускача 26 та кнопкових постів керування 27 (увімкнення двигуна) і 28 (вимкнення), розміщених на щиті (у разі потреби можна дублювати кнопковими постами керування місцевого розташування). Сигнальні лампочки HL1 і HL2 показують стан електродвигуна М1.

Аналогічно здійснюється керування (на схемі не показано) роботою 191 тою електроприводів сировинного млина електродвигуна М2), барабанної обертової печі (електродвигуна М3) та кульового цементного млина (електродвигуна М4).

ЕЛЕКТРОАПАРАТИ										
SB1, SB2	Витрата	Насоси Н-1, Н-2	80 м <sup>3</sup> /год	Щит керування	Пост керування кнопковий	ПКУ-123-11		Трест «Променергоавтоматика», м. Київ	2 од.	0,27
МП1, МП2	–	–	–	Місцевий	Пускач магнітний безконтактний реверсивний	ПБР-3А		ВО «Електроприбор», м. Чебоксари	2 од.	3,5
МП3, МП4	–	–	–	Місцевий	Пускач магнітний безконтактний нереверсивний з реле IP-20	ПМ 12-01025		ВО «Електроприбор», м. Чебоксари	2 од.	3,2
HL1, HL2	–	–	–	–	Лампа сигнальна із зеленим індикатором	ЛС 47-1		«ІЕК Україна», м. Київ	2 од.	0,12
HL3, HL4	–	–	–	–	Лампа сигнальна з червоним індикатором	ЛС 47-2		«ІЕК Україна», м. Київ	2 од.	0,12
КАБЕЛІ ТА ПРОВІДНИКИ										
	–	–	–	–	Кабель 4 × 1,5	КВВШ		Завод «Південкабель», м. Харків	450 м	0,08
МОНТАЖНІ ВИРОБИ										
	–	–	–	–	Коробка з'єднувальна вибухозахисного виконання	КСВ-1-1		Завод «Південкабель», м. Харків	4 од.	2,2
МАТЕРІАЛИ										
	–	–	–	–	Труби захисні для електропроводок (з муфтами БНМ 100)	БНТ 100		Південнотрубний завод, м. Нікополь	3 комплекти	0,4

Рис.8.1 - таблиця

### 8.3 Розробка принципової електричної схеми живлення

Підключення до електричної мережі. Силовий кабель від блоку керування оснащений стандартним штепселем ІЕС на кінці. підключити використовуваний у вашій країні штепсель або кабель до штепселя стандарту ІЕС.

Щоб під'єднати живлення до робота, необхідно під'єднати блок керування до джерела живлення. Живлення через стандартний штепсель ІЕС С20, розташований у нижній частині блоку керування, з використанням відповідного кабелю ІЕС С19.

Живлення від мережі оснащено наступними елементами:

- Заземлювальне з'єднання
- Головний запобіжник
- Пристрій захисного вимкнення

Рис.8.2 – Живлення контролеру

Для зручного блокування та блокування живлення на період обслуговування рекомендується встановити основний вимикач живлення всього обладнання робототехнічної системи.

### Рис.8.3 - Електричні вимоги

Розробка принципової електричної схеми живлення передбачає створення докладного плану системи живлення, включаючи джерела живлення, електричні мережі та захисні пристрої

### Рис.8.4 - Загальна схема підключення

#### 8.4 Розробка принципової електричної схеми підключення до блоку аналогових та дискретних входів та виходів ПЛК

Серія контролерів **R-30iB** від компанії FANUC має різні модулі вводу/виводу (I/O), які забезпечують можливості підключення та керування зовнішніми пристроями. Деякі з популярних модулів I/O, доступних для контролерів серії R-30iB, включають:

**Digital Input Module (DI):** Цей модуль дозволяє підключати цифрові сигнали вхідних пристроїв, таких як датчики або перемикачі. Він забезпечує зчитування стану цифрових сигналів для подальшого використання в програмах контролера.

