

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук та технологій
Кафедра комп'ютерних систем та мереж

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему СИСТЕМА КЕРУВАННЯ АВТОМАТИЗОВАНИМ
ПРОЦЕСОМ СОРТУВАННЯ ПАКУНКІВ

Виконав(ла): студент(ка) 2 курсу, групи 513м

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

Освітня програма (спеціалізація)
«Комп'ютерні системи та мережі»

ДЕМЧАН Д.С.

(ПРИЗВИЩЕ та ініціали)

Керівник ГОЛУБ Т.В.

(ПРИЗВИЩЕ та ініціали)

Рецензент ЄФИМЕНКО М.В.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет Комп'ютерних наук і технологій
Кафедра «Комп'ютерні системи та мережі»
Ступінь вищої освіти магістерський
Спеціальність 123 Комп'ютерна інженерія
(код і найменування)
Освітня програма (спеціалізація) «Комп'ютерні системи та мережі»
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри Кудерметов Р.К.

«_____» _____ 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ДЕМЧАН Дмитро Сергійович

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту Система керування автоматизованим процесом сортування пакунків

керівник проєкту (роботи) к.т.н. ГОЛУБ Тетяна Василівна,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від «18» жовтня 2024 року №149

2. Строк подання студентом проєкту (роботи) 18 грудня 2024

3. Вихідні дані до проєкту (роботи) алгоритм автоматизованого процесу сортування

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1) Теоретична частина

2) Аналіз алгоритмів систем керування автоматизованим процесом сортування пакунків

3) Дослідження алгоритму систем сортування на основі програмованої логічної

4) Експериментальна частина

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

6. Консультанти розділів проєкту (роботи)

| Розділ | ПРИЗВИЩЕ, ініціали та посада консультанта | Підпис, дата | |
|---------------|---|----------------|---------------------------|
| | | завдання видав | прийняв виконане завдання |
| 1-4 | ГОЛУБ Т.В. к.т.н., доцент | | |
| нормоконтроль | ЩЕРБАК Н.В., ст. викл. | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

7. Дата видачі завдання « 4 » листопада 2024 року.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломного проєкту (роботи) | Строк виконання етапів проєкту (роботи) | Примітка |
|-------|--|---|----------|
| 1 | Аналіз існуючих систем керування автоматизованим процесом сортування пакунків та їх різновидів | 10.10.2024 р. | |
| 2 | Розробка структури системи | 16.10.2024 р. | |
| 3 | Розробка алгоритму роботи системи | 21.10.2024 р. | |
| 4 | Дослідження автоматизованої системи сортування | 26.10.2024 р. | |
| 5 | Реалізація автоматизованої системи сортування | 30.10.2024 р. | |
| 6 | Оформлення отриманих результатів у ПЗ | 10.11.2024 р. | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Студент(ка)

_____ Дмитро ДЕМЧАН _____
 (підпис) (Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

_____ Тетяна ГОЛУБ _____
 (підпис) (Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

ПЗ: 87 с., 30 рис., 2 табл., 18 джерел.

АВТОМАТИЗАЦІЯ, СИСТЕМИ СОРТУВАННЯ, КОНВЕЄР, ПОВОРОТНИЙ МЕХАНІЗМ, ПЛІС, FPGA

Об'єкт розробки – система керування автоматизованим процесом сортування пакунків.

Метою роботи є розробка системи керування автоматизованим процесом сортування пакунків.

Проєкт складається з чотирьох розділів. Перший розділ присвячено вивченню сучасних методів автоматизації процесів сортування пакунків. Проводиться детальний огляд основних складових автоматизованих систем сортування. У другому розділі проводиться детальний аналіз алгоритмів систем керування автоматизованим процесом сортування пакунків. Третій розділ присвячено моделюванню алгоритму системи керування автоматизованим процесом сортування пакунків на основі програмованої логічної інтегральної схеми. В четвертому розділі проводиться детальний аналіз роботи алгоритму системи автоматного керування на основі програмованої логічної схеми. Наприкінці розділу наведено рекомендації щодо підбору оптимальної ПЛІС для створення системи керування автоматизованим процесом сортування пакунків.

У рамках дипломного дослідження було виконано аналіз розробленої системи автоматичного керування конвеєром, моделювання її роботи та дослідження перспектив застосування подібних систем в інших галузях промисловості. В результаті виконаної роботи отримано модульну систему сортування пакунків, реалізовану на ПЛІС, яка надає можливість легкого масштабування та адаптації для різних комбінацій вхідних умов.

ABSTRACT

Explanatory note to the master's work: 87 p., 30 figures, 2 tables, 18 sources.

AUTOMATION, SORTING SYSTEMS, CONVEYOR, ROTARY MECHANISM

Object research – control system for the automated package sorting process.

The subject of research is methods of building secure computer networks based on cisco network technologies.

The aim of the work is to develop a control system for the automated package sorting process and analyze existing package sorting systems. The project consists of four sections.

The first section is devoted to the study of modern methods of automating the processes of sorting packages. A detailed overview of the main components of automated sorting systems is provided.

The second section provides a detailed analysis of the algorithms of control systems for the automated package sorting process.

The third section is devoted to modeling the algorithm of the control system for the automated process of sorting packages based on a programmable logic integrated circuit.

Section 4 provides a detailed analysis of the algorithm of the automatic control system based on a programmable logic circuit. The chapter concludes with recommendations for selecting the optimal FPGA to create a control system for the automated package sorting process.

As part of the diploma research, a comprehensive analysis of the developed automatic conveyor control system was performed, including modeling its operation and researching the prospects for the use of similar systems in other industries.

ЗМІСТ

| | |
|--|----|
| Перелік скорочень та умовних позначень | 9 |
| Вступ..... | 10 |
| 1 Теоретична частина..... | 11 |
| 1.1 Опис систем керування автоматизованим процесом сортування пакунків та їх різновидів | 11 |
| 1.1.1 Висувний сортувальник коліс/роликів/стрічок | 12 |
| 1.1.2 Сортувальник з поворотною рукою | 13 |
| 1.1.3 Штовхачий сортувальник | 13 |
| 1.1.4 Сортувальники Tilt Tray та Crossbelt..... | 14 |
| 1.1.5 Сортувальник штовхаючих лотків | 15 |
| 1.1.6 Сортувальник типу "бомбардувальник" | 15 |
| 1.1.7 Сортувальник мішечків/кишень | 16 |
| 1.1.8 Пересувні стрічкові конвеєри | 17 |
| 1.1.9 Вузкий стрічковий сортувальник | 18 |
| 1.2 Огляд основних складових систем керування автоматизованими процесом сортування пакунків..... | 18 |
| 1.2.1 Конвеєрні системи..... | 19 |
| 1.2.2 Сканери штрихкодів та RFID-рідери | 20 |
| 1.2.3 Системи зважування та вимірювання розмірів | 20 |
| 1.2.4 Розподільчі механізми (сортувальники) | 21 |
| 1.2.5 Централізовані системи керування WMS і ERP | 23 |
| 1.2.6 Системи візуалізації та моніторингу | 26 |
| 1.3 Вивчення сучасних методів автоматизації процесів сортування пакунків... | 28 |
| 1.4 Актуальність портативної системи керування автоматизованим процесом сортування пакунків для малих складів..... | 30 |
| 1.5 Основні задачі та вимоги до портативної системи керування автоматизованим процесом сортування пакунків | 31 |

| | |
|--|----|
| 2 Аналіз алгоритмів систем керування автоматизованим процесом сортування пакунків | 33 |
| 2.1 Загальний опис роботи систем керування автоматизованим процесом сортування пакунків..... | 33 |
| 2.1.1 Стан очікування..... | 33 |
| 2.1.2 Ідентифікація | 34 |
| 2.1.3 Передача інформації..... | 35 |
| 2.1.4 Визначення маршруту..... | 36 |
| 2.1.5 Перевірка стану конвеєра..... | 36 |
| 2.1.6 Перевірка місцезнаходження пакунку | 38 |
| 2.1.7 Перевірка сортувальних механізмів..... | 38 |
| 2.2 Критерії СКАПСП..... | 39 |
| 2.3 Алгоритм систем керування автоматизованим процесом сортування пакунків на основі WMS..... | 40 |
| 2.3.1 Стан очікування..... | 40 |
| 2.3.2 Перевірка наявності пакунку на першому контрольному пункті..... | 41 |
| 2.3.3 Перевірка наявності пакунку на другому контрольному пункті | 42 |
| 2.3.4 Перевірка наявності пакунку на третьому контрольному пункті..... | 43 |
| 2.3.5 Перевірка наявності пакунку на четвертому контрольному пункті | 44 |
| 2.3.6 Перевірка наявності пакунку на п'ятому контрольному пункті..... | 45 |
| 2.3.7 Перевірка наявності пакунку на шостому контрольному пункті | 46 |
| 2.4 Недоліки систем керування автоматизованим процесом сортування пакунків на основі WMS та ERP..... | 47 |
| 3 Дослідження алгоритму систем сортування на основі програмованої логічної інтегральної схеми | 48 |
| 3.1 Огляд АСС на основі мікросхем | 48 |
| 3.2 Алгоритм роботи систем сортування на основі ПЛІС | 50 |
| 3.2.1 Стан очікування..... | 51 |
| 3.2.2 Перевірка наявності пакунку на першому контрольному пункті..... | 51 |
| 3.2.2.1 Пакунок відсутній на першому поворотному механізмі..... | 51 |

| | | |
|-------|--|----|
| 3.2.3 | Перевірка наявності пакунку на другому контрольному пункті | 52 |
| 3.2.4 | Перевірка наявності пакунку на третьому контрольному пункті..... | 53 |
| 3.2.5 | Перевірка наявності пакунку на четвертому контрольному пункті | 53 |
| 3.3 | Структура та компоненти алгоритму систем сортування на основі ПЛІС . | 54 |
| 4 | Експериментальна частина | 62 |
| 4.1 | Аналіз результатів експерименту та оцінка ефективності системи сортування на основі ПЛІС | 62 |
| 4.2 | Розробка керуючого автомата Мура для СКАПСП | 64 |
| 4.3 | Реалізація СКАПСП на Cyclone III, Cyclone IV та Arria II GX..... | 71 |
| | Висновки..... | 81 |
| | Перелік джерел посилання | 82 |
| | Додаток А Лістинги програм..... | 84 |

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

АСС -Автоматизовані системи сортування;

ПЛІС – Програмована логічна інтегральна схема;

ПЛК - Програмованого логічний контролер;

СКАПСІ - Система керування автоматизованим процесом сортування пакунків;

CRM -Customer Relationship Management;

ERP - Enterprise Resource Planning;

FPGA - Field-Programmable Gate Array;

RFID - Radio Frequency Identification;

VHDL – Hardware Description Language;

WMS - Warehouse Management System.

ВСТУП

Автоматизовані системи сортування на складах є критично важливими у сучасних умовах через глобалізацію логістики, швидке зростання електронної комерції та потребу в оптимізації ланцюгів постачання. Зі зростанням обсягів товарообігу, високою конкуренцією та вимогами до швидкої доставки, автоматизація складів стає не просто актуальною, а обов'язковою для успіху бізнесу.

АСС, як правило, складаються з взаємопов'язаних елементів, включаючи транспортні системи (конвеєри, роботи), системи ідентифікації (штрихкод-сканери, RFID), контрольні пристрої (сенсори, вимірювальні модулі) та інтегровані системи управління на основі ПЛІС і програмного забезпечення, що дозволяє ефективно керувати всіма компонентами системи та забезпечувати обмін даними в режимі реального часу через Інтернет речей.

Автоматизовані системи сортування на складах пропонують низку переваг, включаючи підвищення продуктивності за рахунок паралельної обробки великих обсягів даних та мінімізації людських помилок. Вони забезпечують гнучкість, адаптуючись до різних вимог, і оптимізують використання простору. Завдяки автоматизації знижуються витрати на оплату праці, підвищується точність і швидкість обробки замовлень, що призводить до підвищення задоволеності клієнтів. Сучасні системи є енергоефективними та безпечними, а також легко інтегруються з іншими системами управління підприємством, забезпечуючи повну автоматизацію логістичних процесів.

Автоматизовані системи сортування на складах є критично важливими для сучасних підприємств, оскільки вони підвищують ефективність, скорочують витрати, забезпечують точність і швидкість обробки пакунків. Завдяки гнучкій архітектурі та можливості інтеграції з іншими технологіями, такі системи є вигідним рішенням для підприємств будь-якого масштабу.

1 ТЕОРЕТИЧНА ЧАСТИНА

1.1 Опис систем керування автоматизованим процесом сортування пакунків та їх різновидів

Сортування вантажів – це організований процес розподілу товарів на групи за певними критеріями. Для виробництва це означає розподіл готової продукції за типами та призначенням, а для логістики – формування багатоасортиментних замовлень шляхом збору товарів з різних зон складу.

Система керування автоматизованим процесом сортування пакунків (або коротко – система сортування) – це комплекс програмного забезпечення та обладнання, який автоматизує процес розподілу пакунків за різними критеріями (адреса, тип товару, розмір тощо) на складі або сортувальному центрі. Ця система використовує різноманітні технології, такі як сканування штрих-кодів, розпізнавання зображень, роботизовані руки та конвеєри, для ефективного та точного виконання завдання.

Для вибору оптимальної системи сортування на підприємстві необхідно комплексно проаналізувати ряд факторів. Це стосується як загальних характеристик виробництва (пропускна спроможність, кількість напрямків сортування, графік роботи, розміри складу, умови довкілля), так і специфічних особливостей товарів, що сортуються (розмір, вага, крихкість). Враховуючи всі ці параметри, можна підібрати обладнання з необхідною продуктивністю та функціоналом, яке забезпечить ефективне та безперебійне виконання поставлених завдань.

Загалом, АСС поділяються на дві широкі категорії: кейсові сортувальники, які сортують і транспортують цілі коробки, контейнери або замовлення з одного місця на інше в межах об'єкта, та унітарні сортувальники, які сортують і транспортують окремі предмети по всьому об'єкту.

Кейсові сортувальники допомагають зменшити кількість ручних операцій, необхідних для обробки замовлень і відправлень, скорочуючи

загальний цикл виконання замовлення та дозволяючи ефективніше використовувати трудові ресурси для виконання функцій, які додають цінність вашій діяльності.

Унітарні сортувальники дозволяють значно підвищити ефективність у процесі виконання замовлень. Багато з цих технологій, хоча й не всі, інколи називають "петльовими сортувальниками" через їхню конфігурацію, яка часто утворює кругову петлю.

Вибір системи сортування залежить від індивідуальних потреб виробництва. Крім традиційних рішень, існують й інші опції, які можуть бути більш ефективними в конкретних випадках.

1.1.1 Висувний сортувальник коліс/роликів/стрічок

Висувний колісний сортувальник має конструкцію з коліс, роликів або стрічок, вбудованих у стрічковий конвеєр, які «висуваються» для підняття або переміщення товарів під кутом 30 або 90 градусів на інший конвеєр нижче за потоком. Він особливо ефективний для обробки однотипних продуктів, як-от товари в пакетах або коробках, що робить його чудовим варіантом для пакувальних процесів. Завдяки своїй конструкції, висувний колісний сортувальник забезпечує точне та контрольоване відведення продукції.

Пропускна здатність колісних висувних сортувальників зазвичай складає 40-100 коробок на хвилину.

Переваги: висувні колісні сортувальники потребують відносно низьких капіталовкладень порівняно з іншими сортувальними системами високої продуктивності. Вони мають модульну конструкцію, що дозволяє швидко додавати або видаляти секції за потреби, а також забезпечують надійність роботи.

Недоліки: цей тип сортувальника має низьку або середню швидкість сортування, що може обмежувати його максимальну пропускну здатність за один робочий цикл. Підприємствам, що працюють з дуже маленькими, легкими або нерівномірно завантаженими коробками, може знадобитися протестувати свою продукцію перед вибором більш дорогого типу сортувальної системи.

1.1.2 Сортувальник з поворотною рукою

Сортувальник з поворотною рукою, або лопатевий сортувальник, має одну або кілька поворотних «рук», розташованих уздовж конвеєра. Коли рука активується, вона повертається, перенаправляючи і сортує коробки на вторинний конвеєр для подальшої обробки. Сортувальники з поворотним маніпулятором широко використовуються для сортування вантажів, особливо в обробці багажу та посилок.

Продуктивність таких сортувальників зазвичай становить 50-100 одиниць на хвилину.

Переваги: цей тип сортувальника є відносно недорогим порівняно з іншими рішеннями. При роботі з коробками відповідного розміру він забезпечує високу ефективність та надійність.

Недоліки: сортувальники з поворотною рукою найбільше підходять для сортування коробок. Поліетиленові пакети, які часто використовуються в електронній комерції, можуть застрягти під лопаттю маніпулятора, спричиняючи пошкодження або зупинку системи. Через силу, з якою продукт перенаправляється, ці сортувальники не підходять для дуже крихких предметів.

1.1.3 Штовхачий сортувальник

Штовхаючі сортувальники складаються з одного або кількох пневматичних штовхачів, розміщених на стрічковому конвеєрі. При отриманні сигналу штовхач висувається під прямим кутом до конвеєра, спрямовуючи коробку або сумку на додатковий конвеєр або жолоб для подальшої обробки. Такі сортувальники широко застосовуються в пакувальній та транспортній галузях.

Продуктивність штовхачоючих сортувальників зазвичай становить 10-30 коробок на хвилину.

Переваги: ці прості машини забезпечують ефективне сортування за економічною ціною. Вони добре працюють з упаковками з менш якісного картону та з товарами нерівномірної ваги.

Недоліки: штовхачоючі сортувальники мають нижчу швидкість і меншу

делікатність порівняно з іншими сортувальниками, як-от сортувальники з висувними башмаками чи колесами, що обмежує обсяги обробки. Підприємствам з високими темпами зростання, ймовірно, знадобиться інший тип сортувальника, якщо прогнозований обсяг перевищить можливості цього варіанту.

1.1.4 Сортувальники Tilt Tray та Crossbelt

Сортувальники Tilt Tray та Crossbelt є окремими типами сортувальних систем, проте через деякі спільні характеристики їх часто розглядають разом. Зазвичай вони мають схожу ходову частину і привід. Tilt Tray-сортувальник складається з піддонів, закріплених на ряді візків, що з'єднані у «потяг» і безперервно переміщують продукцію по замкненій траєкторії.

У системі є одна або кілька зон індукції, де продукція завантажується на піддони, та точки сортування (зазвичай жолоби), де формуються окремі замовлення. Піддони нахиляються, коли потрібно перемістити продукцію до відповідного жолоба. Завантаження продукції на піддони може виконуватися вручну або автоматично за допомогою станцій індукції, розміщених у кількох точках петлі.

Стрічковий сортувальник призначений для розподілу товарів, направляючи їх у жолоб. Головна особливість полягає в механізмі, за допомогою якого це здійснюється. На відміну від нахильного лотка, поперечна стрічка являє собою мініатюрний конвеєр, що проходить перпендикулярно до підлоги петлі. Коли стрічка активується, вона переміщує або штовхає продукт з петлевого конвеєра в жолоб або на вивідний конвеєр.

Обидва типи систем часто використовуються для сортування великих обсягів товарів при виконанні замовлень і їх відвантаженні, особливо в індустрії електронної торгівлі одягом.

Продуктивність таких сортувальників зазвичай становить від 100 до 200+ коробок на хвилину.

Переваги: сортувальники з нахиленим лотком і поперечною стрічкою забезпечують високу швидкість сортування в порівнянні з іншими системами.

Вони здатні обробляти різноманітні види продукції, що робить їх ідеальними для підприємств, що працюють з багатьма категоріями товарів. Крім того, низький рівень шуму створює безпечніше і комфортніше робоче середовище для співробітників.

Недоліки: найзначнішим недоліком цих сортувальників є їхня висока вартість, яка зазвичай перевищує ціну інших сортувальних систем. Також вони не завжди ефективно справляються з крихкими товарами.

1.1.5 Сортувальник штовхаючих лотків

Сортувальник із штовхаючими лотками складається з лотків, які розташовані на візках і переміщують продукцію по безперервному конвеєру. Основна відмінність полягає в тому, що, на відміну від сортувальників з нахиленим лотком, які використовують гравітаційний жолоб для відведення товару, у сортувальнику з штовхаючими лотками для цього застосовується штовхаюча планка.

Продуктивність таких сортувальників зазвичай становить від 30 до 60 коробок на хвилину.

Переваги: оскільки продукт не спускається по гравітаційному жолобу, сортувальник із штовхаючими лотками забезпечує більш м'яке сортування, що робить його ідеальним для роботи з крихкими товарами.

Недоліки: незважаючи на м'якість процесу, штовхаючі лотки мають нижчу швидкість сортування, що може бути недостатньо швидко для операцій, які потребують більшої пропускної здатності.

1.1.6 Сортувальник типу "бомбардувальник"

Сортувальник типу "бомбардувальник", також відомий як плоский сортувальник, складається з лотків, які діють як своєрідні люки (технологія отримала свою назву від бомбового відсіку військового літака), що відкриваються, щоб скинути продукт. Продукт може бути скинутий у гравітаційний жолоб для подальшого ручного сортування або безпосередньо в транспортну коробку або контейнер. Ці сортувальники в основному використовуються в текстильній, ювелірній та фармацевтичній промисловості.

Продуктивність сортувальників типу "бомбардувальник" зазвичай коливається в межах 100-230 лотків на хвилину.

Переваги: висока швидкість сортування. Сортувальники типу "бомбардувальник" забезпечують високу швидкість сортування одиничних товарів за більш доступною ціною порівняно з сортувальниками з нахиленим лотком та поперечною стрічкою.