**Digital Output Module (DO):** Цей модуль дозволяє керувати цифровими вихідними пристроями, такими як клапани, реле або індикатори. Він дозволяє контролювати стан цифрових виходів для передачі сигналів у зовнішні пристрої.

**Analog Input Module (AI):** Цей модуль дозволяє підключати аналогові вхідні пристрої, такі як датчики тиску або температури. Він забезпечує зчитування аналогових сигналів з цих пристроїв для подальшої обробки та використання в програмах контролера.

**Analog Output Module (AO):** Цей модуль дозволяє керувати аналоговими вихідними пристроями, такими як приводи або регулятори. Він дозволяє встановлювати аналогові значення на виходах для управління зовнішніми пристроями.

Виходи та виходи всередині блоку керування можуть бути використані для підключення обладнання різного призначення, у тому числі пневматичні реле, ПЛК і кнопка аварійної зупинки, а саме аварійної

зупинки. На рис. показано розташування груп електричного інтерфейсу всередині блоку управління.

Рис.8.5 - Описання колірної схеми

Описання колірної схеми

- Жовті з червоним текстом - Спеціальні сигнали безпеки
- Жовті з чорним текстом - Налаштовувані в якості безпечних
- Сірі з чорним текстом - Цифрові входи/виходи загального призначення
- Зелений з чорним текстом - Аналогові входи/виходи загального призначення

Підключення кнопок аварійної зупинки

У більшості випадків використання робота потрібно під'єднати одну або декілька додаткових кнопок аварійної зупинки. На рис демонструю як можна під'єднати одну або кілька кнопок аварійної зупинки.

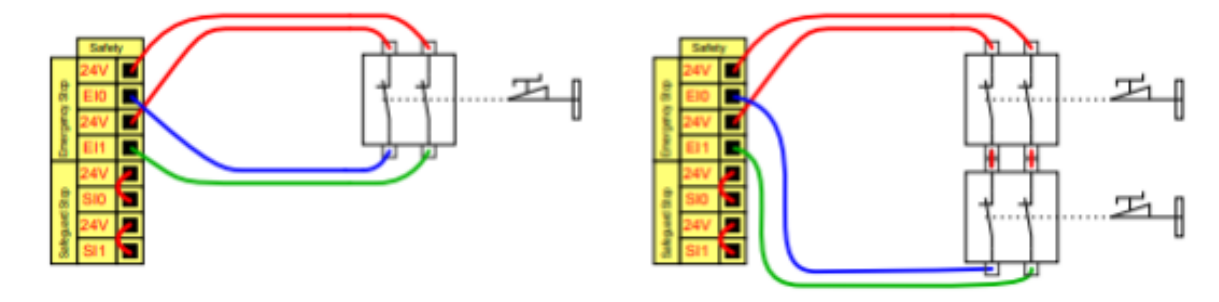


Рис.8.6 - Підключення кнопок аварійної зупинки

Цифрові входи/виходи загального призначення

У даному розділі міститься опис входів/виходів загального призначення 24 В (сірі контакти) і настроюваних входів/виходів (жовті контакти з чорним текстом) .

Входи/виходи загального призначення можуть використовуватися для непосреднього керування різним обладнанням, наприклад, пневматичним реле або для обміну даними з іншими системами ПЛК. Можливе автоматичне вимкнення всіх цифрових виходів у разі зупинці виконання програми, див. частину II. У цьому режимі сигнал виводу завжди буде низьким, якщо програма не виконується. Приклади наведено в наступних підрозділах.

У цих прикладах використовуються звичайні цифрові виходи, проте також можливе використання настроюваних виходів у випадку, якщо вони не налаштовані на виконання функцій безпеки.

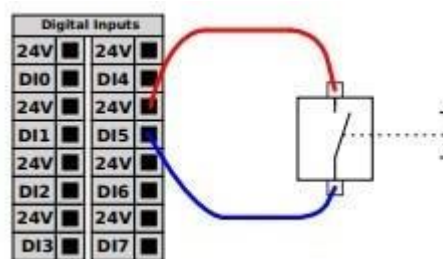


Рис.8.7 - Підключення кнопки до цифрового входу

Розглянемо перелік контурів контролю й керування, які наведені на схемі автоматизації по позиційним номерам контурів:

- 1 – керування та контроль стану ЕД М1 пневматичного захвату ;
- 2 – керування та контроль стану ЕД М2 транспортера цементних мішків;
- 3 – керування та контроль стану ЕД М3 машини для фасування мішків;
- 4 – керування та контроль стану ЕД М4 ваготримача;
- 5 – контроль наявності заготівки на транспортері ;
- 6 – контроль наявності заготівки у машині для фасування ;;

7 – контроль наявності заготовки на столі ;

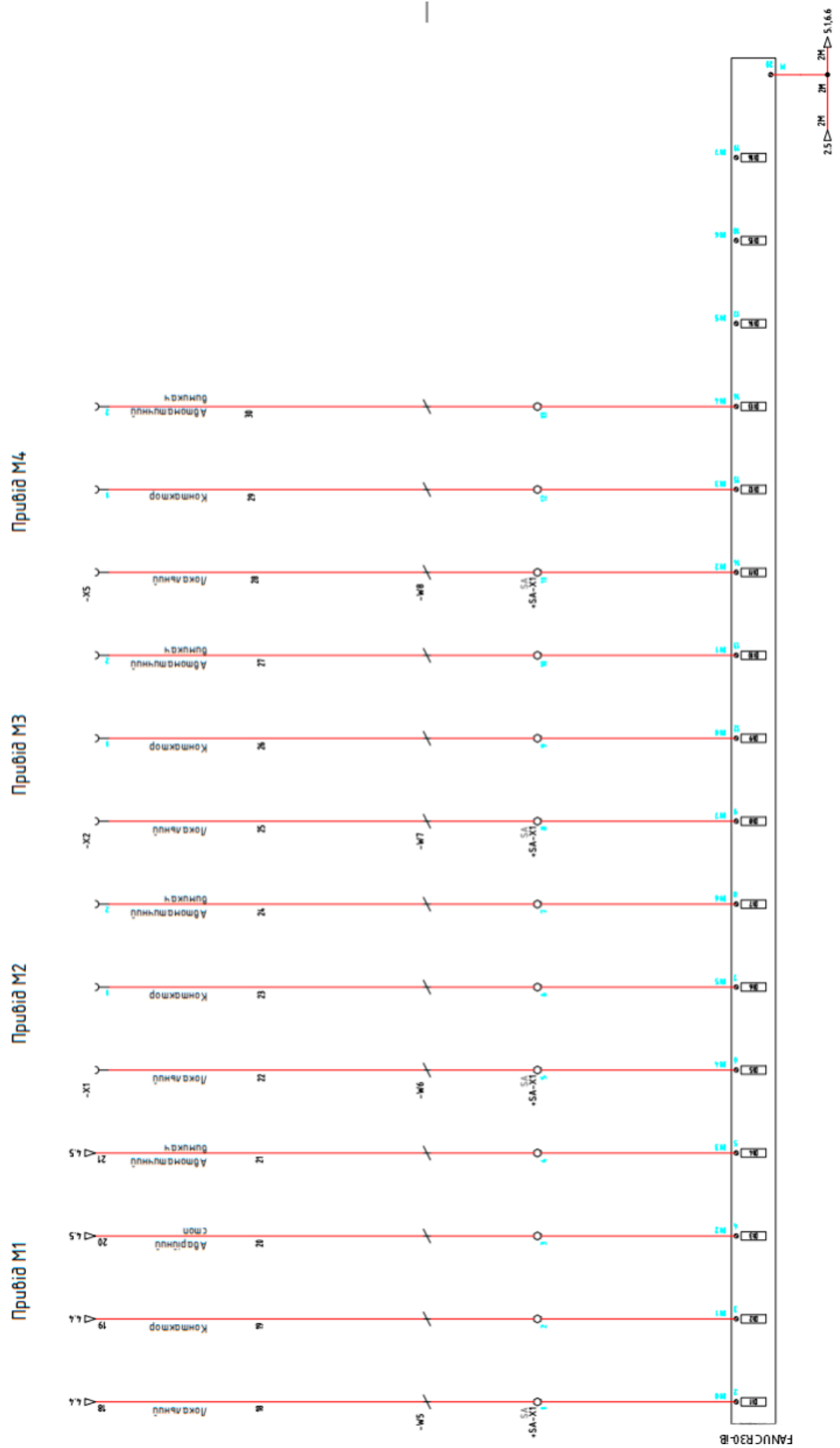


Рис.8.8 - ПЕС вводу дискретних сигналів

## Аналогові входи/виходи загального призначення

Для під'єднання аналогового інтерфейсу призначений зелений контакт. Він використовується для встановлення або вимірювання напруги (0-10 В) чи струму (4-20 мА) до та від іншого обладнання.