Ефективність: вони особливо ефективні, якщо продукт можна відразу сортувати в транспортну коробку, значно скорочуючи час упаковки та трудові витрати.

Недоліки: обмеження за розміром товарів. Ці сортувальники найкраще підходять для обробки невеликих товарів і зазвичай не здатні обробляти широкий діапазон розмірів і типів продукції.

Виникають проблеми з обробкою великих товарів. Вони, як правило, не дуже добре справляються з великими коробками і не працюють так швидко, як сортувальники з нахиленим лотком і поперечною стрічкою.

1.1.7 Сортувальник мішечків/кишень

Сортувальник мішків або кишень складається з ряду мішків, які підвішуються до підвісного візка (по одному мішку на візок). Ці пристрої часто використовуються в процесах комплектації, коли товари пакуються в пакети партіями (по одній одиниці на пакет). Деякі сортувальники можуть зберігати товар у карусельних петлях, звідки продукція автоматично витягується для замовлення. Ця функція робить їх відмінним вибором для обробки повернень: повернуті товари можна помістити в петлі зберігання, поки не накопичиться достатня кількість одиниць одного найменування, після чого їх можна автоматично витягнути для упаковки в коробки та подальшої відправки на зберігання.

Такі сортувальники часто використовуються в індустрії одягу та електронної комерції. Більшість моделей мають обмеження по вазі до 10 фунтів, хоча деякі можуть обробляти товари вагою до 20 фунтів. Вони здатні працювати з одягом, що висить або лежить на підлозі, а також з невеликими

галантерейними товарами та коробками з-під взуття.

Продуктивність сортувальників мішків зазвичай становить 100-120 пакетів на хвилину.

Переваги: при правильному використанні сортувальники мішків можуть бути високоефективними для різних застосувань. Вони здатні обробляти широкий спектр розмірів і типів продукції; все, що може поміститися в мішок, можна ефективно сортувати. Крім того, вони можуть організовувати пакування товарів таким чином, щоб важкі предмети розміщувалися першими внизу коробки, а легкі - зверху. Товари також можна впорядковувати за стилем, кольором або розміром, що прискорює викладку в роздрібному магазині.

Недоліки: хоча сортувальники мішків можуть ефективно обробляти тверді товари, вони не підходять для роботи з крихкими предметами або надто великими чи важкими товарами.

1.1.8 Пересувні стрічкові конвеєри

Пересувний стрічковий транспортер (він же переносний або мобільний) – ще один популярний на сьогодні вид даної техніки. Оснащений спеціальними колесами, що робить його ефективним пристроєм для транспортування вантажів. А ще пересувний стрічковий конвеєр можна використовувати для завантаження, розвантаження продукції безпосередньо з транспорту для цього достатньо лише встановити обладнання біля машини. Сфера застосування такої моделі не обмежена: вона здатна транспортувати будь-які види вантажів(кускові, сипучі, штучні). Можна регулювати рукояткою висоту підймання або опускання (трикутник складається та розкладається).Стрічкові транспортери змішаного типу: змішаний тип по суті являє собою комбінацію всіх вище перелічених видів складського обладнання. Наприклад, для складських приміщень з обмеженим простором, де немає можливості встановити прямі або похилі моделі, встановлюються Z-образні або L-образні конвеєри. Як ви вже могли здогадатися, вони поєднують в собі відразу 2 види: горизонтальний і вертикальний. За бажанням замовника виготовляються транспортери різноманітних конструкцій: горизонтальні пересувні.

1.1.9 Вузкий стрічковий сортувальник

Вузькосмуговий сортувальник використовує вузькі стрічки для транспортування товарів і поділяється на два типи: з 90-градусним відхиленням (NBS-90) та з 30-градусним відхиленням (NBS-30). У моделі NBS-90 ряд роликів з високим коефіцієнтом тертя, розташованих між несучими ременями, піднімаються над ними, захоплюючи продукт і відводячи його під прямим кутом. Цей тип особливо ефективний для сортування товарів безпосередньо в транспортні гейлорди або контейнери, без необхідності ручного втручання. Завдяки своїй компактній конструкції та високій точності відведення, вузькосмуговий сортувальник оптимізує використання простору, що дозволяє інтегрувати його в існуючі системи з мінімальними змінами в інфраструктурі.

У NBS-30 колеса з фіксованим кутом нахилу піднімаються між несучими стрічками, відводячи продукт під кутом 30 градусів.

Переваги: вузькосмуговий сортувальник є дуже гнучким і економічним рішенням для сортування. Він працює тихо і споживає мінімум енергії.

Недоліки: хоча ці сортувальники гнучкі, вони не підходять для всіх видів продукції та швидкостей. Дуже маленькі коробки (6 дюймів або менше) або ті, що мають нерівномірну вагу при високих швидкостях, можуть оброблятися не найкращим чином. Крім того, їх зазвичай не використовують при швидкості понад 100 картонних коробок на хвилину.

1.2 Огляд основних складових систем керування автоматизованими процесом сортування пакунків

Основні компоненти системи сортування включають програмне забезпечення для управління всіма функціями системи, обладнання (конвеєри, сканери, роботизовані руки тощо), датчики для збору інформації про пакунки та штрих-коди для їх унікальної ідентифікації див. рис 1.1.

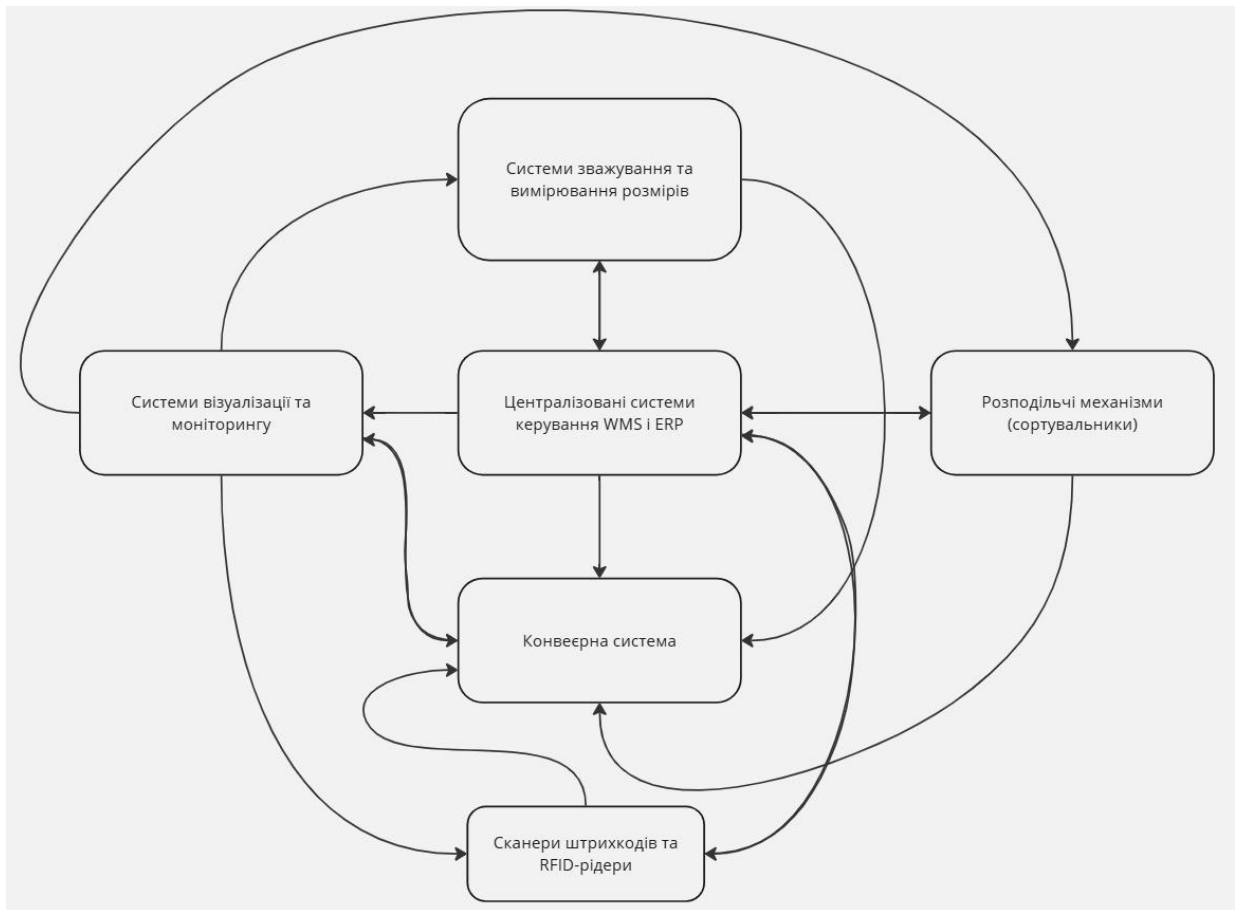


Рисунок 1.1- Схема структури СКАПСІ

1.2.1 Конвеєрні системи

Конвеєрні системи відіграють ключову роль у сучасних складах та логістичних центрах. Вони забезпечують ефективне переміщення та сортування великих обсягів товарів. Різні типи конвеєрів мають свої особливості та застосовуються для вирішення конкретних задач у процесі сортування пакунків.

Стрічкові конвеєри є одними з найпоширеніших. Вони складаються з безперервної стрічки, яка плавно переміщує пакунки. Стрічкові конвеєри ідеально підходять для транспортування легких та середніх вантажів на довгі відстані, а також для сортування за пунктами призначення.

Рольганги, або роликові конвеєри, складаються з ряду роликів. Вони можуть бути як приводними, так і не приводними. Рольганги часто використовують для переміщення важких пакунків або контейнерів, а також як

проміжний елемент між різними типами конвеєрів.

Кожен з перелічених типів конвеєрів має свої переваги та застосовується для вирішення конкретних завдань у процесі сортування. Вибір того чи іншого типу конвеєра залежить від таких факторів, як розмір і вага пакунків, необхідна швидкість транспортування, конфігурація складу та інші.

1.2.2 Сканери штрихкодів та RFID-рідери

Автоматичні сканери та RFID-рідери зчитують інформацію з пакунків, таку як призначення, пріоритет і категорію. Це дозволяє системі швидко та точно ідентифікувати кожен пакунок. Коли RFID-картка пред'являється до RFID-зчитувача, картка передає код назад до зчитувача, і цей процес називається зворотним розсіюванням. RFID-зчитувач приймає код (у шістнадцятковому форматі) та передає його до мікроконтролера для автентифікації. Якщо код відповідає тому, що зберігається у зовнішній пам'яті, мікроконтролер передає сигнал до ПЛК. У разі невідповідності мікроконтролер не генерує сигнал.

Після отримання сигналу від мікроконтролера ПЛК приводить у дію пневматичні циліндри. На РК-дисплеї можна побачити тег. Якщо пред'явлено невірну картку, мікроконтролер покаже на дисплеї повідомлення «незаконний доступ».

RFID складається з маленького чипа та антени і здатний зберігати 2000 біт даних. Оскільки ця технологія перевершує штрихкоди, вона почала замінювати їх. Позначений об'єкт випромінює радіочастотний сигнал, що робить RFID технологію придатною для різних застосувань: контроль доступу, відстеження продуктів, відстеження людей та тварин, ідентифікація людей.

1.2.3 Системи зважування та вимірювання розмірів

Вагові датчики і модулі для вимірювання габаритів автоматично визначають вагу та розміри пакунків, що є важливим для їхньої подальшої обробки, сортування та оптимального розподілу на складі. Вони забезпечують точне визначення ваги та габаритів кожного пакунка, що необхідно для оптимальної обробки та подальшої доставки.

Спеціальні вагові датчики автоматично зчитують вагу кожного пакунка під час його переміщення на конвеєрі. Дані про вагу передаються в центральну систему керування, яка використовує цю інформацію для сортування. Система контролює, щоб вага пакунків відповідала допустимим параметрам для конкретних каналів сортування або пунктів призначення. У разі виявлення відхилень пакунок перенаправляється для додаткової перевірки або обробки. За допомогою лазерних сканерів або ультразвукових сенсорів система вимірює габарити пакунків (довжину, ширину, висоту) під час їх руху по конвеєру. Дані про розміри також надсилаються в систему керування. Габарити пакунка впливають на вибір оптимального маршруту та контейнера для транспортування, що дозволяє раціонально використовувати вантажні відсіки та зменшувати витрати на логістику.

Системи зважування та вимірювання розмірів інтегровані з програмним забезпеченням сортувальної системи, яка на основі цих даних обирає оптимальні параметри сортування для кожного пакунка. Якщо пакунок відповідає всім заданим критеріям, він продовжує шлях до відповідного контейнера або зони відправки. У випадку невідповідності (наприклад, коли вагу або розміри неможливо визначити), пакунок направляється на додаткову перевірку.

Автоматизація процесу зважування та вимірювання розмірів підвищує точність сортування, мінімізує ручну працю та дозволяє зменшити кількість помилок. Це забезпечує більшу швидкість обробки пакунків, що особливо важливо при обробці великих обсягів замовлень у логістичних центрах.

Завдяки системам зважування та вимірювання розмірів сортувальні процеси стають ефективнішими, що сприяє зниженню витрат на логістику, підвищенню якості обслуговування та забезпеченню точності доставки.

1.2.4 Розподільчі механізми (сортувальники)

Механізми Push Tray, Swing Arm та Cross-Belt Sorters автоматично направляють пакунки у відповідні канали або відсіки залежно від їхніх характеристик, зменшуючи потребу в ручному втручанні.

Push Tray Sorter використовує піддони (trays), на яких рухаються пакунки по конвеєру. Коли пакунок досягає потрібного місця для розподілу, піддон здійснює «штовхальний» рух, переміщуючи пакунок у бік визначеного контейнера чи жолоба.

Використовується для сортування пакунків різного розміру та ваги, забезпечуючи точне позиціонування. Ідеально підходить для обробки крихких або нестандартних пакунків, оскільки забезпечує м'яке переміщення.

Swing Arm Sorter працює за принципом «поворотної руки», яка розташована вздовж конвеєра. Коли пакунок проходить повз, механізм «підштовхує» його в сторону до призначеного контейнера або відсіку, здійснюючи рух під певним кутом.

Цей механізм підходить для обробки невеликих і середніх пакунків. Swing Arm Sorter ефективний у системах з високою пропускнуою здатністю, оскільки забезпечує швидке сортування на обмеженому просторі.

Cross-Belt Sorter складається з безлічі рухомих платформ з короткими стрічковими конвеєрами, які перетинають основний конвеєр під прямим кутом. Коли пакунок досягає потрібної зони, невеликий конвеєр на платформі автоматично вмикається і переміщує пакунок у відповідний жолоб або контейнер.

Використовується для точного сортування різних типів пакунків незалежно від їхньої форми, ваги або розміру. Cross-Belt Sorter забезпечує високу швидкість сортування і може обробляти сотні пакунків на годину, що робить його ефективним у великих логістичних центрах.

Кожен механізм забезпечує точне переміщення пакунків у призначені зони сортування.

Механізми дозволяють сортувати великі обсяги пакунків за короткий час, що значно підвищує ефективність.

Вони здатні обробляти пакунки різного розміру, форми та ваги, що робить їх універсальними для логістичних центрів.

Механізми Push Tray, Swing Arm і Cross-Belt Sorters автоматизують та

пришвидшують сортувальний процес, забезпечуючи точний і надійний розподіл пакунків у великих логістичних системах.

1.2.5 Централізовані системи керування WMS і ERP

Програмне забезпечення, яке контролює весь процес сортування, відстежує кожен пакунок у реальному часі, оптимізує розподіл і забезпечує моніторинг всіх операцій, включаючи можливість інтеграції з WMS та ERP.

WMS - це інтелектуальне програмне забезпечення, яке автоматизує і оптимізує всі процеси на складі. Уявіть собі склад як добре змащений годинник, де кожна деталь працює синхронно. WMS – це саме той механізм, який забезпечує цю синхронність.

WMS дозволяє максимально ефективно використовувати кожен квадратний метр складу. Завдяки детальному аналізу розмірів піддонів, обладнання та товарів, система створює оптимальний план розміщення, мінімізуючи порожні простори.

Система WMS це інструмент для точного планування роботи складу. Вона аналізує майбутні замовлення, кількість персоналу та інші фактори, щоб скласти оптимальний графік роботи. Це дозволяє ефективно розподіляти навантаження на працівників і вчасно виконувати всі завдання.

WMS допомагає утримувати оптимальний рівень запасів. Система автоматично відстежує кількість товарів на складі і попереджає про необхідність дозамовлення. Це дозволяє уникнути як дефіциту, так і надлишків товарів, що зменшує витрати на зберігання.

WMS сортує товари за різними критеріями, щоб забезпечити швидкий доступ до потрібних товарів. Наприклад, товари з великим попитом розміщуються ближче до зони відвантаження, а товари з обмеженим терміном зберігання – в спеціально відведених зонах.

WMS автоматизує всі процеси, пов'язані з прийомом і відправкою товарів. Система зчитує штрих-коди, перевіряє дані про товар і генерує необхідну документацію. Це дозволяє мінімізувати помилки і прискорити обробку вантажів.

Завдяки WMS процес комплектації замовлень стає швидким і точним. Система генерує пакувальні листи, в яких вказані всі необхідні товари і їхнє місцезнаходження.

Оптимізація складських процесів за допомогою WMS дозволяє обробляти більший обсяг замовлень і забезпечувати високу швидкість обслуговування клієнтів.

WMS забезпечує постійний моніторинг всіх складських процесів. Система генерує різноманітні звіти, які дозволяють оцінити ефективність роботи складу, виявити вузькі місця і приймати обґрунтовані управлінські рішення.

Система WMS надає додаткові функції, такі як автоматизована інвентаризація для точного підрахунку товарів, аналітика для збору та аналізу даних про складські процеси, що допомагає приймати обґрунтовані рішення. Вона також оптимізує розміщення товарів для зменшення часу на збір замовлень, управляє персоналом шляхом планування роботи та контролю виконання завдань, а також інтегрується з іншими корпоративними системами для безперервного обміну даними.

Використання WMS сприяє підвищенню ефективності роботи складу завдяки швидкій обробці замовлень, мінімізації помилок та оптимальному використанню площі. Система забезпечує точний облік залишків, що виключає втрати, а також підвищує прозорість складських операцій, дозволяючи відстежувати рух товарів у реальному часі. Крім того, WMS допомагає знижувати витрати, оптимізуючи використання ресурсів і скорочуючи потребу в персоналі.

ERP-база даних – це централізоване сховище інформації, яке лежить в основі будь-якої системи планування ресурсів підприємства. Вона об'єднує дані про всі аспекти діяльності компанії, від фінансів і бухгалтерії до виробництва та логістики ERP складається з таких елементів: платформа, управління, даними, модулі. Платформа це основа, на якій будуються всі функції системи. Це як фундамент будинку, що забезпечує роботу всіх модулів

та компонентів. Змінювати код платформи можуть лише розробники, а користувачі та спеціалісти з впровадження не мають доступу до нього. Управління даними це "серце" системи. Тут зберігаються всі дані, які використовує система, і здійснюються всі операції з ними. Уявіть це як величезний склад, де зберігається вся інформація, необхідна для роботи підприємства. Модулі це окремі "цеглинки", з яких будується система. Кожен модуль відповідає за свою частину роботи (наприклад, облік продажів, управління персоналом тощо). Всі модулі працюють разом, використовуючи загальну базу даних і базові функції платформи.

ERP-база даних у логістиці виконує широкий спектр функцій, включаючи управління запасами, де здійснюється детальний облік товарів, прогнозування попиту, оптимізація рівнів запасів та контроль партій товарів. Вона також забезпечує управління транспортом через планування маршрутів, моніторинг транспортних засобів у реальному часі та управління автопарком. Для управління складом ERP-система підтримує реєстрацію та перевірку вхідних товарів, організацію відпуску продукції та проведення інвентаризації. Важливими є і функції управління замовленнями: від прийому й обробки замовлень до відстеження їхнього статусу та управління поверненнями товарів.

ERP-система дозволяє аналізувати ефективність логістичних операцій, оцінюючи продуктивність складів, транспортних засобів і персоналу.

Використання ERP-бази даних забезпечує єдиний інформаційний простір, де вся інформація про компанію зберігається централізовано. Це дозволяє автоматизувати рутинні процеси, звільняючи співробітників від ручної роботи, та підвищити точність даних завдяки зменшенню кількості помилок. Наявність актуальної та достовірної інформації сприяє покращенню прийняття управлінських рішень, а прозорість бізнес-процесів дозволяє відстежувати всі етапи виконання завдань у реальному часі.

Недоліки цього типу програмних продуктів є наслідком їхніх переваг. Єдина база даних і інтегрована система створюють велику кількість взаємозв'язків, що підвищує складність системи та вимоги до апаратного

забезпечення (серверів). Тому для ефективної роботи ERP-системи необхідне потужне обладнання, що призводить до додаткових витрат.

Ще однією поширеною проблемою під час впровадження ERP-систем є забезпечення безпеки даних. Оскільки з системою працюють усі підрозділи та співробітники компанії, необхідно налаштувати індивідуальні права доступу для кожного користувача. Якщо у спеціалізованих програмах зазвичай створюють кілька рівнів доступу (наприклад, для рядових працівників, керівників відділів і топ-менеджерів), то в ERP-системах структура прав доступу є значно складнішою. Потрібно налаштувати доступ до різних модулів для кожного підрозділу, а також створити ієрархію всередині кожного відділу. Така складна конфігурація часто призводить до помилок і потребує додаткового часу для тестування та налаштування.