Для отримання найвищої точності слід виконати наступні дії.

- Контакт AG, найближчий до входу/виходу. Паралельно необхідно одночасно використовувати фільтр загального режиму.
- Необхідно використовувати загальне заземлення (0 В) для обладнання та блоку керування.
- Аналогові входи/виходи не мають гальванічної ізоляції від блока керування.
- Необхідно використовувати екранований кабель або кабель з витими парами. Необхідно під'єднати екран до контакту заземлення (GND) клеми живлення Power.
- Необхідно використовувати обладнання, що працює в режимі струму. Сигнали струму менш сприйнятливі до перешкодам

## Використання аналогового виходу

Даний приклад показує керування стрічкою конвеєра з використанням аналогічного входу управління швидкістю.

входу керування швидкістю.

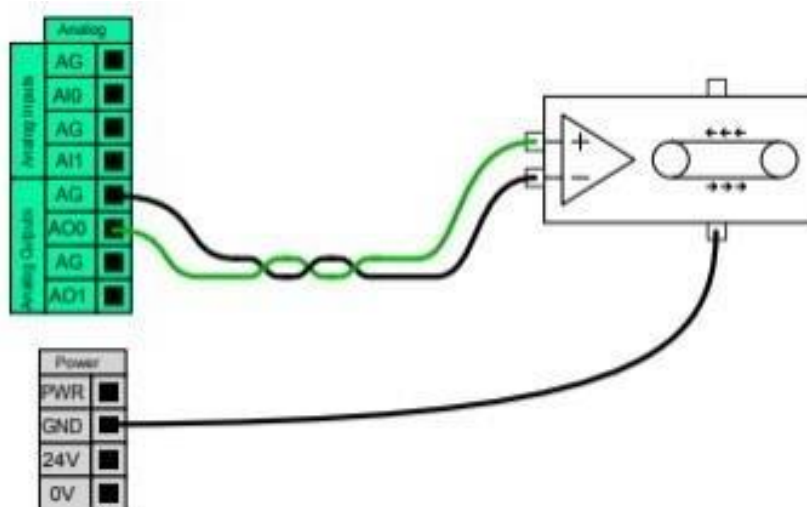


Рис.8.9 - Використання аналогового виходу

Даний приклад показує підключення аналогового датчика.

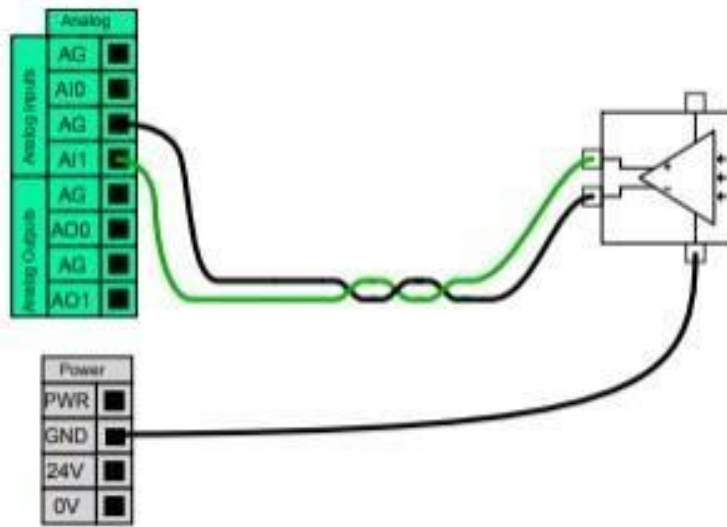


Рис.8.10 - Використання аналогового виходу



## 8.12 Охорона праці.

У нецементному виробництві існує декілька потенційно небезпечних і шкідливих факторів, що можуть впливати на персонал. Нижче наведені деякі з них:

- **Пил:** У нецементному виробництві можуть утворюватися пил із різних матеріалів, таких як вугілля, пісок, шлаки тощо. Вдихання пилу може призвести до подразнення дихальних шляхів, а також спричинити проблеми з легенями та іншими органами дихання.
- **Хімічні речовини:** У процесі нецементного виробництва можуть використовуватися різні хімічні речовини, які можуть бути токсичними або шкідливими для здоров'я. Наприклад, розчинники, фарби, лаки, розчини кислот тощо. Контакт з цими речовинами може викликати подразнення шкіри, алергічні реакції або навіть серйозні отруєння.
- **Фізичні навантаження:** В неконтрольованих умовах нецементного виробництва працівники можуть зазнавати фізичного перенапруження. Це може включати підйом тяжких предметів, повторювані рухи, виходи на висоту, що призводить до ризику травм.
- **Шум:** У деяких видів нецементного виробництва може бути високий рівень шуму внаслідок роботи машин і обладнання. Довготривалий вплив шуму може призвести до пошкодження слуху та інших проблем зі здоров'ям.
- **Вібрація:** Робота з деякими машинами або інструментами в нецементному виробництві може викликати вібрацію.

Можливі аварійні ситуації в системі автоматизації можуть включати наступні:

- **Відмова електропостачання:** Наслідки можуть включати втрату електричного живлення для всієї системи, зупинку обладнання та

можливу втрату даних. Для локалізації такої ситуації можна використовувати резервне джерело живлення, UPS (установки безперебійного живлення) або генератори електричної енергії. Шляхи евакуації персоналу повинні бути чітко вказані та доступні.

- Витік газу або рідини: Це може призвести до пожежі, вибуху, отруєння або інших небезпек. Для локалізації таких ситуацій можуть використовуватися автоматичні системи виявлення витіку, газові детектори або системи автоматичного відключення газопостачання. Шляхи евакуації персоналу повинні бути віддаленими від джерела витіку та чітко позначеними.

- Перевищення параметрів роботи системи: Це може призвести до перегріву, перевантаження або пошкодження обладнання. Для локалізації таких ситуацій можна використовувати автоматичні системи контролю та регулювання параметрів, які вимикатимуть або зупинятимуть обладнання в разі перевищення заданих меж. Шляхи евакуації персоналу повинні бути віддаленими від області потенційної небезпеки.

- Кібератаки або несанкціонований доступ: Це може призвести до порушення працездатності системи, втрати даних або зламу контролю над обладнанням.