Крім того, використання єдиної системи з централізованою базою даних має певний ризик. У разі збою в роботі ERP-системи (наприклад, через відключення електроенергії або проблеми з сервером) може зупинитися робота всієї компанії. Тому особливо важливо забезпечити надійну роботу серверного обладнання та своєчасне технічне обслуговування.

ERP-системи також мають загальні недоліки, притаманні всім складним системам: високий рівень складності впровадження, значні витрати на навчання персоналу та тривалий процес адаптації до нових умов роботи.

1.2.6 Системи візуалізації та моніторингу

Екрани та інтерфейси для операторів відображають статус кожного пакунка, час до відправлення і загальну ефективність системи, дозволяючи оперативно втручатися в разі необхідності.

Системи візуалізації та моніторингу в системах керування автоматизованим процесом сортування пакунків виконують важливу функцію контролю і відстеження пакунків у реальному часі. Вони дозволяють операторам отримувати візуальну інформацію про рух пакунків, загальний стан системи та ефективність сортувальних процесів.

Системи візуалізації надають оператору актуальну інформацію про

розташування кожного пакунка на сортувальній лінії. Це дозволяє контролювати процес, вчасно виявляти затримки або інші проблеми та оперативно на них реагувати, забезпечуючи ефективність роботи.

Системи моніторингу відстежують функціонування всіх ключових елементів автоматизації, таких як конвеєри, сенсори, сканери тощо. При виявленні збоїв або перевантажень система надсилає повідомлення або сигнали тривоги, щоб оператори могли швидко втрутитися й усунути проблему, запобігаючи зупинці процесу.

Ці системи також фіксують ключові показники ефективності, включаючи швидкість обробки, кількість відхилених або зупинених пакунків. Це допомагає оцінити загальну продуктивність системи, виявити вузькі місця та внести необхідні налаштування для оптимізації процесу.

На основі зібраних даних система може прогнозувати можливі затримки, пікові навантаження або потребу в технічному обслуговуванні. Це сприяє запобіганню простоїв і дозволяє ефективно планувати ресурси, забезпечуючи безперервну роботу.

Усі дані зберігаються в базі для подальшого аналізу або аудиту. На великих екранах або моніторах відображаються візуальні елементи, такі як графіки, схеми, потоки й статуси кожного конвеєра або сортувального вузла, що забезпечує зручність контролю й управління.

Системи візуалізації та моніторингу забезпечують покращений контроль і прозорість, надаючи операторам швидкий доступ до повної інформації про стан сортувального процесу. Вони сприяють швидкому реагуванню на несправності, знижуючи час простоїв завдяки оперативним сповіщенням. Дані системи також допомагають прогнозувати потреби в обслуговуванні й оптимально розподіляти ресурси, підтримуючи стабільну роботу. Крім того, аналіз продуктивності дозволяє виявляти проблемні ділянки та налаштовувати систему для максимальної ефективності. Системи візуалізації та моніторингу є невід'ємною частиною автоматизованих сортувальних комплексів, оскільки вони забезпечують надійність, прозорість і ефективність усього процесу сортування.

1.3 Вивчення сучасних методів автоматизації процесів сортування пакунків

Процес може дещо відрізнятись залежно від системи, але загальні кроки схожі.

Пакунки розміщуються на конвеєрі або системі подачі, що може здійснюватися вручну або автоматично.

Сканери зчитують етикетки пакунків для отримання ключової інформації, такої як адреса, номер для відстеження, розміри та вага.

Ці дані надсилаються до програмного забезпечення сортувальної системи, яке за допомогою алгоритмів визначає оптимальне місце сортування для кожного пакунка.

Конвеєри доставляють пакунки до зони сортування, де роботизовані маніпулятори переміщують їх у жолоби, що ведуть до потрібних точок сортування.

Відсортовані пакунки спускаються по жолобах у клітки, мішки чи контейнери, звідки їх переміщують до вантажівок або літаків для подальшої доставки.

Протягом усього процесу система відстежує та реєструє рух кожного пакунка для забезпечення відповідального зберігання та надання даних про відстеження в реальному часі. Діагностичні системи фіксують можливі помилки або блокування, які вимагають людського втручання для вирішення.

Посилки з різних місць надходять на сортувальний пункт, звідки їх необхідно доставити до інших місць. Розподіл відбувається у двох випадках: вхідні посилки, вихідні посилки.

Вхідні посилки можуть призначатися як для одного регіону, так і для інших регіонів. Наприклад, у Чхаттісгарху централізованим пунктом для розподілу є Райпур. У Райпурі посилки потрібно сортувати для доставки в той самий регіон або в інші регіони (наприклад, Делі, Мумбаї, Колката тощо).

Вихідні посилки це ті, які оформляються в цьому ж центрі та потребують доставки або в той самий регіон, або в інші регіони.

Після надходження посилки розміщуються на конвеєрній стрічці. Зчитувач штрихкодів зчитує код призначення і передає інструкції комп'ютерній системі про адресу доставки. Комп'ютерна система передає цю адресу призначення в блок управління. Блок управління визначає місце доставки згідно з виконаними умовами.

Якщо умови виконані, роликові механізми отримують команду обертатися за годинниковою або проти годинникової стрілки.

Згідно з номером коробки блок управління вирішує, в якому напрямку обертатиметься ролик (за годинниковою або проти годинникової стрілки). Якщо номер коробки непарний, ролик обертається проти годинникової стрілки, інакше за годинниковою стрілкою, щоб помістити посилки в потрібні коробки.

Якщо умова не виконана, посилка переміщується в невизначену коробку, звідки вона направляється до іншої системи, де процес повториться. Коли штрихкод не зчитується, посилка також направляється в невизначену коробку, звідки її буде передано в іншу систему; для вхідних посилок це буде вихідна система, і навпаки.

Важливо враховувати загальні розміри та вагу пакунків. Кожна технологія сортування розроблена для роботи з певною вагою та формою предметів. Дрібні предмети можуть не працювати ефективно в деяких випадках, а довгі предмети також можуть зіткнутися з проблемами в інших ситуаціях.

Наприклад, система сортування з кульковими роликами добре підходить для різноманітних пакунків. Однак, якщо предмети маленькі та легкі й мають мало точок контакту з сортувальником, вони не будуть відхилятися. Як інтегратор, ми маємо досвід у вирішенні подібних поширених проблем і можемо допомогти обрати правильну технологію.

Найпоширенішим методом, який ми використовуємо для сортування, є використання обміну даними з нашим програмним забезпеченням. Зазвичай,

коли пакет потрапляє в кінцеву систему виконання замовлень, ми ініціюємо введення пакета та надсилаємо запит на дані на сервер доставки (WMS, багатоносійний).

Цей запит містить номер LPN пакета, і ми запитуємо у сервера доставки як етикетку для доставки, так і код сортування. Сервер доставки надсилає нам обидва елементи, і ми наклеюємо етикетку та зберігаємо код сортування в нашій базі даних. Цей код сортування, як правило, є просто текстовою рядком, який надає клієнт.

Коли пакет надходить до сортувальника, ми повторно скануємо LPN, запитуємо нашу базу даних про наданий код сортування і запускаємо відповідні правила логіки сортування, які ми встановили.

1.4 Актуальність портативної системи керування автоматизованим процесом сортування пакунків для малих складів

Стрімкий розвиток електронної комерції спричиняє зростання обсягів відправлень і посилює конкуренцію, що створює низку викликів для малих складів інтернет-магазинів і логістичних компаній. Вони повинні ефективно обробляти дедалі більшу кількість замовлень, забезпечуючи високу точність сортування, оскільки помилки можуть призвести до затримок і незадоволення клієнтів. Крім того, зростають вимоги до відстеження посилок у реальному часі, адже клієнти хочуть мати доступ до актуальної інформації про свої замовлення. Однак малі склади часто стикаються з обмеженнями у площі, ресурсах і персоналі, що ускладнює виконання цих завдань.

Портативна система вирішує ці проблеми завдяки автоматизації процесів сортування, що значно підвищує швидкість обробки посилок навіть при великому обсязі відправлень. Використання сканерів та RFID-технологій мінімізує ризик помилок, забезпечуючи точність сортування. Унікальні

ідентифікатори дозволяють відстежувати посилки на всіх етапах логістичного ланцюга, що покращує контроль доставки. Система також оптимізує використання складського простору шляхом ефективного розміщення товарів і знижує витрати, скорочуючи потребу в ручній праці. Завдяки гнучкості портативна система легко адаптується до різних типів товарів.

Портативні системи сприяють підвищенню конкурентоспроможності завдяки швидкій і точній сортуванню пакунків в межах складу, що дозволяє ефективно конкурувати з більшими компаніями. Крім того, оптимізація операцій і зниження витрат допомагають збільшити прибутковість бізнесу.

Портативна СКАПСП є необхідним інструментом для сучасних малих складів. Вона дозволяє підвищити ефективність роботи, забезпечити високий рівень обслуговування клієнтів та адаптуватися до вимог динамічного ринку електронної комерції.

1.5 Основні задачі та вимоги до портативної системи керування автоматизованим процесом сортування пакунків

Портативна СКАПСП оптимізує логістичні операції, підвищує ефективність складу та мінімізує помилки. Вона забезпечує точну ідентифікацію пакунків за допомогою штрих-кодів, QR-кодів, RFID-міток та технологій розпізнавання зображень. Сортування здійснюється за різними критеріями, такими як розмір, вага, пріоритет і місце призначення, з можливістю формування груп для подальшої обробки. Система контролює рух пакунків на конвеєрах, оперативно виявляє та усуває збої. Завдяки зручному інтерфейсу оператор може налаштовувати параметри та моніторити процес у реальному часі. Інтеграція з іншими системами, такими як WMS і бухгалтерське програмне забезпечення, забезпечує обмін даними та синхронізацію з центральною базою. Система також формує звіти щодо

продуктивності, інвентаризації та помилок сортування.

Обов'язковою вимогою портативної системи керування сортуванням є точне та швидке зчитування різних типів маркувань (штрих-коди, QR-коди, RFID-мітки) на пакунках. Для пакунків без штрих-кодів система повинна мати можливість розпізнавання зображень для ідентифікації товару. Система повинна забезпечувати сортування за різними критеріями (розмір, вага, пріоритет), контролювати рух кожного пакунка та мати інтуїтивно зрозумілий інтерфейс для управління процесом. Важливим є також забезпечення інтеграції з іншими системами управління складом і можливість генерації звітів для аналізу продуктивності. Крім того, портативна система повинна бути ергономічною та зручною у використанні, забезпечуючи мобільність оператора під час виконання сортувальних операцій. Це дозволяє оптимізувати робочий процес та підвищити загальну ефективність сортування без прив'язки до стаціонарного місця.

Технічні вимоги до системи передбачають її мобільність і надійність для роботи в умовах складу, а також швидкодію для обробки інформації у реальному часі. Необхідна наявність потужної батареї для тривалої автономної роботи та захист від зовнішніх впливів, таких як пил, волога та механічні пошкодження. Програмне забезпечення має бути інтуїтивно зрозумілим, гнучким для налаштування під потреби користувача, стабільним і регулярно оновлюватися для підтримки актуальності функціоналу. Для забезпечення ефективної інтеграції в існуючу інфраструктуру підприємства, система повинна підтримувати стандартні протоколи обміну даними.

Безпека є ключовим аспектом роботи системи: необхідно забезпечити захист даних про товари та клієнтів, а також обмежити доступ до системи для авторизованих користувачів. Ергономічні вимоги включають зручне розташування елементів управління, компактність і легкість системи для зручності перенесення. Виконання цих вимог дозволить створити ефективну та надійну систему сортування, що підвищить продуктивність роботи складу та оптимізує логістичні процеси.

2 АНАЛІЗ АЛГОРИТМІВ СИСТЕМ КЕРУВАННЯ АВТОМАТИЗОВАНИМ ПРОЦЕСОМ СОРТУВАННЯ ПАКУНКІВ

2.1 Загальний опис роботи систем керування автоматизованим процесом сортування пакунків

Автоматизовані системи керування процесом сортування пакунків є складними технологічними рішеннями, які забезпечують ефективну організацію, контроль і виконання сортувальних операцій. Вони використовуються на підприємствах логістики, поштових службах і складських центрах для обробки великих обсягів пакунків у мінімальні терміни. Розглянемо основні етапи роботи системи.

2.1.1 Стан очікування

Система знаходиться в стані очікування пакунку. Після отримання пакунків з транспорту, складу або попередніх етапів обробки - Пакунки автоматично або вручну завантажуються на стрічковий конвеєр.

2.1.1.1 Аварія під час роботи конвеєра

Якщо аварія трапилась під час роботи конвеєра (або в стані очікування), система автоматично виявляє зупинку конвеєра (через датчики або інші контрольні механізми). Або оператор вручну активує аварійну зупинку. Вмикається світлові та звукові сигналізації для попередження персоналу про аварійну ситуацію. Відправляється повідомлення на панель управління. Автоматично зупиняється всі механізми, пов'язаних із процесом сортування (системи транспортування, сортувальні механізми тощо). Перекривається живлення конвеєра, для безпеки. Проводиться діагностика системи для виявлення причин зупинки. Проводиться перевірка стану пакунків на конвеєрі (чи не заблоковані вони). Оператор або технічний персонал аналізує причини зупинки (оператор системи проводить візуальний огляд пошкодженої ділянки конвеєра). Оператор системи робить спроби усунення проблеми (усунення блокування, ремонт обладнання, процедура вилучення застряглому пакунку,

тощо). Якщо не вдається усунути проблему і виконати необхідний ремонт обладнання, відправляється запит до сервісної служби. Після усунення проблеми відбувається перевірка всіх систем і механізмів на предмет їхньої готовності до роботи. Упевнитися, що всі пакунки знаходяться у безпечному положенні. Повернення системи в робочий режим (перезапуск конвеєра, пакунки знову починають рухатись по сортувальній лінії). Вимкнення сигналізації, якщо все в порядку. Спостереження за роботою системи після перезапуску для виявлення можливих повторних проблем. Збір даних про причини зупинки для подальшого аналізу та запобігання подібним ситуаціям у майбутньому.

2.1.2 Ідентифікація

Кожен пакунок проходить через сканер, який зчитує штрих-код або інший ідентифікатор. Штрихкоди, QR-коди або RFID-мітки зчитуються автоматизованими сканерами. Отримана інформація звіряється з базою даних системи (WMS/ERP). Якщо ідентифікація успішна, система записує дані про пакунок (розміри, вага, маршрут, статус).

2.1.2.1 Помилка ідентифікації

Якщо не вдається зчитати ідентифікатор (бо він пошкоджений, або відсутній), система здійснює кілька спроб зчитати ідентифікатор з різних ракурсів (якщо вдалось зчитати ідентифікатор, відбувається перехід на наступний етап). Якщо після декількох спроб не вдалось зчитати ідентифікатор, оператору повідомляється про проблему, він відправляє пакунок на спеціальну станцію, де вручну зчитує інформацію (робить візуальний огляд пакунка), щоб визначити, чи є інші способи ідентифікації. Наприклад, штрих-коди на інших частинах пакунка, етикетки або маркування. Якщо є доступ до бази даних, спробувати знайти інформацію про пакунок за іншими параметрами (вага, розміри, тип упаковки). Якщо немає інформації в базі даних – відбувається алгоритм дій з пункту 2.1.3.1 і оператор вводить її в систему (якщо вдалось зчитати ідентифікатор, пакунок повертається на лінію сортування і відбувається перехід на наступний етап). Якщо все одно не вдалось

ідентифікувати пакунок - пакунок тимчасово поміщається в окрему зону для подальшого розгляду (це дозволяє уникнути змішування з іншими пакунками та запобігти помилкам у сортуванні). Після цього створюється запит на додаткову інформацію (надсилається запит відправнику або одержувачу для уточнення інформації про відправлення). Для цього може використовуватися електронна пошта, SMS-повідомлення або інший спосіб зв'язку. Якщо вдалось отримати інформацію, пакунок повертається на лінію сортування і відбувається перехід на наступний етап. Вся інформація про проблему (тип проблеми, час виникнення, дії, які були вжиті) записується в журнал подій для подальшого аналізу.

2.1.3 Передача інформації

Зчитана інформація передається в центральну комп'ютерну систему. Комп'ютер аналізує отримані дані та порівнює їх з інформацією в базі даних (WMS/ERP). Визначаються точна адреса призначення, розміри, вага та інші характеристики пакунка.

2.1.3.1 Проблеми зі співставленням даних

Якщо в базі даних відсутня (або не вистачає) інформація про пакунок, генерується сигнал тривоги. Оператору повідомляється про проблему, він відправляє пакунок на спеціальну станцію, де вручну зчитує інформацію. Він перевіряє інші системи чи бази даних, які можуть містити інформацію про пакунок (наприклад, системи управління запасами або логістики). Якщо інформацію не вдалось знайти, пакунок тимчасово поміщається в окрему зону для подальшого розгляду (це дозволяє уникнути змішування з іншими пакунками та запобігти помилкам у сортуванні). Після цього створюється запит на додаткову інформацію (надсилається запит відправнику або одержувачу для уточнення інформації про відправлення). Для цього може використовуватися електронна пошта, SMS-повідомлення або інший спосіб зв'язку. Якщо вдалось отримати інформацію, пакунок повертається на лінію сортування і відбувається перехід на наступний етап. Вся інформація про проблему записується в журнал подій для подальшого аналізу.

2.1.4 Визначення маршруту

На основі зібраних даних система розраховує оптимальний маршрут для пакунка. Алгоритм визначає оптимальний шлях для пакунка, мінімізуючи час і ресурси. Враховуються такі фактори: відстань до кінцевого пункту, наявність пробок на дорогах (якщо це стосується доставки транспортними засобами), завантаженість різних зон складу, спеціальні вимоги до транспортування (наприклад, температурний режим). Паунок додається до черги обробки, враховуючи пріоритети (терміновість, тип вантажу). Система постійно відстежує стан конвеєрів, сортувального обладнання та інших компонентів системи. Система регулярно оновлює інформацію про маршрути, адреси доставки та пріоритети пакунків.

2.1.4.1 Зміна маршруту

Якщо відбувається зміна маршруту в процесі сортування, оператор системи отримує повідомлення про зміну маршруту і причини такої зміни. Інформація про новий маршрут записується в базу даних. Система використовує алгоритми маршрутизації для розрахунку оптимального нового маршруту з урахуванням поточних умов і нових даних. Система вибирає альтернативний маршрут з урахуванням таких факторів, як відстань, час доставки, наявність вільних ресурсів. Система оцінює, як зміна маршруту одного пакунка вплине на доставку інших пакунків. Система відправляє нову команду на конвеєрну систему для перенаправлення пакунка на новий маршрут. Система автоматично перенаправляє пакунки до нових маршрутів (може бути задіяно механізми зміни напрямку, пересувні конвеєри тощо).

2.1.5 Перевірка стану конвеєра

Якщо конвеєр вільний та працює справно, то відправити команду на рух, інакше – очікувати. Паунок починає рухатись по стрічковому конвеєру або через рольганги. На кожній ключовій точці сканер перевіряє статус пакунка, щоб гарантувати точність маршруту та корегувати маршрут при зміні умов.

Система постійно відстежує стан конвеєра (регулярно проводиться моніторинг стану конвеєрів, датчиків та іншого обладнання). Перевірка всіх

систем і механізмів на предмет їхньої готовності до роботи). Упевнитися, що всі пакунки знаходяться у безпечному положенні за допомогою різних датчиків (температури, вібрації, тиску тощо). При виявленні відхилень від нормальних параметрів, система фіксує аварійну ситуацію .

2.1.5.1 Зупинка конвеєра

Якщо аварія трапилась під час роботи конвеєра (або в стані очікування), система автоматично виявляє зупинку конвеєра (через датчики або інші контрольні механізми). Або оператор вручну активує аварійну зупинку. Вмикається світлові та звукові сигналізації для попередження персоналу про аварійну ситуацію. Відправляється повідомлення на панель управління. Автоматично зупиняється всі механізми, пов'язаних із процесом сортування (системи транспортування, сортувальні механізми тощо). Перекривається живлення конвеєра, для безпеки. Проводиться діагностика системи для виявлення причин зупинки. Проводиться перевірка стану пакунків на конвеєрі (чи не заблоковані вони). Оператор або технічний персонал аналізує причини зупинки (оператор системи проводить візуальний огляд пошкодженої ділянки конвеєра). Оператор системи робить спроби усунення проблеми (усунення блокування, ремонт обладнання, процедура вилучення застряглого пакунка, тощо) Якщо не вдається усунути проблему і виконати необхідний ремонт обладнання, відправляється запит до сервісної служби.

Після усунення проблеми відбувається перевірка всіх систем і механізмів на предмет їхньої готовності до роботи. Упевнитися, що всі пакунки знаходяться у безпечному положенні. Повернення системи в робочий режим (перезапуск конвеєра, пакунки знову починають рухатись по сортувальній ліній). Вимкнення сигналізації, якщо все в порядку. Спостереження за роботою системи після перезапуску для виявлення можливих повторних проблем. Збір даних про причини зупинки для подальшого аналізу та запобігання подібним ситуаціям у майбутньому.

Якщо під час руху пакунка змінюються умови (наприклад, пункт призначення закрито), система перераховує маршрут і коригує рух пакунка.