Безпека під час планово-попереджувальних і аварійно-відбудовних робіт, пов'язаних із системою автоматизації, є надзвичайно важливою. Для запобігання можливим аварійним ситуаціям та їх наслідкам необхідно вживати наступні заходи:

- Провести оцінку ризиків перед початком робіт, визначити потенційні небезпеки та розробити заходи для їх запобігання або зниження.
- Детально спланувати роботи, включаючи послідовність дій, розподіл завдань та визначення ролей та відповідальності. Забезпечити доступ до всіх необхідних даних про систему автоматизації, включаючи технічну документацію, схеми та процедури безпеки.

Встановити чітку границю для зони робіт і обмежити доступ до неї лише спеціалізованому персоналу.

- Використовувати резервне джерело живлення, UPS або генератори електроенергії для локалізації випадків відмови електропостачання.
- Встановити автоматичні системи виявлення витoku, газові детектори або системи автоматичного відключення газопостачання для локалізації випадків витoku газу або рідини.
- Використовувати автоматичні системи контролю та регулювання параметрів для запобігання перевищенню параметрів роботи системи.
- Забезпечити захист системи від кібератак та несанкціонованого доступу, встановивши відповідні заходи безпеки.

Мати чітко позначені та доступні шляхи евакуації персоналу, враховуючи потенційні небезпеки та місця безпеки.

## **ВИСНОВОК ЗА РОЗДІЛОМ 8**

Висновуючи розділ, безпека під час планово-попереджувальних і аварійно-відбудовних робіт, пов'язаних із системою автоматизації, є надзвичайно важливою для забезпечення безпеки персоналу та запобігання негативних наслідків. Оцінка ризиків, детальне планування робіт, забезпечення доступу до необхідних даних, локалізація можливих аварійних ситуацій та чітко позначені шляхи евакуації персоналу є ключовими аспектами безпеки в цьому контексті.

Використання резервного джерела живлення, автоматичних систем виявлення витoku та контролю параметрів, а також заходи для захисту від кібератак є ефективними засобами локалізації потенційних небезпек і зниження ризиків.

В ході виконання розділу було розроблено варіант комплексу технічної документації САК на базі контролера фірми Fanuc R-301iB.

Розроблений комплект включає основні документи технічного забезпечення системи автоматизації керування процесу сушіння чаю. В результаті виконання отримані навички з розробки технічної документації на системи автоматизації з використанням ПЛК та сучасних пристроїв автоматизації.

## РОЗДІЛ 9. ПОПЕРЕДНЄ ОБҐРУНТУВАННЯ ЕКОНОМІЧНОЇ ДОЦІЛЬНОСТІ ВПРОВАДЖЕННЯ РТК

### 9.1. Розрахунок інвестиційних витрат на розробку та реалізацію робототехнічного комплексу

Розрахунок інвестиційних витрат на розробку та реалізацію робототехнічного комплексу включає оцінку всіх необхідних витрат на розробку, виробництво, маркетинг, обладнання та інфраструктуру. Для зручності розрахунку можна розділити витрати на кілька основних складових:

#### 1. Розробка програмного забезпечення і алгоритмів:

- Зарплата програмістів і розробників, які займаються розробкою програмного забезпечення для роботів.
- Витрати на закупівлю або оренду необхідного програмного забезпечення та розробницьких інструментів.
- Витрати на тестування, відлагодження та вдосконалення програмного забезпечення.

#### 2. Розробка апаратної складової:

- Зарплата інженерів, які працюють над розробкою апаратних компонентів роботів.
- Витрати на закупівлю або виготовлення необхідних електронних компонентів, сенсорів, механізмів та інших деталей.
- Витрати на тестування, вдосконалення та виробництво прототипів.

#### 3. Маркетинг та просування:

- Рекламні кампанії, виставкові виступи, демонстрації та інші заходи для просування робототехнічного комплексу.

- Розробка маркетингових матеріалів, веб-сайту та інших засобів зв'язку зі споживачами.
- Витрати на дослідження ринку та аналіз конкурентів.

#### 4. Обладнання та інфраструктура:

- Витрати на закупівлю необхідного обладнання, включаючи комп'ютери, робочі станції, робототехнічне обладнання тощо.
- Оренда або купівля виробничого приміщення
- 

Перелік обладнання та ціноутворення ( Ціни вказані у гривні з рахунком ПДВ, ціна може змінюватися залежно від курсу валюти )

1. Робот Universal Robot 10CB3 – 890 000 грн
2. Пневматичний захват – 35.000 грн
3. Металеві конструкції – 15.000 грн
4. Захват для робота – 37. 000 грн
5. Система тикку – 12.000 грн
6. Додаткові прилади – 20,000 грн

Загальна ціна обладнання – 1 010 00 грн

Але окрім придбання обладнання, потрібно врахувати ще послуги програмування, налагоджування та розробки РТК

Ціни на інтелектуальну працю :

1. Програмування – 250.000 грн
2. Пуско- налаштовувальні роботи – 80.000 грн
3. Розробка та проектування РТК – 50.000 грн

Загальна ціна – 380.000 грн

Враховуючи, що на виробництві є приміщення, тобто орендну плату платити не треба. Загальна сума інвестицій становить 1 390 00 грн

9.2. Розрахунок показника економічного ефекту від модернізації системи керування

Два працівники на виробництві за 8 годину зміну фаусують 480 мішків, тобто на один мішок разом витрачає одну хвилину . Робот в свою чергу витрачає цей час на зварювання 4 мішків. Підприємство з одного мішка отримує 13 грн чистого прибутку. Тобто два працівника приносить підприємству за 25 робочих днів (1 місяць) –  $480 \cdot 25 \cdot 13 = 156\,000$  грн прибутку, це вируавши затрати на маркетинг меджнмент, логістику, собівартість матеріалу та працю працівника .

Робот в свою чергу фасує, якщо в 1 зміну – 960деталей/добу , якщо в 2 – 1920 деталей/добу. Тільки прибуток з деталі вже виростає до 15, бо браку стало сутєво менше , та фасовщика вже не потрібно, а потрібен оператор , який буду забивати кондуктор заготівками та знімати їх. Данна праця оцінюється між меншою заробітньою платою на ринку України.

Тобто робот заробляю для підприємства в 1 зміну місяць  $960 \cdot 25 \cdot 15 = 360.000$  грн, а за дві зміни 720.000 грн. Тобто повне повернення інвестицій, при повній завантаженості виробництва, відбудеться, якщо робот буде працювати в 1 зміну  $1\,390\,000 / 360\,000 = 3,8$  місяці , якщо в 2 зміни  $1\,390\,000 / 720\,000 = 1,9$  місяці .

## **ВИСНОВОК ЗА РОЗДІЛОМ 9**

Підвищення продуктивності: Роботизація дозволяє автоматизувати багато процесів на виробництві цементу, що забезпечує підвищення продуктивності. Роботи можуть виконувати повторювані завдання швидше та ефективніше, порівняно з ручним працівником.

Зниження витрат: Впровадження роботів може допомогти знизити витрати на виробництво цементу. Роботи працюють безперервно та не потребують відпочинку, вони не мають втрат через втому або людські помилки. Крім того, вони можуть бути більш точними в роботі, що дозволяє зменшити кількість відходів та покращити якість продукції.

Забезпечення безпеки: Виробництво цементу може бути небезпечним для людського здоров'я через пил та інші шкідливі речовини. Використання роботів у таких умовах допомагає зменшити ризик для працівників, оскільки вони можуть виконувати завдання в умовах, які для людей були б шкідливими або небезпечними.

Покращення якості продукції: Роботи можуть бути програмовані для виконання завдань з високою точністю та стабільністю. Це може допомогти покращити якість виготовленого цементу та забезпечити його відповідність вимогам стандартів.

Використання роботів на виробництві цементу має потенціал зробити процес більш ефективним, безпечним та економічним.