2.1.6 Перевірка місцезнаходження пакунку

Спеціальні датчики (наприклад, фотоелементи, індуктивні датчики) виявляють наявність пакунка на станції. Для деяких типів пакунків може використовуватися зважування для додаткової перевірки. Відбувається повторна ідентифікація. Сканер підтверджує ідентифікатор пакунка (штрихкод, QR-код або RFID-мітка) для перевірки правильності маршруту. Система звіряє отриманий код із записом у базі, щоб підтвердити, що пакунок знаходиться на правильній станції.

Якщо виникають проблеми з ідентифікацією відбувається алгоритм дій з пункту 2.1.2.1

Якщо виникають проблеми з перевіркою отриманим кодом із записом у базі даних відбувається алгоритм дій з пункту 2.1.3.1

2.1.7 Перевірка сортувальних механізмів.

Якщо сортувальний механізм вільний та працює справно, то відправити команду на рух, інакше – очікувати. Сигнали про необхідність переміщення пакунка передаються на виконавчі сортувальної механізми (конвеєри, відхилячі, штовхачі тощо). Активація сортувального механізму. Пакунки відправляються з конвеєрів на визначені палети і контейнери. Пакунки, що мають спільний пункт призначення, консолідуються в більші контейнери або палети. Це дозволяє оптимізувати використання транспортних засобів.

Система постійно відстежує стан сортувального механізму (регулярно проводиться моніторинг стану сортувальних механізмів, датчиків та іншого обладнання). Перевірка всіх систем і механізмів на предмет їхньої готовності до роботи. Упевнитися, що всі пакунки знаходяться у безпечному положенні за допомогою різних датчиків (температури, вібрації, тиску тощо). При виявленні відхилень від нормальних параметрів, система фіксує аварійну ситуацію. Відбувається алгоритм дій з пункту 2.1.7.1.

2.1.7.1 Несправності сортувальних механізмів

Якщо виникли проблеми з одним з сортувальних механізмів, вмикається світлові та звукові сигналізації для попередження персоналу про аварійну

ситуацію. Відправляється повідомлення на панель управління. Автоматично зупиняється всі механізми, пов'язаних із процесом сортування (системи транспортування, сортувальні механізми тощо). Перекривається живлення конвеєра та сортувальних механізмів, для безпеки. Проводиться діагностика системи для виявлення причин зупинки. Проводиться перевірка стану пакунків на конвеєрі (чи не заблоковані вони). Оператор або технічний персонал аналізує причини зупинки (оператор системи проводить візуальний огляд пошкодженої ділянки сортувального механізму). Оператор системи робить спроби усунення проблеми (усунення блокування, ремонт обладнання, процедура вилучення застряглого пакунка, тощо). Якщо не вдається усунути проблему і виконати необхідний ремонт обладнання, відправляється запит до сервісної служби.

2.2 Критерії СКАПС

Кількість критеріїв, які можуть мати автоматизовані системи сортування пакунків, досить велика і залежить від багатьох факторів, таких як: тип сортувальної системи, розмір і складність логістичного центру, вимоги клієнтів, тип товарів, тощо.

Типові критерії сортування можуть включати: розмір пакунка, вага, форма, адреса доставки, пріоритет доставки, особливі позначки (крихке, горюче, зберігати у прохолодному місці), номер замовлення, термін придатності.

Сучасні АСС можуть використовувати комбінацію цих та інших критеріїв для досягнення максимальної ефективності. Завдяки великій кількості критеріїв сортування, система може швидко та точно визначити маршрут кожного пакунка, що підвищує ефективність роботи складу та задоволеність клієнтів.

Важливо зазначити, що технології постійно розвиваються: з'являються

нові можливості для сортування, наприклад, за кольором, матеріалом, вмістом пакунка (за допомогою рентгенівського сканування). Конкретний набір критеріїв залежить від конкретної системи сортування і її призначення.

2.3 Алгоритм систем керування автоматизованим процесом сортування пакунків на основі WMS

Алгоритм систем керування автоматизованим процесом сортування пакунків на основі WMS залежить від комбінації критеріїв, що аналізуються. До найбільш розповсюджених критеріїв, що є критичними при сортуванні пакунків, відносяться наступні: крихкість вмісту пакунка (X), вага пакунка (Y), його габаритний розмір (Z), вологостійкість (J) вмісту пакунка та оптимальний температурний діапазон зберігання пакунка (T). При такій комбінації критеріїв алгоритм сортування матиме наступну послідовність.

2.3.1 Стан очікування

Усі стрічкові конвеєри (КККК1, КККК2, КККК3, КККК4, КККК5) і поворотні механізми (P1, P2, P3, P4, P5) вимкнені (=0), якщо кнопка запуску К не була натиснута. Система обробки даних постійно перевіряє, чи була натиснута кнопка. Після натискання кнопки система перевіряє, чи працюють стрічкові конвеєри.

Якщо стрічкові конвеєри були вимкнені під час натискання кнопки К, то вони вмикаються (КККК1=КККК2=КККК3=КККК4=КККК5=1). Якщо стрічкові конвеєри були увімкнені під час натискання кнопки К, то вони вимикаються (КККК1=КККК2=КККК3=КККК4=КККК5=0). Мотори, які відповідають за рух стрічок усіх стрічкових конвеєрів, перебувають у стані спокою (ККК1=0, ККК2=0, ККК3=0, ККК4=0, ККК5=0). В системі є всього 6 контрольних пунктів (місця на сортувальній лінії, де відбувається перевірка наявності пакунка та ідентифікація). Контрольний пункт складається з:

поворотного механізму (складається з поворотних роликів (наприклад, P1) та мотора, який рухає стрічку (наприклад, P11)), RFID-сканера (L) та датчиків, які перевіряють наявність пакунка на поворотному механізмі (I). Для запобігання зіткненням між пакунками, стрічковий конвеєр активується тільки тоді, коли на наступному поворотному механізмі немає пакунка. І навпаки, стрічковий конвеєр зупиняється, тільки коли на наступному поворотному механізмі є пакунок

2.3.2 Перевірка наявності пакунка на першому контрольному пункті

Після запуску всіх конвеєрів, система перевіряє на наявність пакунка на першому контрольному пункті (I1).

2.3.2.1 Пакунок відсутній на першому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі першого стрічкового конвеєра ККК1. Якщо діагностика пройшла успішно, вмикається мотор першого конвеєра ККК1. Якщо отримані значення температур не збігаються з визначеними нормами, всі конвеєри автоматично вимикаються. Як тільки пакунок опиниться на першому поворотному механізмі, перший стрічковий конвеєр вимикається ($ККК1=0$).

2.3.2.2 Пакунок присутній на першому поворотному механізмі

У разі знаходження пакунка на поворотному механізмі (сигнал I1=1) активуються RFID-сканери (сигнал L1=1) для зчитування параметрів: крихкості (X), ваги (Y), вологостійкості (J), температури (T) та розмірів (Z). Після завершення зчитування RFID-сканери вимикаються ($L1=0$). Отримані дані порівнюються з відповідними значеннями з бази даних системи управління складом (WMS), позначеними як X1, Y1, J1, T1 та Z1. У випадку ідентичності зчитаних та отриманих з WMS даних ($X=X1, Y=Y1, J=J1, T=T1, Z=Z1$), система обробки даних проводить верифікацію на відповідність заданим критеріям: вага не менше 60 одиниць ($Y \geq 60$), вологостійкість не більше 55 одиниць ($J \leq 55$), температура більше 4 одиниць ($T > 4$), розмір не менше 100 одиниць ($Z \geq 100$). Якщо зчитані параметри не збігаються з даними WMS ($X \neq X1, Y \neq Y1, J \neq J1, T \neq T1, Z \neq Z1$), система здійснює дві додаткові спроби зчитування штрих-коду. У

разі невдалого зчитування після трьох спроб, система генерує повідомлення про помилку та переходить в режим очікування ручного вилучення пакунка (контроль наявності пакунка на контрольному пункті I1). Після вилучення пакунка (сигнал I1=0), система повертається до виконання дій, описаних у пункті 2.3.2.1, де відбувається перевірка першої частини штрих-коду, що відповідає за крихкість (X).

2.3.3 Перевірка наявності пакунку на другому контрольному пункті

При значенні $X = 0$ поворотний механізм продовжує перебувати в стані спокою $P1=0$ (якби був би $X=1$ - $P1=1$). Система перевіряє на наявність пакунку на другому контрольному пункті (I2).

2.3.3.1 Паунок відсутній на другому поворотному механізмі

Якщо паунок відсутній, то відбувається діагностика температури на моторі другого стрічкового конвеєру ККК2. Якщо діагностика пройшла успішно, вмикається мотор другого конвеєра ККК2 та мотор першого поворотного механізму P11. Якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються. Як тільки датчики зареєструють, що паунок залишив перший контрольний пункт - мотор першого поворотного механізму вмикається ($P11=0$). Відбуваються алгоритми дій з пункту 2.3.2.1.

2.3.3.2 Паунок присутній на другому поворотному механізмі

Якщо паунок присутній на поворотному механізмі ($I2=1$), то вмикаються RFID-сканери ($L2=1$). Зчитуються значення крихкості (X), ваги (Y), вологостійкості (J), температури (T) і розмірів (Z). Після цього RFID-сканери вимикаються ($L2=0$). В систему обробки даних вводяться значення, отримані з бази даних WMS ($X1, Y1, J1, T1, Z1$). Якщо $X=X1, Y=Y1, J=J1, T=T1$ та $Z=Z1$, то система обробки даних звіряє отримані дані із заданими сталими значеннями: вага не менше 60 одиниць ($Y \geq 60$), вологостійкість не більше 55 одиниць ($J \leq 55$), температура більше 4 одиниць ($T > 4$), розмір не менше 100 одиниць ($Z \geq 100$). Якщо зчитати значення не вдалося ($X \neq X1, Y \neq Y1, J \neq J1, T \neq T1, Z \neq Z1$), система здійснює ще дві спроби зчитування штрих-коду. Якщо

після трьох спроб зчитати штрих-код все одно не вдається, система виводить повідомлення про помилку та переходить в режим очікування ручного вилучення пакунка (перевірка наявності пакунка на контрольному пункті I2). Після вилучення пакунка (I2=0) виконуються дії, описані у пункті 2.3.2.1. Відбувається перевірка другої частини штрих-коду – ваги (Y).

2.3.4 Перевірка наявності пакунку на третьому контрольному пункті

При значенні $Y \geq 60$ поворотний механізм активується P2=1 (якби був би $Y < 60$ - P2=0). Система перевіряє на наявність пакунку на третьому контрольному пункті (I3).

2.3.4.1 Пакунок відсутній на третьому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі третього стрічкового конвеєру КККЗ. Якщо діагностика пройшла успішно, вмикається мотор третього конвеєра КККЗ та мотор другого поворотного механізму P21. Якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично .каються. Як тільки датчики зареєструють, що пакунок залишив другий контрольний пункт - мотор другого поворотного механізму вимикається (P21=0). Робота поворотного механізму зупиняється P2=0. Відбуваються алгоритми дій з пункту 2.3.3.1.

2.3.4.2 Пакунок присутній на третьому поворотному механізмі

Якщо пакунок присутній на поворотному механізмі (I3=1), то вмикаються RFID-сканери (L3=1). Зчитуються значення крихкості (X), ваги (Y), вологостійкості (J), температури (T) і розмірів (Z). Після цього RFID-сканери вимикаються (L3=0). В систему обробки даних вводяться значення, отримані з бази даних WMS (X1, Y1, J1, T1, Z1). Якщо $X=X1$, $Y=Y1$, $J=J1$, $T=T1$ та $Z=Z1$, то система обробки даних звіряє отримані дані із заданими сталими значеннями: вага не менше 60 одиниць ($Y \geq 60$), вологостійкість не більше 55 одиниць ($J \leq 55$), температура більше 4 одиниць ($T > 4$), розмір не менше 100 одиниць ($Z \geq 100$). Якщо зчитати значення не вдалося ($X \neq X1$, $Y \neq Y1$, $J \neq J1$, $T \neq T1$, $Z \neq Z1$), система здійснює ще дві спроби зчитування штрих-коду. Якщо після трьох спроб зчитати штрих-код все одно не вдається, система виводить

повідомлення про помилку та переходить в режим очікування ручного вилучення пакунка (перевірка наявності пакунка на контрольному пункті I3). Після вилучення пакунка (I3=0) виконуються дії, описані у пункті 2.3.3.1. Відбувається перевірка третьої частини штрих-коду – розміру (Z).

2.3.5 Перевірка наявності пакунку на четвертому контрольному пункті

При значенні $Z > 100$ поворотний механізм активується P3=1 (якби був би $Z < 60$ - P3=0). Система перевіряє на наявність пакунку на четвертому контрольному пункті (I4).

2.3.5.1 Пакунок відсутній на четвертому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі четвертого стрічкового конвеєру ККК4. Якщо діагностика пройшла успішно, вмикається мотор третього конвеєра ККК3 та мотор третього поворотного механізму P31. Якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються. Як тільки датчики зареєструють, що пакунок залишив третій контрольний пункт - мотор третього поворотного механізму вимикається (P31=0). Робота поворотного механізму зупиняється P3=0. Відбуваються алгоритми дій з пункту 2.3.4.1.

2.3.5.2 Пакунок присутній на четвертому поворотному механізмі

У разі фіксації присутності пакунка на поворотному механізмі (сигнал I4=1), відбувається активація RFID-сканерів (L4=1). Здійснюється зчитування наступних параметрів: крихкості (X), ваги (Y), вологостійкості (J), температури (T) та розмірів (Z). Після завершення процесу зчитування RFID-сканери деактивуються (L4=0). Отримані значення передаються до системи обробки даних, де відбувається їх порівняння з відповідними даними, отриманими з бази даних системи управління складом (WMS) – X1, Y1, J1, T1 та Z1 відповідно. У випадку збігу всіх зчитаних параметрів з даними WMS ($X=X1$, $Y=Y1$, $J=J1$, $T=T1$ та $Z=Z1$), система обробки даних проводить додаткову перевірку на відповідність пакунка встановленим критеріям: вага повинна бути не менше 60 одиниць ($Y \geq 60$), вологостійкість – не більше 55 одиниць ($J \leq 55$), температура

– більше 4 одиниць ($T > 4$), а розмір – не менше 100 одиниць ($Z \geq 100$). У разі розбіжності хоча б одного з параметрів між зчитаними та даними WMS ($X \neq X1$, $Y \neq Y1$, $J \neq J1$, $T \neq T1$, $Z \neq Z1$), система ініціює дві додаткові спроби зчитування штрих-коду. Якщо після трьох спроб зчитування штрих-коду результат залишається негативним, система генерує повідомлення про помилку та переходить у стан очікування ручного вилучення пакунка з поворотного механізму (контроль здійснюється шляхом перевірки стану сигналу $I4$). Після фізичного вилучення пакунка (сигнал $I4=0$) система повертається до виконання дій, описаних у пункті 2.3.4.1, де відбувається перевірка четвертої частини штрих-коду, що відповідає параметру вологостійкості (J).

2.3.6 Перевірка наявності пакунка на п'ятому контрольному пункті

При значенні $J > 55$ поворотний механізм активується $P4=1$ (якби був би $J < 55$ - $P4=0$). Система перевіряє на наявність пакунка на п'ятому контрольному пункті ($I5$).

2.3.6.1 Пакунок відсутній на п'ятому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі четвертого стрічкового конвеєру ККК5. Якщо діагностика пройшла успішно, вмикається мотор четвертого конвеєра ККК4 та мотор четвертого поворотного механізму $P41$. Якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються. Як тільки датчики зареєструють, що пакунок залишив четвертий контрольний пункт - мотор четвертого поворотного механізму вимикається ($P41=0$). Робота поворотного механізму зупиняється $P4=0$. Відбуваються алгоритми дій з пункту 2.3.5.1.

2.3.6.2 Пакунок присутній на п'ятому поворотному механізмі

У разі фіксації присутності пакунка на поворотному механізмі (сигнал $I5=1$), відбувається активація RFID-сканерів ($L5=1$). Здійснюється зчитування наступних параметрів: крихкості (X), ваги (Y), вологостійкості (J), температури (T) та розмірів (Z). Після завершення зчитування RFID-сканери деактивуються ($L5=0$). Отримані значення передаються до системи обробки даних, де

відбувається їх порівняння з відповідними даними, отриманими з бази даних системи управління складом (WMS) – X_1 , Y_1 , J_1 , T_1 та Z_1 відповідно. У випадку повної ідентичності зчитаних параметрів та даних WMS ($X=X_1$, $Y=Y_1$, $J=J_1$, $T=T_1$ та $Z=Z_1$), система обробки даних проводить додаткову верифікацію на відповідність пакунка встановленим критеріям: вага повинна бути не менше 60 одиниць ($Y \geq 60$), вологостійкість – не більше 55 одиниць ($J \leq 55$), температура – більше 4 одиниць ($T > 4$), а розмір – не менше 100 одиниць ($Z \geq 100$). У разі розбіжності хоча б одного з параметрів між зчитаними значеннями та даними WMS ($X \neq X_1$, $Y \neq Y_1$, $J \neq J_1$, $T \neq T_1$, $Z \neq Z_1$), система ініціює дві додаткові спроби зчитування штрих-коду. Якщо після трьох спроб зчитування штрих-коду результат залишається негативним, система генерує повідомлення про помилку з текстом «Помилка ідентифікації» та переходить у стан очікування ручного вилучення пакунка з поворотного механізму (контроль здійснюється шляхом перевірки стану сигналу I_5). Після фізичного вилучення пакунка з поворотного механізму (сигнал $I_5=0$) система повертається до виконання дій, описаних у пункті 2.3.5.1, де відбувається перевірка четвертої частини штрих-коду, що відповідає параметру температури (T).

2.3.7 Перевірка наявності пакунка на шостому контрольному пункті

При значенні $T > 4$ поворотний механізм активується $P_5=1$ (якби був би $T < 4$ $P_5=0$). Система перевіряє на наявність пакунка на шостому контрольному пункті (I_6).

2.3.7.1 Пакунок відсутній на шостому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі шостого стрічкового конвеєру ККК6. Якщо діагностика пройшла успішно, вмикається мотор п'ятого конвеєра ККК5 та мотор п'ятого поворотного механізму P_51 (якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються). Як тільки датчики зареєструють, що пакунок залишив п'ятий контрольний пункт - мотор п'ятого поворотного механізму вимикається ($P_51=0$). Робота поворотного механізму зупиняється $P_5=0$. Відбуваються алгоритми дій з пункту 2.3.6.1.

2.3.7.2 Паунок присутній на шостому поворотному механізмі

Якщо паунок присутній на поворотному механізмі (сигнал $l_6=1$), то система переходить у стан очікування. Система безперервно контролює наявність паунка на контрольному пункті l_6 (перевіряє стан сигналу l_6). Очікування триває до моменту ручного вилучення паунка. Після того, як паунок буде вилучено (сигнал $l_6=0$), система відновлює виконання дій, описаних у пункті 2.3.7.1.

2.4 Недоліки систем керування автоматизованим процесом сортування паунків на основі WMS та ERP

ACC, що працюють на базі WMS, хоча й підвищують ефективність роботи складів, мають ряд суттєвих недоліків. Складне обладнання, програмне забезпечення та необхідність висококваліфікованого персоналу для обслуговування суттєво підвищують початкові та поточні витрати. Крім того, така система вимагає значних інвестицій в енергоресурси та потребує регулярного технічного обслуговування. Для невеликих складів впровадження WMS може бути економічно невиправданим, оскільки вартість системи може перевищувати вигоду від її використання. Обмежена гнучкість є ще одним недоліком. Зміна конфігурації системи може бути трудомістким і дорогим процесом, що ускладнює адаптацію до змін у бізнес-процесах або асортименті товарів. Крім того, збої в роботі обладнання можуть призвести до тривалих простоїв і значних збитків. Якщо постачальник припиняє підтримку програмного забезпечення, це може стати критичною проблемою.

Залежність від електроенергії та ризику кібератак також є важливими факторами. Відключення електроенергії може паралізувати роботу всього складу, а хакерські атаки можуть призвести до втрати даних та порушення роботи системи. WMS часто працює в хмарі або вимагає постійного

підключення до мережі. Проблеми з інтернетом можуть порушити роботу складу.

Впровадження WMS може вимагати значних зусиль для інтеграції з іншими системами управління підприємством та мати високі вимоги до кваліфікації персоналу. Для обслуговування та управління системою потрібні фахівці з відповідною кваліфікацією.

3 ДОСЛІДЖЕННЯ АЛГОРИТМУ СИСТЕМ СОРТУВАННЯ НА ОСНОВІ ПРОГРАМОВАНОЇ ЛОГІЧНОЇ ІНТЕГРАЛЬНОЇ СХЕМИ

3.1 Огляд АСС на основі мікросхем

Автоматизовані системи сортування, що працюють не на складних WMS-системах, а на базі простіших мікросхем, мають свої унікальні переваги. Їх можна створювати під замовлення, адаптуючи до конкретних вимог.

Мікросхеми, спеціально розроблені для конкретних завдань, дозволяють створити гнучку та масштабовану систему, яка ідеально відповідає унікальним потребам. На відміну від стандартних WMS, мікросхеми можна налаштувати під нестандартні критерії, такі як контроль специфічних сенсорів, нестандартні алгоритми сортування чи унікальні протоколи обміну даними. Це забезпечує оптимальну обробку будь-якого типу товарів, сигналів або функцій, а також дозволяє легко додавати нові можливості без необхідності

Такі системи часто дешевші у розробці та впровадженні порівняно з великими WMS-системами. ПЛІС не вимагають придбання дорогого програмного забезпечення чи оновлень ліцензій. Простішим системам, створеним на базі мікросхем, потрібно значно менше енергії, ніж серверам, які працюють з WMS.

Простіші системи мають обмежений набір функцій, що зменшує ризик помилок або конфліктів. На відміну від WMS, які часто працюють на серверах

із підключенням до інтернету (це забезпечує автономність і незалежність від хмарних сервісів чи центрального серверу), автономні системи на мікросхемах менш вразливі до кібератак. Апаратна система (hardware) зазвичай витриваліша до умов, що можуть призводити до відмов (наприклад, температурні перепади чи вібрація).

Системи на основі простих мікросхем вирізняються швидкістю впровадження, оскільки не потребують складних програмних інтеграцій. Їх експлуатація також вимагає менше ресурсів, адже для обслуговування таких систем потрібно менше технічних спеціалістів. Крім того, модернізація цих систем є більш гнучкою та економічно вигідною, оскільки внесення змін до мікросхем або логіки роботи не вимагає значних витрат, на відміну від складних систем.

Через відсутність зайвих проміжних рівнів програмного забезпечення, мікросхеми обробляють сигнали практично миттєво. Замість складної логіки з базами даних, мікросхеми реалізують пряме управління компонентами системи (наприклад, сортувальними лініями, сенсорами, двигунами).

Такі системи легко інтегруються з сенсорами для збору даних (температура, вологість, рух, об'єми). Деякі сучасні мікросхеми (наприклад, FPGA чи ASIC) можуть містити елементи, які дозволяють виконувати базові функції машинного навчання.

Ці системи легко інтегруються з різноманітними сенсорами, забезпечуючи збір даних про температуру, вологість, рух, об'єми тощо. Більше того, сучасні мікросхеми, такі як FPGA чи ASIC, можуть містити спеціалізовані елементи, які дозволяють виконувати базові функції машинного навчання прямо на пристрої, що відкриває нові можливості для аналізу даних в реальному часі та прийняття більш інтелектуальних рішень. Системи на основі простих ПЛІС знаходять широке застосування в різних галузях. Наприклад, вони використовуються для автоматизації сортувальних ліній на складах, де логічні алгоритми керують процесом розподілу товарів. У виробництві такі системи забезпечують контроль над робочими процесами, виконуючи задані

алгоритми без необхідності складних інтеграцій. У сфері безпеки вони використовуються для контролю доступу та обробки сигналів від камер і сенсорів, створюючи автономні системи безпеки. Крім того, прості мікросхеми знаходять застосування в транспортних системах, де вони керують рухом конвеєрів, ліфтів та транспортерів. Автоматизовані системи на основі ПЛІС під замовлення є чудовим варіантом для специфічних завдань, які не потребують складного функціоналу WMS. Вони економічно ефективні, надійні, гнучкі та прості у використанні. Це робить їх ідеальним рішенням для багатьох сфер, де ключовими є швидкість, автономність та адаптація під конкретні умови.

3.2 Алгоритм роботи систем сортування на основі ПЛІС

В рамках дослідження АСС буде використано критерії крихкості (X), ваги (Y) та розміру (Z) пакунка, оскільки вони є ключовими для забезпечення безпечного транспортування, точного розміщення товарів на складі та оптимізації використання складських площ. Ці критерії є загальноприйнятими в логістиці та дозволяють досягти високої ефективності роботи складу. Розглянемо одну гілку руху пакунку див. рис 3.1.

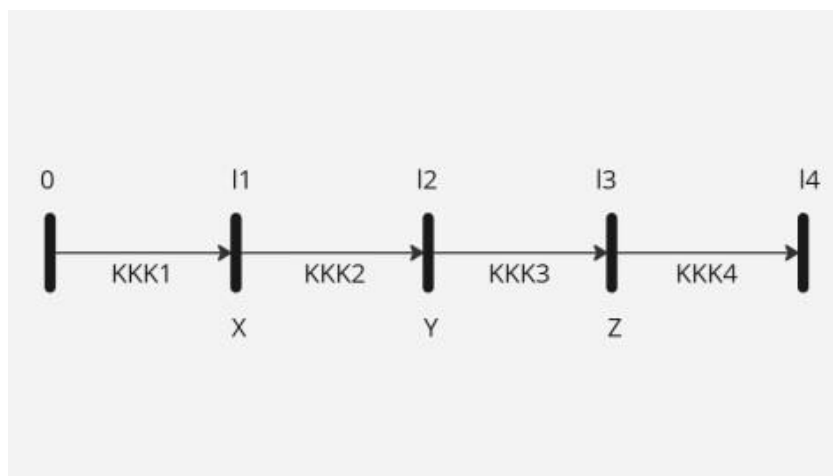


Рисунок 3.1- Рух пакунку по гілці

3.2.1 Стан очікування

Всі стрічкові конвеєри (КККК1, КККК2, КККК3, КККК4) і поворотні механізми (конвеєри) (P1, P2, P3, P4) вимкнені (=0), якщо кнопка запуску К не була натиснута. Система обробки даних завжди перевіряє, чи була натиснута кнопка. Після натискання на кнопку, система перевіряє, чи працюють стрічкові конвеєри. Якщо стрічкові конвеєри були вимкнені під час натискання кнопки К, то вони вмикаються: $КККК1=КККК2=КККК3=КККК4=1$ (якщо стрічкові конвеєри були увімкнені під час натискання кнопки К, то вони вимикаються: $КККК1=КККК2=КККК3=КККК4=0$). Мотори, які відповідають за рух стрічок всіх стрічкових конвеєрів, перебувають у стані спокою ($ККК1=0$, $ККК2=0$, $ККК3=0$, $ККК4=0$).

В системі є всього чотири контрольних пункти (місця на сортувальній лінії, на яких відбувається перевірка наявності пакунка та ідентифікація). Контрольний пункт складається з: поворотного механізму (складається з поворотних роликів (наприклад, P1) та мотора, який рухає стрічку (наприклад, P11)), RFID-сканера (L) і датчиків, які перевіряють наявність пакунка на поворотному механізмі. Задля запобігання зіткнень між пакунками, стрічковий конвеєр активується тільки тоді, коли на наступному поворотному механізмі немає пакунка. І навпаки, стрічковий конвеєр зупиняється тільки тоді, коли на наступному поворотному механізмі є пакунок.

3.2.2 Перевірка наявності пакунка на першому контрольному пункті

Після запуску всіх конвеєрів система перевіряє наявність пакунка на першому контрольному пункті (I1).

3.2.2.1 Пакунок відсутній на першому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі першого стрічкового конвеєра ККК1. Якщо діагностика пройшла успішно, вмикається мотор першого конвеєра ККК1 (якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються). Як тільки пакунок опиниться на першому поворотному механізмі, перший стрічковий конвеєр вмикається ($ККК1 = 0$).

3.2.2.2 Паунок присутній на першому поворотному механізмі

Якщо вже паунок присутній на поворотному механізмі ($l1=1$), то вмикаються RFID-сканери ($L1=1$). Зчитується значення крихкості (X), ваги (Y) і розмірів (Z). RFID-сканери ($L1=1$) вимикаються. Якщо не вдалося зчитати значення, система робить ще дві спроби зчитати штрихкод. Якщо все одно не вдалося зчитати, система виводить помилку «Помилка ідентифікації». Система чекає (перевіряє наявність пакунка на контрольному пункті $l1$) ручного вилучення пакунка. Після вилучення пакунка ($l1=0$) відбуваються дії з пункту 3.2.2.1. Відбувається перевірка першої частини штрихкоду - крихкості (X).

3.2.3 Перевірка наявності пакунка на другому контрольному пункті

При значенні $X = 0$ поворотний механізм продовжує перебувати в стані спокою $P1=0$ (якби було $X=1$, то $P1=1$). Система перевіряє наявність пакунка на другому контрольному пункті ($l2$).

3.2.3.1 Паунок відсутній на другому поворотному механізмі

Якщо паунок відсутній, то відбувається діагностика температури на моторі другого стрічкового конвеєра ККК2. Якщо діагностика пройшла успішно, вмикається мотор другого конвеєра ККК2 та мотор першого поворотного механізму $P11$. Якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються. Як тільки датчики зареєструють, що паунок залишив перший контрольний пункт, мотор першого поворотного механізму вимикається ($P11=0$). Відбуваються алгоритми дій з пункту 3.2.2.1.

3.2.3.2 Паунок присутній на другому поворотному механізмі

Якщо вже паунок присутній на поворотному механізмі ($l2=1$), то вмикаються RFID-сканери ($L2=1$). Зчитується значення крихкості (X), ваги (Y) і розмірів (Z). RFID-сканери ($L2=1$) вимикаються. Якщо не вдалося зчитати значення, система робить ще дві спроби зчитати штрихкод. Якщо все одно не вдалося зчитати, система виводить помилку «Помилка ідентифікації». Система чекає (перевіряє наявність пакунка на контрольному пункті $l1$) ручного вилучення пакунка. Відбувається перевірка другої частини штрихкоду – ваги.

3.2.4 Перевірка наявності пакунку на третьому контрольному пункті

При значенні $Y \geq 60$ поворотний механізм активується ($P2=1$). Якщо $Y < 60$, то $P2=0$. Система перевіряє наявність пакунка на третьому контрольному пункті (13).

3.2.4.1 Пакунок відсутній на третьому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі третього стрічкового конвеєра ККК3. Якщо діагностика пройшла успішно, вмикається мотор третього конвеєра ККК3 та мотор другого поворотного механізму P21. Якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються. Як тільки датчики зареєструють, що пакунок залишив другий контрольний пункт, мотор другого поворотного механізму вмикається ($P21=0$). Робота поворотного механізму зупиняється ($P2=0$). Відбуваються алгоритми дій з пункту 3.2.3.1.

3.2.4.2 Пакунок присутній на третьому поворотному механізмі

Якщо вже пакунок присутній на поворотному механізмі ($I3=1$), то вмикаються RFID-сканери ($L3=1$). Зчитується значення крихкості (X), ваги (Y) і розмірів (Z). RFID-сканери ($L3=1$) вимикаються. Якщо не вдалося зчитати значення, система робить ще дві спроби зчитати штрихкод. Якщо все одно не вдалося зчитати, система виводить помилку «Помилка ідентифікації». Система чекає (перевіряє наявність пакунка на контрольному пункті 13) ручного вилучення пакунка. Після вилучення пакунка ($I3=0$) відбуваються дії з пункту 3.2.3.1). Відбувається перевірка третьої частини штрихкоду - розміру (Z).

3.2.5 Перевірка наявності пакунку на четвертому контрольному пункті

При значенні $Z \geq 100$ поворотний механізм активується ($P3=1$). Якщо $Z < 100$, то $P3=0$. Система перевіряє наявність пакунка на четвертому контрольному пункті (14).

3.2.5.1 Пакунок відсутній на четвертому поворотному механізмі

Якщо пакунок відсутній, то відбувається діагностика температури на моторі четвертого стрічкового конвеєра ККК4. Якщо діагностика пройшла

успішно, вмикається мотор третього конвеєра ККК3 та мотор третього поворотного механізму P31. Якщо отримані значення температур не сходяться з визначеними нормами, всі конвеєри автоматично вимикаються. Як тільки датчики зареєструють, що пакунок залишив третій контрольний пункт, мотор третього поворотного механізму вмикається ($P31=0$). Робота поворотного механізму зупиняється ($P3=0$). Відбуваються алгоритми дій з пункту 3.2.4.1.

3.2.5.2 Пакунок присутній на третьому поворотному механізмі

Якщо вже пакунок присутній на поворотному механізмі ($I4=1$), то система переходить у стан очікування. Система чекає (перевіряє наявність пакунка на контрольному пункті I4) ручного вилучення пакунка. Після вилучення пакунка ($I4=0$) відбуваються дії з пункту 3.2.5.1.

3.3 Структура та компоненти алгоритму систем сортування на основі ПЛС

В результаті аналізу процесу сортування пакунків було визначено, що в алгоритмі спостерігається використання ряду однотипних послідовностей дій, які можна реалізувати як окремі блоки при проектуванні.

В результаті проектування було виділено такі блоки :

- ККК1, ККК2, ККК,3, ККК4;
- $A_n=x_n$;
- L_n ;
- I_n ;
- i_n ;
- помилка ідентифікації;
- P_n0 ;
- P_n1 ;
- $P3=0$;

- P3=1;
- P_n=0, P_{n1}=0;

При цьому структура алгоритму системи сортування на основі ПЛС наступна див. рис 3.1.

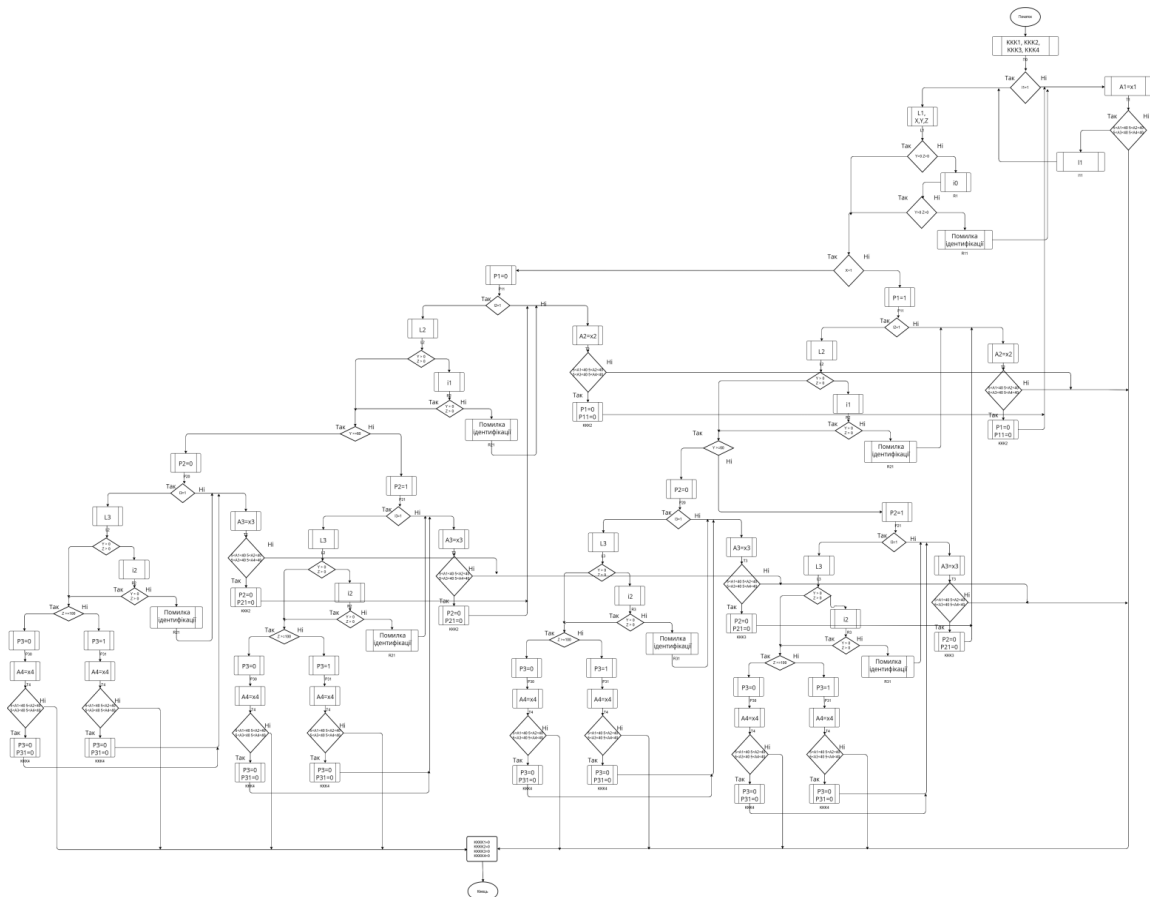


Рисунок 3.1 - Блок схема алгоритму

Спочатку всі конвеєри та поворотні механізми знаходяться у вимкненому стані. При натисканні кнопки К система змінює стан стрічкових конвеєрів на протилежний. Мотори конвеєрів запускаються або зупиняються залежно від їхнього поточного стану.

При натисканні кнопки К система перевіряє стан конвеєрів. Якщо вони були вимкнені, то вмикаються (КККК1-КККК4=1), а якщо були увімкнені, то вимикаються (КККК1-КККК4=0) див. рис 3.2. Цей принцип роботи забезпечує простий та ефективний спосіб керування конвеєрною системою, дозволяючи оператору одним натисканням змінювати режим її роботи.

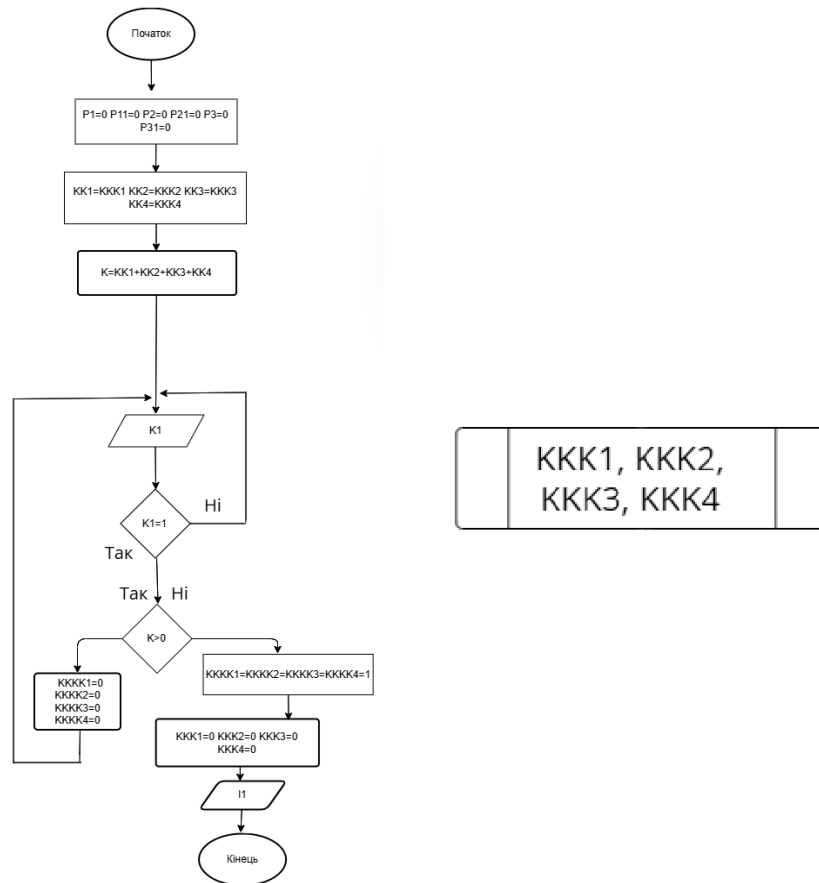


Рисунок 3.2- Алгоритм блоку ККК1, ККК2, ККК,3, ККК4

У разі відсутності пакунка ($I_n=0$) на вході системи проводиться діагностика ($A_n=x_n$) температурного режиму ($5 < A_1 < 40$, $5 < A_2 < 40$, $5 < A_3 < 40$, $5 < A_4 < 40$) двигуна всіх конвеєрів ККК n див. рис 3.3.

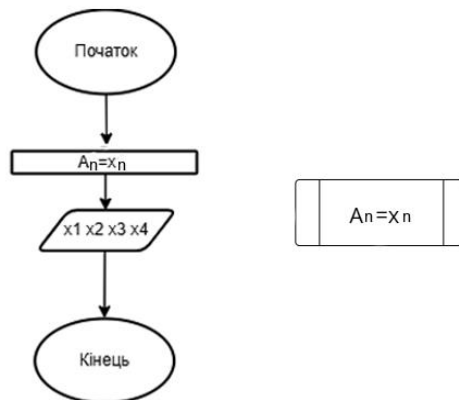


Рисунок 3.3 – Алгоритм блоку $A_n=x_n$

За умови успішного проходження діагностики запускається двигун конвеєра ($ККК_n=1$). Після того, як пакунок досягає першого поворотного механізму, двигун першого конвеєра зупиняється ($ККК_n=0$) див. рис 3.4.

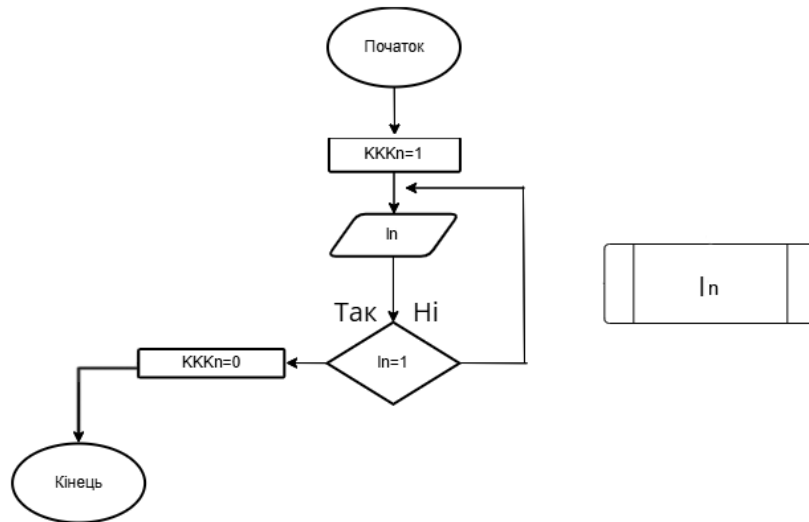


Рисунок 3.4 – Алгоритм блоку In_i

При виявленні пакунка на поворотному ($L_n=1$) механізмі номер n активуються RFID-сканери цього механізму. Відбувається зчитування значень крихкості (X), ваги (Y) та розмірів (Z) пакунка. Після зчитування сканери відключаються див. рис 3.5.