## СПИСОК ЛІТЕРАТУРИ

1. Хобін В. А. Конспект лекцій з курсу "Ідентифікація та моделювання технологічних об'єктів" [Електронний ресурс] : для студентів, які навчаються за спец. 151 - "Автоматизація комп'ютерно-інтегровані технології" ден. та заоч. форм навчання / В. А. Хобін ; відп. за вип. В. А. Хобін ; Каф. автоматизації технологічних процесів і робототехнічних систем. — Одеса : ОНАХТ, 2016. — Електрон. текст. дан.: 96 с.

<https://elc.library.ontu.edu.ua/library-w/DocumentDescription?docid=OdONAHТcnv.BibRecord.157312>

2. Хобін В.А., Світий І.М. Теорія автоматичного керування: Посібник до виконання комплексної курсової роботи. Одеська національна академія харчових технологій, 2020. — 277 с.

3. Муратов В. Г. Метрологія, технологічні вимірювання та прилади [Текст] : навч. посіб./ В. Г. Муратов. — Вид. 2-ге, допов. — Київ : Освіта України, 2016. — 364 с.

<https://elc.library.onaft.edu.ua/library-w/DocumentDescription?docid=OdONAHТcnv.BibRecord.155904>

4. Савицький, Володимир Костянтинович Технічні засоби автоматизації [Текст]: навч. посіб. /В.К. Савицький, Р.М. Федоришин; Нац. ун-т "Львівська політехніка".— Львів: Вид-во Львів. політехніки, 2018.— 292 с.

<https://elc.library.ontu.edu.ua/library-w/DocumentSearchResult;jsessionid=6CD40B15F14DD9265BF7FD9A559C892C>

5. Левінський В. М. Конспект курсу лекцій з курсу "Технічні засоби автоматизації". Розділ "Виконавчі пристрої" [Електронний ресурс] : для студентів спец. 151 "Автоматизація та комп'ютерно-інтегровані технології" ден. та заоч. форм навчання / В. М. Левінський, М. В. Левінський ; відп. за вип. В. А. Хобін ; Каф. автоматизації технологічних процесів і робототехн. систем. — Одеса : ОНАХТ, 2018. — Електрон. текст. дані: 150 с.

<https://elc.library.ontu.edu.ua/library-w/DocumentDescription?docid=OdONAHТcnv.BibRecord.162866>

6. Левінський В. М. Основи створення проектів автоматизації в середовищі ТІАPortal [Електронний ресурс] : електрон. навч. посіб. для підгот. студентів спец. "Автоматизація та комп'ютерно-інтегровані технології" / В. М. Левінський, М. В. Левінський ; Одес. нац. акад. харч. технологій. — Одеса : ОНАХТ, 2019. — Електрон. текст. дані: 177 с.

<https://elc.library.ontu.edu.ua/library-w/DocumentDescription?docid=OdONAHТcnv.BibRecord.166190>

7 . <https://www.universal-robots.com/>

8 . <https://www.universal-robots.com/articles/>

9. Скаковський Ю. М. Конспект лекцій з курсу "Проектування систем автоматизації". Ч. 2 [Електронний ресурс] : для студентів проф. напряму підгот. 6.050202 всіх форм навчання / Ю. М. Скаковський ; відп. за вип. В. А. Хобін ; Каф. АТП і РС. — Одеса: ОНАХТ, 2017. Електрон. текст. дані: 117 с.

<https://elc.library.ontu.edu.ua/library-w/DocumentDescription?docid=OdONAHТcnv.BibRecord.160548>

10. Безпека життєдіяльності та основи охорони праці : конспект лекцій [Електронний ресурс]/ О. О. Фесенко, В. М. Лисюк, С. М. Неменуца, З. М. Сахарова ; Каф. безпеки життєдіяльності. — Одеса : ОНАХТ, 2021. — Електрон. текст. дані: 149 с.

<https://elc.library.ontu.edu.ua/library-w/DocumentDescription?docid=OdONAHТ.1739079>