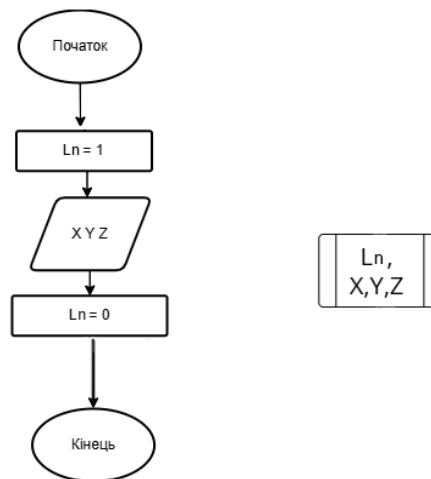


Рисунок 3.5 – Алгоритм блоку L_n

У разі невдалої першої спроби (i_n) зчитування штрихкоду система здійснює ще дві спроби ідентифікації товару див. рис 3.6.

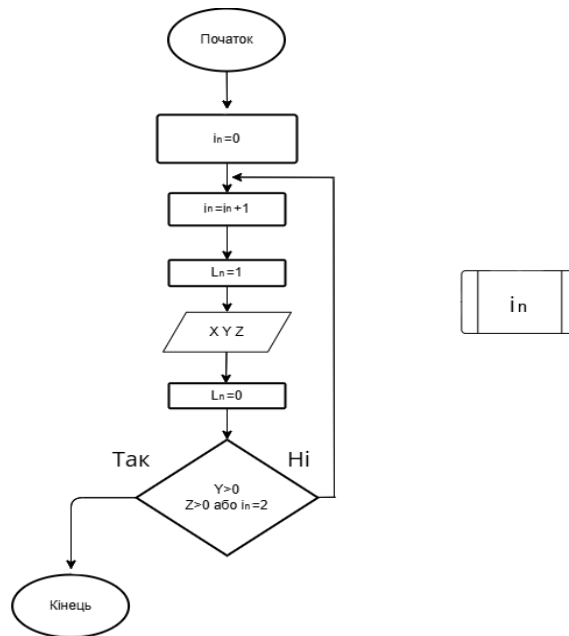


Рисунок 3.6 – Алгоритм блоку i_n

У разі невдалого зчитування штрихкоду після всіх спроб система генерує повідомлення про помилку ідентифікації та переходить у режим очікування ручного втручання оператора для видалення пакунка з контрольного пункту i_n див. рис 3.7.

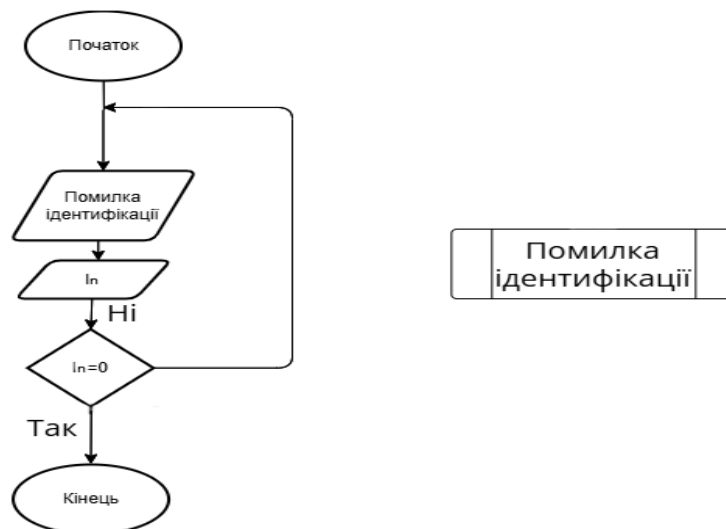


Рисунок 3.7 – Алгоритм блоку «Помилка ідентифікації»

В залежності від значень X та Y , відбувається запуск відповідних поворотних механізмів (P_n) здійснюється при виявленні пакунка на відповідному контрольному пункті див. рис 3.8 та рис 3.9.

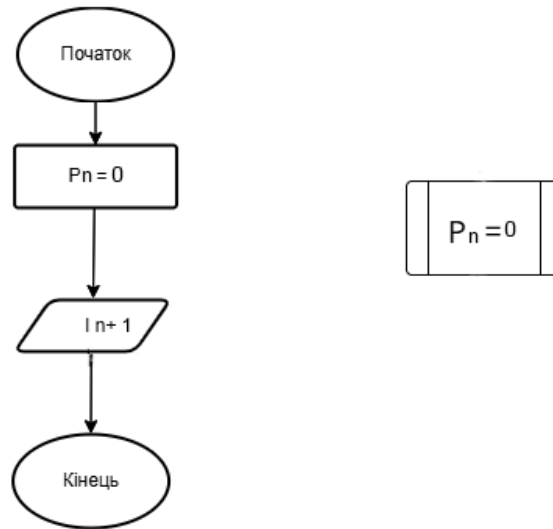


Рисунок 3.8 – Алгоритм блоку Pn0

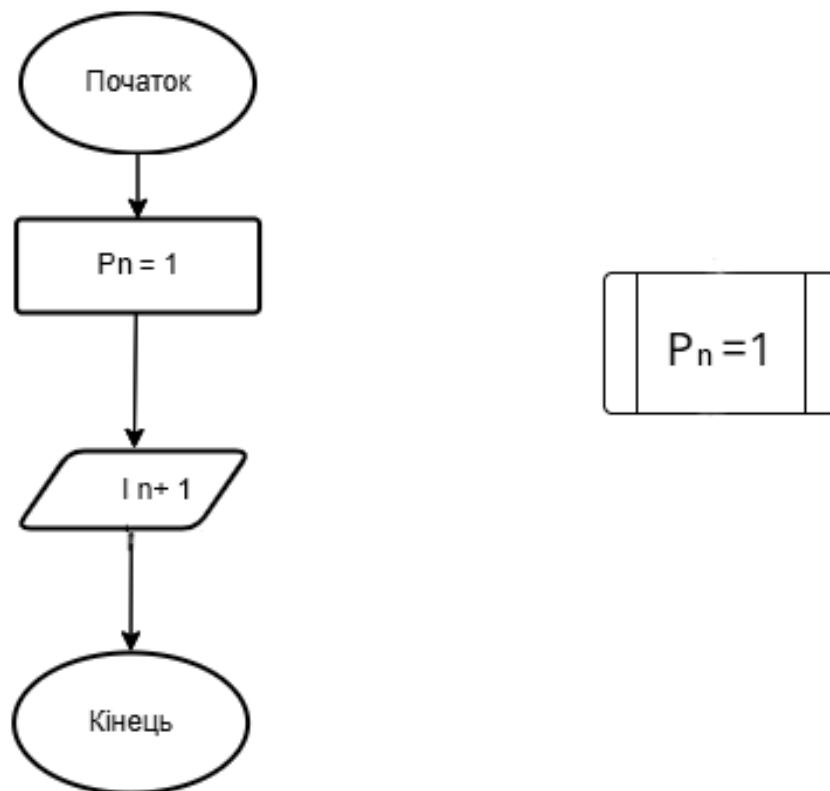


Рисунок 3.9 - Алгоритм блоку Pn1

При значенні $Z \geq 100$ (або $Z \leq 100$) поворотний механізм активується, $P3 = 1$ (якби було $Z < 60$ - $P3 = 0$). Система перевіряє наявність пакунку на четвертому контрольному пункті І4 див. рис 3.10 та 3.11.

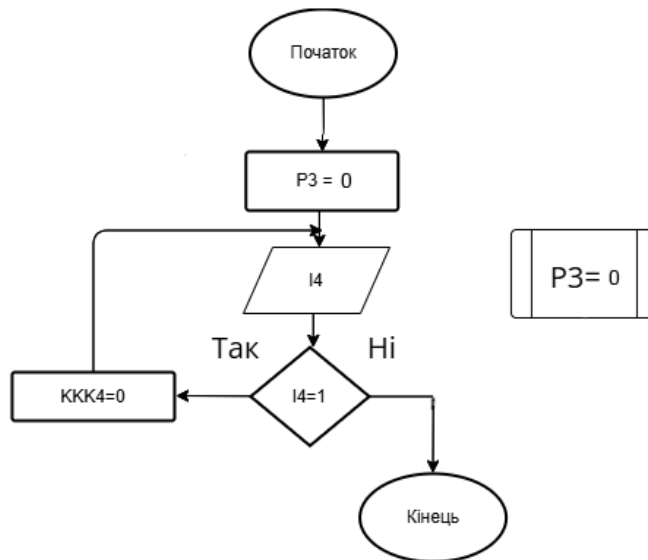


Рисунок 3.10 - Алгоритм блоку P3=0

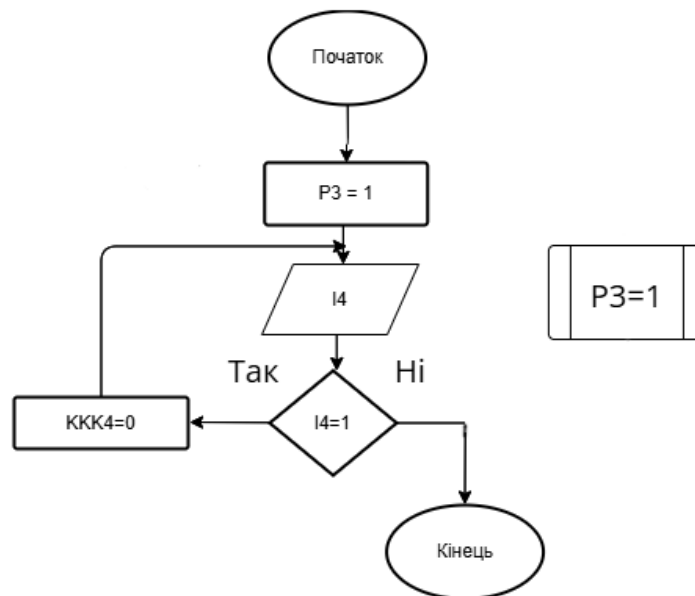


Рисунок 3.11 - Алгоритм блоку P3=1

Система перевіряє, чи залишив пакунок другий поворотний механізм див. рис 3.12.

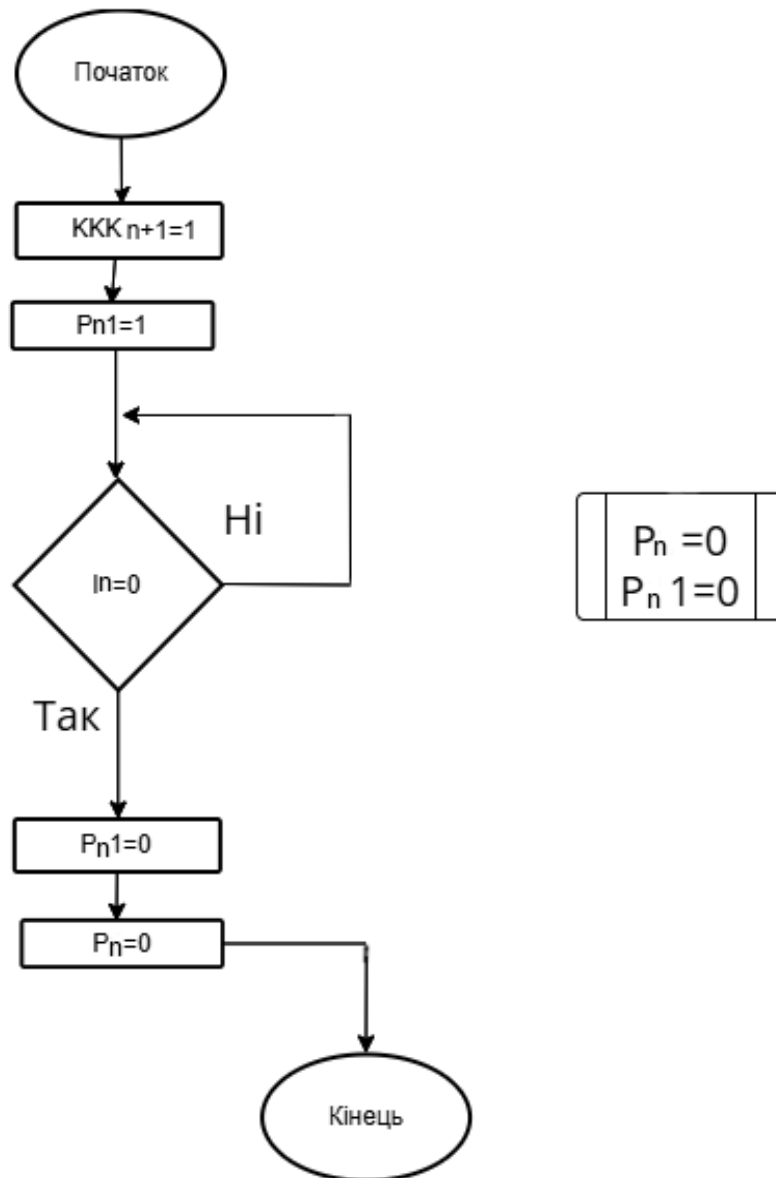


Рисунок 3.12 – Алгоритм блоку $P_n=0$, $P_{n1}=0$

За результатами проектування алгоритму структури та компонентів алгоритму розроблено модульну блокову систему алгоритму сортування, блок-схема якої наведена на рисунку 3.1.

Використання даної блокової структури дозволяє спростити процес масштабування системи сортування до необхідних розмірів шляхом комбінації елементарних блоків зі своїми підпрограмами, модернізуючи лише систему загального керування. Це дозволяє скоротити часові витрати на етапі проектування шляхом використання однотипних блоків.

4 ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА

4.1 Аналіз результатів експерименту та оцінка ефективності системи сортування на основі ПЛІС

Розроблена автоматизована система сортування на основі блочної структури характеризується однотипністю послідовності дій. Тому для перевірки працездатності системи в цілому достатньо переконатися в коректності роботи однієї з гілок цієї системи. Крихкість, вага та розмір є ключовими критеріями для автоматизованих систем сортування, оскільки вони визначають необхідність обережного поводження з пакунками для запобігання пошкодженню, впливають на ефективність транспортування та використання простору, визначають вибір відповідного методу сортування, забезпечують безпеку працівників та обладнання, а також використовуються для розрахунку вартості доставки. Ці фізичні характеристики є фундаментальними для оптимізації всіх етапів логістичного процесу. Крім того, інформація про крихкість, вагу та розмір дозволяє автоматично розподіляти пакунки за відповідними маршрутами та транспортними засобами, враховуючи їхні габарити та допустиме навантаження. Це, в свою чергу, сприяє мінімізації витрат на транспортування та оптимізації логістичних витрат в цілому. Наявність точної інформації про фізичні параметри вантажу також важлива для формування оптимальних палет та контейнерів, що підвищує ефективність використання складського простору та знижує ризики пошкодження під час зберігання.

З метою оптимізації процесу тестування та верифікації розробленої автоматизованої системи сортування, що складається з восьми ідентичних гілок, було застосовано принцип модульності. Тому в рамках дослідження розробленої автоматизованої системи сортування для перевірки працездатності проведено аналіз однієї з восьми гілок, що здійснює класифікацію пакунків за трьома параметрами: крихкість (X), вага (Y) та розмір (Z), див. рис. 4.1.

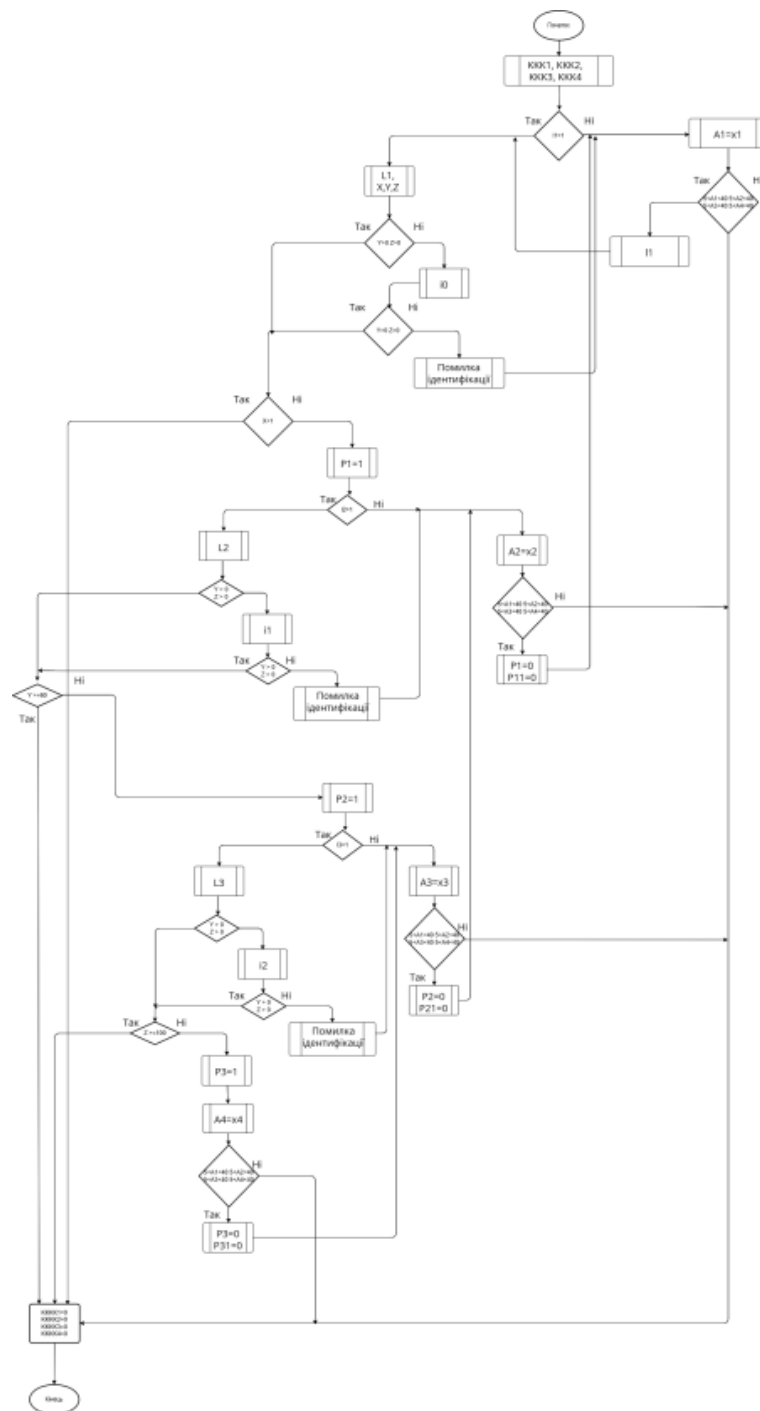


Рисунок 4.1 – Граф-схема одної з гілок СКАПСП

Для забезпечення високої швидкості обробки даних, точності рішень та адаптивності системи сортування до змін, оптимальним є використання програмованих логічних інтегральних схем. Для керування системою обрано автомат Мура, чия стабільність виходів, синхронність та модульна структура є ідеальними для задач автоматизації.

4.2 Розробка керуючого автомата Мура для СКАПСП

Для визначення необхідної розрядності системи автоматичного керування конвєсером необхідно розрахувати загальну кількість можливих станів системи див. рис. 4.2.

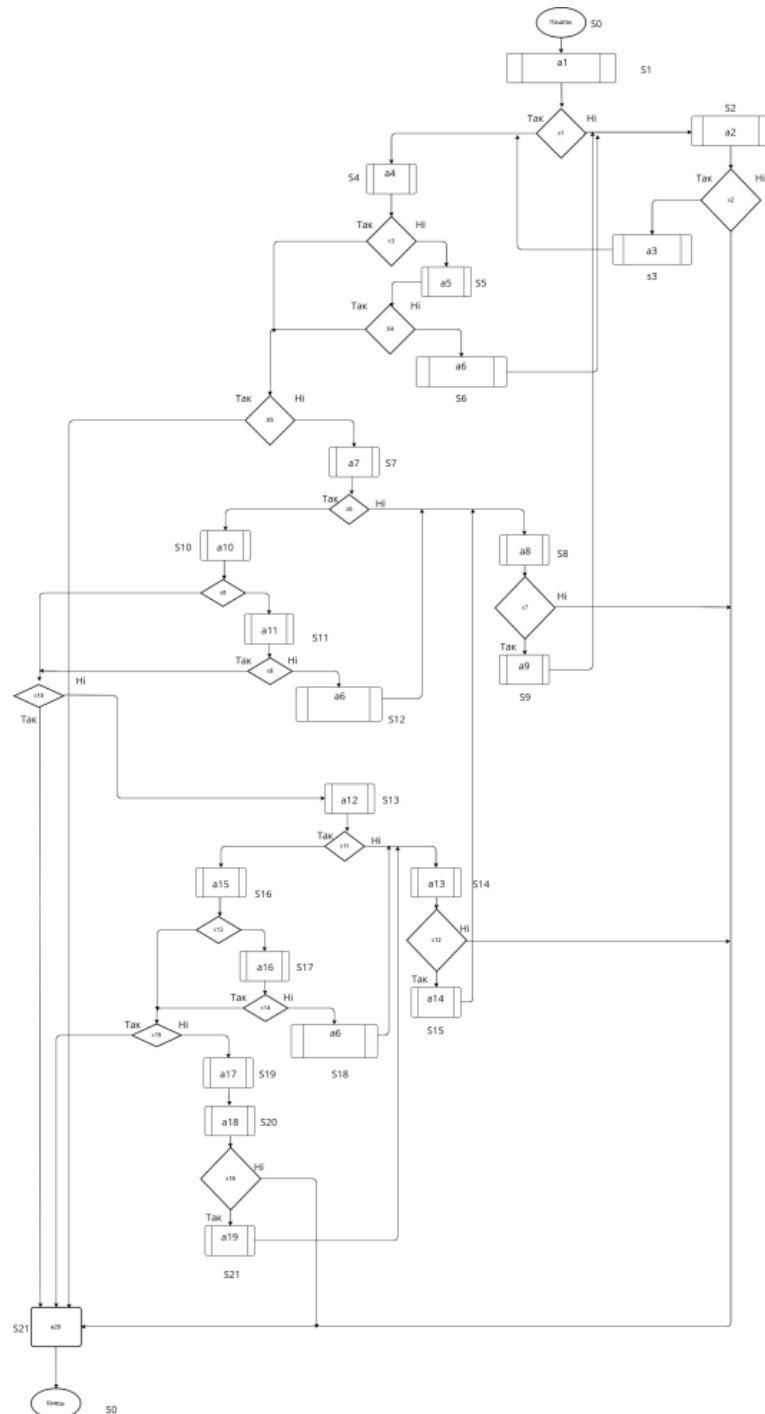


Рисунок 4.2 – Граф-схема станів одної з гілок СКАПСП

Згідно з графом станів (рис. 4.2), система має 22 різні стани. Для представлення кожного стану в двійковому коді потрібно 5 розрядів (Q_4, Q_3, Q_2, Q_1, Q_0). Таким чином, таблиця (Таблиця 4.1) прямого кодування станів буде містити 22 рядки, в кожному з яких буде 5 бітів.

Таблиця 4.1 – Коди станів автомата

| Стан | Код | | | | |
|----------|-------|-------|-------|-------|-------|
| | Q_4 | Q_3 | Q_2 | Q_1 | Q_0 |
| S_0 | 0 | 0 | 0 | 0 | 0 |
| S_1 | 0 | 0 | 0 | 0 | 1 |
| S_2 | 0 | 0 | 0 | 1 | 0 |
| S_3 | 0 | 0 | 0 | 1 | 1 |
| S_4 | 0 | 0 | 1 | 0 | 0 |
| S_5 | 0 | 0 | 1 | 0 | 1 |
| S_6 | 0 | 0 | 1 | 1 | 0 |
| S_7 | 0 | 0 | 1 | 1 | 1 |
| S_8 | 0 | 1 | 0 | 0 | 0 |
| S_9 | 0 | 1 | 0 | 0 | 1 |
| S_{10} | 0 | 1 | 0 | 1 | 0 |
| S_{11} | 0 | 1 | 0 | 1 | 1 |
| S_{12} | 0 | 1 | 1 | 0 | 0 |
| S_{13} | 0 | 1 | 1 | 0 | 1 |
| S_{14} | 0 | 1 | 1 | 1 | 0 |
| S_{15} | 0 | 1 | 1 | 1 | 1 |
| S_{16} | 1 | 0 | 0 | 0 | 0 |
| S_{17} | 1 | 0 | 0 | 0 | 1 |
| S_{18} | 1 | 0 | 0 | 1 | 0 |
| S_{19} | 1 | 0 | 0 | 1 | 1 |
| S_{20} | 1 | 0 | 1 | 0 | 0 |
| S_{21} | 1 | 0 | 1 | 0 | 1 |
| S_{22} | 1 | 0 | 1 | 1 | 0 |

На основі таблиці кодування станів (табл. 4.1) та графу станів однієї з гілок скінченного керуючого автомата паралельної структури (СКАПСП) (рис. 4.2) було сформовано таблицю переходів (табл. 4.2).

Таблиця 4.2 – Таблиця переходів автомата

| Q _t | Код | | | | | Q _{t+1} | Код | | | | | Умо- ва пере- ходу | D ₄ | D ₃ | D ₂ | D ₁ | D ₀ | Фун- кції |
|-----------------|----------------|----------------|----------------|----------------|----------------|------------------|----------------|----------------|----------------|----------------|----------------|------------------------------------|----------------|----------------|----------------|----------------|----------------|-----------------|
| | Q ₄ | Q ₃ | Q ₂ | Q ₁ | Q ₀ | | Q ₄ | Q ₃ | Q ₂ | Q ₁ | Q ₀ | | | | | | | |
| S ₀ | 0 | 0 | 0 | 0 | 0 | S ₁ | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | a ₁ |
| S ₁ | 0 | 0 | 0 | 0 | 1 | S ₂ | 0 | 0 | 0 | 1 | 0 | $\bar{x}_1 x_1$ | 0 | 0 | 0 | 1 | 0 | a ₂ |
| S ₁ | 0 | 0 | 0 | 0 | 1 | S ₄ | 0 | 0 | 1 | 0 | 0 | x ₁ | 0 | 0 | 1 | 0 | 0 | a ₄ |
| S ₂ | 0 | 0 | 0 | 1 | 0 | S ₃ | 0 | 0 | 0 | 1 | 1 | x ₂ | 0 | 0 | 0 | 1 | 1 | a ₃ |
| S ₂ | 0 | 0 | 0 | 1 | 0 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | \bar{x}_2 | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₃ | 0 | 0 | 0 | 1 | 1 | S ₄ | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | a ₄ |
| S ₄ | 0 | 0 | 1 | 0 | 0 | S ₇ | 0 | 0 | 1 | 1 | 1 | x ₃ \bar{x}_5 | 0 | 0 | 1 | 1 | 1 | a ₇ |
| S ₄ | 0 | 0 | 1 | 0 | 0 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | x ₃ x ₅ | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₄ | 0 | 0 | 1 | 0 | 0 | S ₅ | 0 | 0 | 1 | 0 | 1 | \bar{x}_3 | 0 | 0 | 1 | 0 | 1 | a ₅ |
| S ₅ | 0 | 0 | 1 | 0 | 1 | S ₇ | 0 | 0 | 1 | 1 | 1 | x ₄ \bar{x}_5 | 0 | 0 | 1 | 1 | 1 | a ₇ |
| S ₅ | 0 | 0 | 1 | 0 | 1 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | x ₄ x ₅ | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₅ | 0 | 0 | 1 | 0 | 1 | S ₆ | 0 | 0 | 1 | 1 | 0 | \bar{x}_4 | 0 | 0 | 1 | 1 | 0 | a ₆ |
| S ₆ | 0 | 0 | 1 | 1 | 0 | S ₂ | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | a ₂ |
| S ₇ | 0 | 0 | 1 | 1 | 1 | S ₈ | 0 | 1 | 0 | 0 | 0 | \bar{x}_6 | 0 | 1 | 0 | 0 | 0 | a ₈ |
| S ₇ | 0 | 0 | 1 | 1 | 1 | S ₁₀ | 0 | 1 | 0 | 1 | 0 | x ₆ | 0 | 1 | 0 | 1 | 0 | a ₁₀ |
| S ₈ | 0 | 1 | 0 | 0 | 0 | S ₉ | 0 | 1 | 0 | 0 | 1 | x ₇ | 0 | 1 | 0 | 0 | 1 | a ₉ |
| S ₈ | 0 | 1 | 0 | 0 | 0 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | \bar{x}_7 | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₉ | 0 | 1 | 0 | 0 | 1 | S ₂ | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | a ₂ |
| S ₁₀ | 0 | 1 | 0 | 1 | 0 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | x ₈ x ₁₀ | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₁₀ | 0 | 1 | 0 | 1 | 0 | S ₁₁ | 0 | 1 | 0 | 1 | 1 | \bar{x}_8 | 0 | 1 | 0 | 1 | 1 | a ₁₁ |
| S ₁₀ | 0 | 1 | 0 | 1 | 0 | S ₁₃ | 0 | 1 | 1 | 0 | 1 | \bar{x}_{10} x ₈ | 0 | 1 | 1 | 0 | 1 | a ₁₂ |
| S ₁₁ | 0 | 1 | 0 | 1 | 1 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | x ₉ x ₁₀ | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₁₁ | 0 | 1 | 0 | 1 | 1 | S ₁₂ | 0 | 1 | 1 | 0 | 0 | \bar{x}_9 | 0 | 1 | 1 | 0 | 0 | A ₆ |
| S ₁₁ | 0 | 1 | 0 | 1 | 1 | S ₁₃ | 0 | 1 | 1 | 0 | 1 | \bar{x}_{10} x ₉ | 0 | 1 | 1 | 0 | 1 | a ₁₂ |
| S ₁₂ | 0 | 1 | 1 | 0 | 0 | S ₈ | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | a ₈ |
| S ₁₃ | 0 | 1 | 1 | 0 | 1 | S ₁₆ | 1 | 0 | 0 | 0 | 0 | x ₁₁ | 1 | 0 | 0 | 0 | 0 | a ₁₅ |
| S ₁₃ | 0 | 1 | 1 | 0 | 1 | S ₁₄ | 0 | 1 | 1 | 1 | 0 | \bar{x}_{11} | 0 | 1 | 1 | 1 | 0 | a ₁₃ |
| S ₁₄ | 0 | 1 | 1 | 1 | 0 | S ₁₅ | 0 | 1 | 1 | 1 | 1 | x ₁₂ | 0 | 1 | 1 | 1 | 1 | a ₁₄ |
| S ₁₅ | 0 | 1 | 1 | 1 | 1 | S ₈ | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | a ₈ |
| S ₁₄ | 0 | 1 | 1 | 1 | 0 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | \bar{x}_{12} | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₁₆ | 1 | 0 | 0 | 0 | 0 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | x ₁₃ x ₁₅ | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₁₆ | 1 | 0 | 0 | 0 | 0 | S ₁₉ | 1 | 0 | 0 | 1 | 1 | x ₁₃ \bar{x}_{15} | 1 | 0 | 0 | 1 | 1 | a ₁₇ |
| S ₁₆ | 1 | 0 | 0 | 0 | 0 | S ₁₇ | 1 | 0 | 0 | 0 | 1 | \bar{x}_{13} | 1 | 0 | 0 | 0 | 1 | a ₁₈ |

Продовження таблиці 4.2

| | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|-----------------|---|---|---|---|---|------------------------------------|---|---|---|---|-----------------|-----------------|
| S ₁₇ | 1 | 0 | 0 | 0 | 1 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | X ₁₄ X ₁₅ | 1 | 0 | 1 | 1 | 0 | a ₂₀ |
| S ₁₇ | 1 | 0 | 0 | 0 | 1 | S ₁₉ | 1 | 0 | 0 | 1 | 1 | X ₁₄ \bar{X}_{15} | 1 | 0 | 0 | 1 | 1 | a ₁₇ |
| S ₁₇ | 1 | 0 | 0 | 0 | 1 | S ₁₈ | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | a ₆ | |
| S ₁₈ | 1 | 0 | 0 | 1 | 0 | S ₁₄ | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | a ₁₃ |
| S ₁₉ | 1 | 0 | 0 | 1 | 1 | S ₂₀ | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | a ₁₈ |
| S ₂₀ | 1 | 0 | 1 | 0 | 0 | S ₂₁ | 1 | 0 | 1 | 0 | 1 | X ₁₆ | 1 | 0 | 1 | 0 | 1 | a ₁₉ |
| S ₂₀ | 1 | 0 | 1 | 0 | 0 | S ₂₂ | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | a ₂₀ | |
| S ₂₁ | 1 | 0 | 1 | 0 | 1 | S ₁₄ | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | a ₁₃ |
| S ₂₂ | 1 | 0 | 1 | 1 | 0 | S ₀ | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | - |

Формування та мінімізація функцій збудження елементів пам'яті і функцій виходів:

$$\begin{aligned}
 D_0 = & \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 X_2 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 X_3 \bar{X}_5 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 \bar{X}_3 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 X_4 \bar{X}_5 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 X_7 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 \bar{X}_8 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 X_8 \bar{X}_{10} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 Q_0 X_9 \bar{X}_{10} \vee \bar{Q}_4 \bar{Q}_3 Q_2 Q_1 \bar{Q}_0 X_{12} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 X_{13} \bar{X}_{15} \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 \bar{X}_{13} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 X_{14} \bar{X}_{15} \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 X_{16} \\
 D_1 = & \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 \bar{X}_1 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 X_2 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 \bar{X}_2 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 X_3 \bar{X}_5 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 X_3 X_5 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 X_4 \bar{X}_5 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 X_4 X_5 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 \bar{X}_4 \vee \bar{Q}_4 \bar{Q}_3 Q_2 Q_1 \bar{Q}_0 \vee \bar{Q}_4 \bar{Q}_3 Q_2 Q_1 Q_0 X_6 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 \bar{X}_7 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 X_8 X_{10} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 \bar{X}_8 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 Q_0 X_9 X_{10} \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 \bar{X}_{11} \vee \bar{Q}_4 \bar{Q}_3 Q_2 Q_1 \bar{Q}_0 X_{12} \vee \bar{Q}_4 \bar{Q}_3 Q_2 Q_1 \bar{Q}_0 \bar{X}_{12} \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 X_{13} X_{15} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 X_{13} \bar{X}_{15} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 X_{14} X_{15} \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 X_{14} \bar{X}_{15} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 \bar{X}_{14} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 \bar{X}_{16} \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 \\
 D_2 = & \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 X_1 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 \bar{X}_2 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 Q_0 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 X_3 \bar{X}_5 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 X_3 X_5 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 \bar{Q}_0 \bar{X}_3 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 X_4 \bar{X}_5 \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 X_4 X_5 \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 \bar{X}_4 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 \bar{X}_7 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 X_8 X_{10} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 \bar{Q}_0 X_8 \bar{X}_{10} \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 Q_0 X_9 X_{10} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 Q_0 \bar{X}_9 \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 Q_1 Q_0 X_9 \bar{X}_{10} \vee \bar{Q}_4 \bar{Q}_3 Q_2 \bar{Q}_1 Q_0 \bar{X}_{11} \vee \\
 & \vee \bar{Q}_4 \bar{Q}_3 Q_2 Q_1 \bar{Q}_0 X_{12} \vee \bar{Q}_4 \bar{Q}_3 Q_2 Q_1 \bar{Q}_0 \bar{X}_{12} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 X_{13} X_{15} \vee \bar{Q}_4 \bar{Q}_3 \bar{Q}_2 \bar{Q}_1 Q_0 X_{14} X_{15} \vee
 \end{aligned}$$

$$\vee Q_4 \overline{Q_3} \overline{Q_2} Q_1 \overline{Q_0} \vee Q_4 \overline{Q_3} \overline{Q_2} Q_1 Q_0 \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} X_{16} \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} \overline{X_{16}} \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} Q_0$$

$$D_3 = \overline{Q_4} \overline{Q_3} Q_2 Q_1 Q_0 \overline{X_6} \vee \overline{Q_4} \overline{Q_3} Q_2 Q_1 Q_0 X_6 \vee \overline{Q_4} Q_3 \overline{Q_2} \overline{Q_1} \overline{Q_0} X_7 \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 \overline{Q_0} \overline{X_8} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 \overline{Q_0} X_8 \overline{X_{10}} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 Q_0 \overline{X_9} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 Q_0 X_9 \overline{X_{10}} \vee \overline{Q_4} Q_3 Q_2 \overline{Q_1} \overline{Q_0} \vee \overline{Q_4} Q_3 Q_2 \overline{Q_1} Q_0 \overline{X_{11}} \vee \overline{Q_4} Q_3 Q_2 Q_1 \overline{Q_0} X_{12} \vee \overline{Q_4} Q_3 Q_2 Q_1 Q_0 \vee Q_4 \overline{Q_3} \overline{Q_2} Q_1 \overline{Q_0} \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} Q_0$$

$$D_4 = \overline{Q_4} \overline{Q_3} \overline{Q_2} Q_1 \overline{Q_0} \overline{X_2} \vee \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} X_3 X_5 \vee \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} Q_0 X_4 X_5 \vee \overline{Q_4} Q_3 \overline{Q_2} \overline{Q_1} \overline{Q_0} \overline{X_7} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 \overline{Q_0} X_8 X_{10} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 Q_0 X_9 X_{10} \vee \overline{Q_4} Q_3 Q_2 \overline{Q_1} Q_0 X_{11} \vee \overline{Q_4} Q_3 Q_2 Q_1 \overline{Q_0} \overline{X_{12}} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0} X_{13} X_{15} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0} X_{13} \overline{X_{15}} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0} \overline{X_{13}} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 X_{14} X_{15} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 X_{14} \overline{X_{15}} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 \overline{X_{14}} \vee Q_4 \overline{Q_3} \overline{Q_2} Q_1 Q_0 \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} X_{16} \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} \overline{X_{16}}$$

Згідно таблиці переходів і виходів табл. 4.2 ми отримаємо таку функцію виходу:

$$\begin{aligned} a_1 &= \overline{Q_4} \overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0} \\ a_2 &= \overline{Q_4} \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 X_1 \vee \overline{Q_4} \overline{Q_3} Q_2 Q_1 \overline{Q_0} \vee \overline{Q_4} Q_3 \overline{Q_2} \overline{Q_1} Q_0 \\ a_4 &= \overline{Q_4} \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 \overline{X_1} \vee \overline{Q_4} \overline{Q_3} \overline{Q_2} Q_1 Q_0 \\ a_3 &= \overline{Q_4} \overline{Q_3} \overline{Q_2} Q_1 \overline{Q_0} X_2 \\ a_{20} &= \overline{Q_4} \overline{Q_3} \overline{Q_2} Q_1 \overline{Q_0} \overline{X_2} \vee \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} X_3 X_5 \vee \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} Q_0 X_4 X_5 \vee \overline{Q_4} Q_3 \overline{Q_2} \overline{Q_1} \overline{Q_0} \overline{X_7} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 \overline{Q_0} X_8 X_{10} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 Q_0 X_9 X_{10} \vee \overline{Q_4} Q_3 Q_2 Q_1 \overline{Q_0} \overline{X_{12}} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0} X_{13} X_{15} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 X_{14} X_{15} \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} \overline{X_{16}} \\ a_7 &= \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} X_3 \overline{X_5} \vee \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} Q_0 X_4 \overline{X_5} \\ a_5 &= \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} \overline{X_3} \\ a_6 &= \overline{Q_4} \overline{Q_3} Q_2 \overline{Q_1} Q_0 \overline{X_4} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 Q_0 \overline{X_9} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 \overline{X_{14}} \\ a_{12} &= \overline{Q_4} Q_3 \overline{Q_2} Q_1 \overline{Q_0} X_8 \overline{X_{10}} \vee \overline{Q_4} Q_3 \overline{Q_2} Q_1 Q_0 X_9 \overline{X_{10}} \\ a_8 &= \overline{Q_4} \overline{Q_3} Q_2 Q_1 Q_0 \overline{X_6} \vee \overline{Q_4} Q_3 Q_2 \overline{Q_1} \overline{Q_0} \vee \overline{Q_4} Q_3 Q_2 Q_1 Q_0 \\ a_{10} &= \overline{Q_4} \overline{Q_3} Q_2 Q_1 Q_0 X_6 \\ a_{15} &= \overline{Q_4} Q_3 Q_2 \overline{Q_1} Q_0 X_{11} \\ a_{13} &= \overline{Q_4} Q_3 Q_2 \overline{Q_1} Q_0 \overline{X_{11}} \vee Q_4 \overline{Q_3} \overline{Q_2} Q_1 \overline{Q_0} \vee Q_4 \overline{Q_3} Q_2 \overline{Q_1} Q_0 \\ a_9 &= \overline{Q_4} Q_3 \overline{Q_2} \overline{Q_1} \overline{Q_0} X_7 \\ a_{14} &= \overline{Q_4} Q_3 Q_2 Q_1 \overline{Q_0} X_{12} \end{aligned}$$

$$a_{17} = Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0} x_{13} \overline{x_{15}} \vee Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} Q_0 x_{14} \overline{x_{15}}$$

$$a_{11} = \overline{Q_4} Q_3 \overline{Q_2} Q_1 \overline{Q_0} \overline{x_8}$$

$$a_{16} = Q_4 \overline{Q_3} \overline{Q_2} \overline{Q_1} \overline{Q_0} \overline{x_{13}}$$

$$a_{18} = Q_4 \overline{Q_3} \overline{Q_2} Q_1 Q_0$$

$$a_{19} = Q_4 \overline{Q_3} Q_2 \overline{Q_1} \overline{Q_0} x_{16}$$

Створено схему в середовищі розробки Active-HDL див. рис. 4.3.

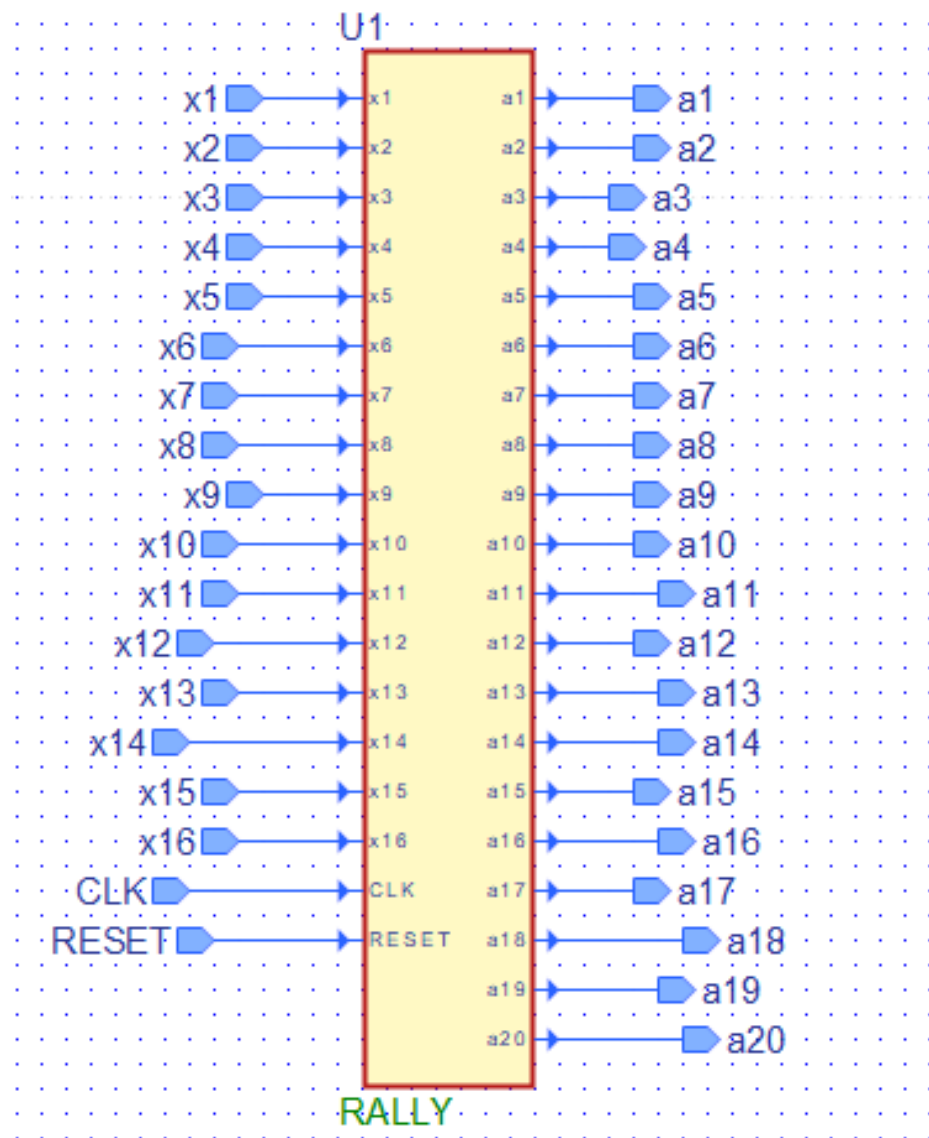


Рисунок 4.3 – Схема СКАПСП

Експериментальні дані (рис 4.4), отримані в результаті моделювання в ActiveHDL (рис4.3), підтверджують теоретичні розрахунки, що базуються на структурній математичній моделі системи.

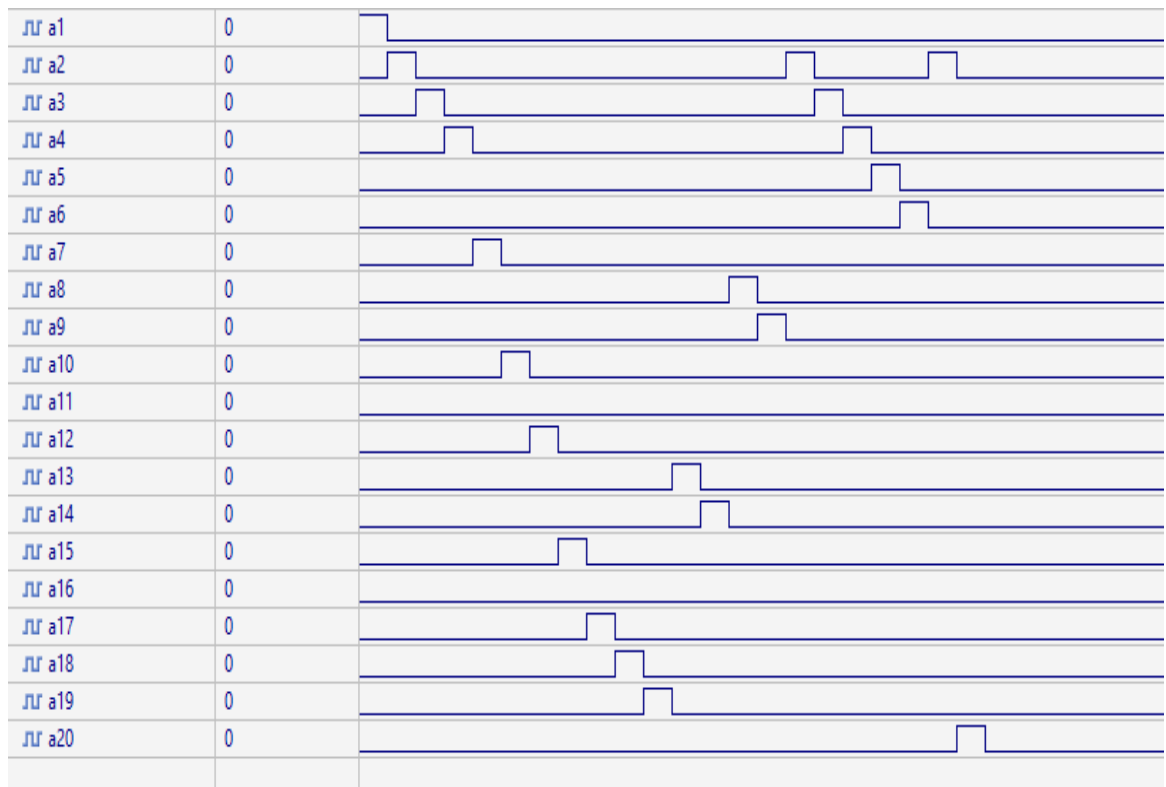


Рисунок 4.4 – Часова діаграма СКАПСП

Якщо на вході системи відсутній пакунок (x_1), то вводяться значення температур усіх (a_2) двигунів конвеєрів. Якщо температура кожного двигуна перебуває в діапазоні від 5 до 40 градусів за Цельсієм, то система запускає відповідний двигун (a_3) та вимикає його після того, як пакунок досягає першого поворотного механізму. Відбувається зчитування штрихкоду на першому поворотному механізмі (a_4). При успішному зчитуванні штрихкоду система перевіряє значення першої частини штрихкоду (x_5). Якщо X не дорівнює або більше 1, активується перший поворотний механізм (a_7). Якщо пакунок присутній (x_6), то відбувається зчитування штрихкоду (a_{10}). При успішному зчитуванні штрихкоду (x_8), відбувається перевірка другої частини штрихкоду (x_{10}). Якщо $Y < 60$, активується другий поворотний механізм (a_{12}). Якщо на вході системи присутній пакунок (x_{11}), то відбувається зчитування штрихкоду (a_{15}). При успішному зчитуванні штрихкоду (x_{13}), відбувається перевірка третьої частини штрихкоду (x_{15}). Якщо $Z < 100$, активується другий поворотний механізм (a_{17}). Вводяться значення температур усіх (a_{18}) двигунів конвеєрів. Якщо температура кожного двигуна перебуває в діапазоні від 5 до 40

градусів за Цельсієм, то система запускає відповідний двигун (a19) та вимикає його після того, як паунок досягає третього поворотного механізму. Після вилучення паунка система вводить (a13) та перевіряє значення температур усіх двигунів. При успішній діагностиці вмикається третій стрічковий конвеєр та другий поворотний механізм. Система перевіряє, чи залишив паунок другий поворотний механізм (a14). Якщо залишив, система знову проводить діагностику усіх двигунів (a2). Якщо немає несправностей, система зчитує значення штрихкоду паунка на першому поворотному механізмі (a4). Якщо при зчитуванні виникають проблеми ($Y < 0$, $Z < 0$), система робить ще дві спроби зчитати штрихкод (a5). При невдалому зчитуванні (x4), система буде виводити повідомлення «Помилка ідентифікації», доки паунок не вилучать з першого поворотного механізму (a6). Після вилучення система знову перевіряє стан усіх конвеєрів (a2). Якщо значення температур усіх конвеєрів не відповідають чітко визначеним нормам (x2), система вимикає усі конвеєри (a20). Цей механізм контролю та діагностики забезпечує безперебійну роботу системи сортування.

4.3 Реалізація СКАПСП на Cyclone III, Cyclone IV та Arria II GX

Проведемо порівняльний аналіз ефективності виконання однієї з восьми гілок алгоритму СКАПСП (рис. 4.1) на трьох різних ПЛІС: Cyclone III EP3C16F484C6, Cyclone IV GX EP4CGX15BF14A7 та Arria II GX EP2AGX95EF35C5. Метою даного дослідження є визначення оптимального вибору ПЛІС для реалізації зазначеного алгоритму на основі результатів проведеного аналізу.

Для підтвердження коректності функціонування розробленого алгоритму на кожній ПЛІС, буде проведено тестування з використанням програмного забезпечення Quartus II. Результатом даного етапу стане генерація часових діаграм для кожної реалізації (рис 4.5, рис 4.6, рис 4.7).

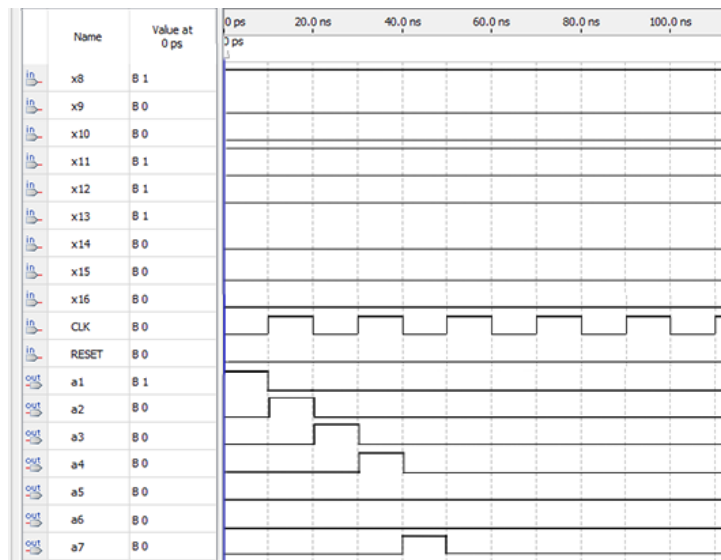


Рисунок 4.5 – Часова діаграма Cyclone III

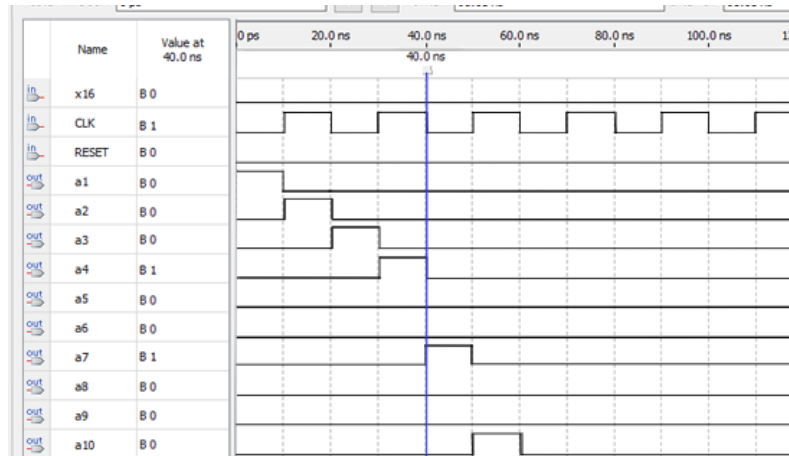


Рисунок 4.6 – Часова діаграма Cyclone III IV GX

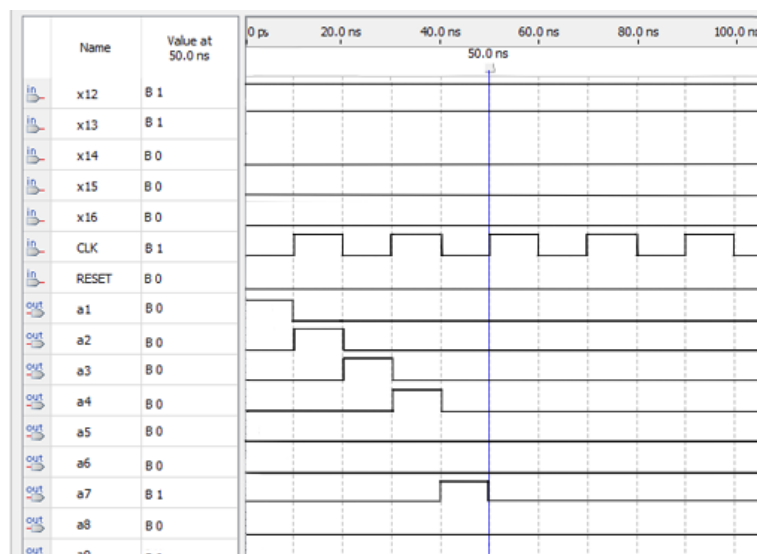


Рисунок 4.7 – Часова діаграма Arria II GX

У результаті компіляції коду алгоритму на кожному з досліджуваних кристалів було отримано детальні звіти про імплементацію, які містять детальну інформацію про використання апаратних ресурсів (рис 4.8, рис 4.9, рис 4.10).

| | |
|------------------------------------|---|
| Flow Status | Successful - Wed Jan 22 02:25:24 2025 |
| Quartus II 64-Bit Version | 13.1.0 Build 162 10/23/2013 SJ Full Version |
| Revision Name | sort |
| Top-level Entity Name | sort |
| Family | Cyclone III |
| Device | EP3C16F484C6 |
| Timing Models | Final |
| Total logic elements | 54 / 15,408 (< 1 %) |
| Total combinational functions | 54 / 15,408 (< 1 %) |
| Dedicated logic registers | 4 / 15,408 (< 1 %) |
| Total registers | 4 |
| Total pins | 38 / 347 (11 %) |
| Total virtual pins | 0 |
| Total memory bits | 0 / 516,096 (0 %) |
| Embedded Multiplier 9-bit elements | 0 / 112 (0 %) |
| Total PLLs | 0 / 4 (0 %) |

Рисунок 4.8 – Звіт імплементації Cyclone III

| | |
|------------------------------------|---|
| Flow Status | Successful - Wed Jan 22 02:57:12 2025 |
| Quartus II 64-Bit Version | 13.1.0 Build 162 10/23/2013 SJ Full Version |
| Revision Name | topology |
| Top-level Entity Name | topology |
| Family | Cyclone IV GX |
| Device | EP4CGX15BF14A7 |
| Timing Models | Final |
| Total logic elements | 54 / 14,400 (< 1 %) |
| Total combinational functions | 54 / 14,400 (< 1 %) |
| Dedicated logic registers | 4 / 14,400 (< 1 %) |
| Total registers | 4 |
| Total pins | 38 / 81 (47 %) |
| Total virtual pins | 0 |
| Total memory bits | 0 / 552,960 (0 %) |
| Embedded Multiplier 9-bit elements | 0 |
| Total GXB Receiver Channel PCS | 0 / 2 (0 %) |
| Total GXB Receiver Channel PMA | 0 / 2 (0 %) |
| Total GXB Transmitter Channel PCS | 0 / 2 (0 %) |
| Total GXB Transmitter Channel PMA | 0 / 2 (0 %) |
| Total PLLs | 0 / 3 (0 %) |

Рисунок 4.9 – Звіт імплементації Cyclone IV GX

| | |
|-----------------------------------|---|
| Flow Status | Successful - Wed Jan 22 22:14:29 2025 |
| Quartus II 64-Bit Version | 13.1.0 Build 162 10/23/2013 SJ Full Version |
| Revision Name | last |
| Top-level Entity Name | last |
| Family | Arria II GX |
| Device | EP2AGX95EF35C5 |
| Timing Models | Final |
| Logic utilization | < 1 % |
| Combinational ALUTs | 41 / 74,940 (< 1 %) |
| Memory ALUTs | 0 / 37,470 (0 %) |
| Dedicated logic registers | 4 / 74,940 (< 1 %) |
| Total registers | 4 |
| Total pins | 38 / 512 (7 %) |
| Total virtual pins | 0 |
| Total block memory bits | 0 / 5,640,192 (0 %) |
| DSP block 18-bit elements | 0 / 448 (0 %) |
| Total GXB Receiver Channel PCS | 0 / 12 (0 %) |
| Total GXB Receiver Channel PMA | 0 / 12 (0 %) |
| Total GXB Transmitter Channel PCS | 0 / 12 (0 %) |
| Total GXB Transmitter Channel PMA | 0 / 12 (0 %) |
| Total PLLs | 0 / 6 (0 %) |
| Total DLLs | 0 / 2 (0 %) |

Рисунок 4.10 – Звіт імплементації Arria II GX

Хоча всі три ПЛІС показують дуже низький відсоток використання логічних елементів (рис 4.11), важливо порівнювати абсолютні значення. Arria II GX використовує найменше логічних елементів (41 ALUT), тоді як Cyclone IV GX та Cyclone III використовують по 54 LE. Зважаючи на різницю в архітектурі (ALUT vs LE), пряме порівняння кількості не зовсім коректне, але в даному випадку, при такому малому використанні, Arria II GX виглядає трохи ефективніше з точки зору використання логіки.

Хоча Arria II GX виглядає трохи ефективнішою з точки зору використання логіки при такому низькому рівні завантаження, остаточний вибір залежить від повного аналізу потреб вашого проекту та порівняння всіх ключових характеристик ПЛІС, включаючи внутрішні з'єднання, використання інших ресурсів, частоту роботи, споживання енергії, вартість та доступність. Рекомендується провести більш детальне порівняння з урахуванням специфічних вимог вашого застосування.

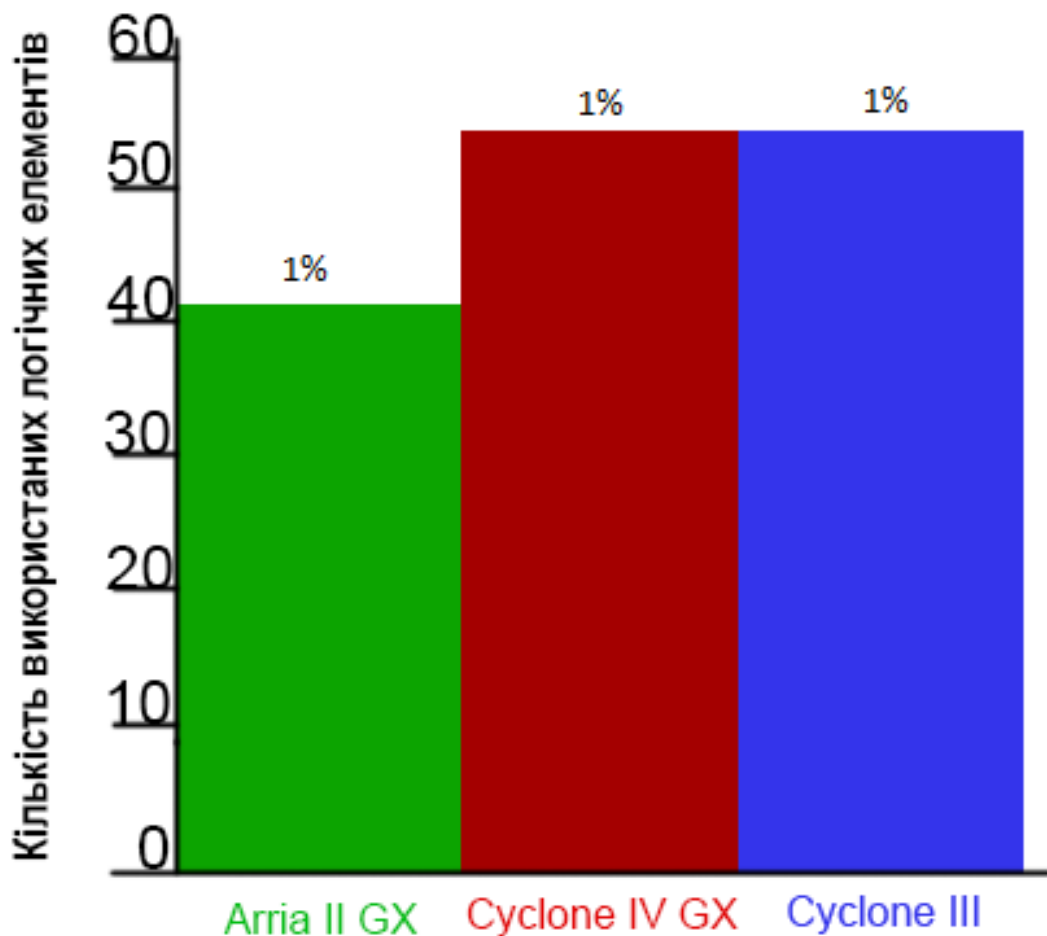


Рисунок 4.11 – Порівняння кількості логічних елементів

Незважаючи на однакову кількість використаних виводів (рис 4.12), для всіх трьох ПЛІС, відсоток їх задіяння суттєво різниться: 47% для Cyclone IV GX, 7% для Arria II GX та 11% для Cyclone III. Cyclone IV GX демонструє найвищий рівень використання, наближаючись до граничної межі, тоді як Arria II GX забезпечує найбільший резерв для майбутніх розширень.

Незважаючи на те, що всі три досліджувані ПЛІС використовують ідентичну кількість виводів (рис 4.12), спостерігається значна різниця у відсотковому співвідношенні використаних виводів до їх загальної кількості. Зокрема, для Cyclone IV GX цей показник досягає 47%, що вказує на близькість до граничного значення та потенційні обмеження у випадку необхідності розширення функціональності проекту, пов'язаної з

використанням додаткових виводів. Arria II GX, з іншого боку, демонструє мінімальне використання виводів (7%), забезпечуючи значний потенціал для модифікацій.

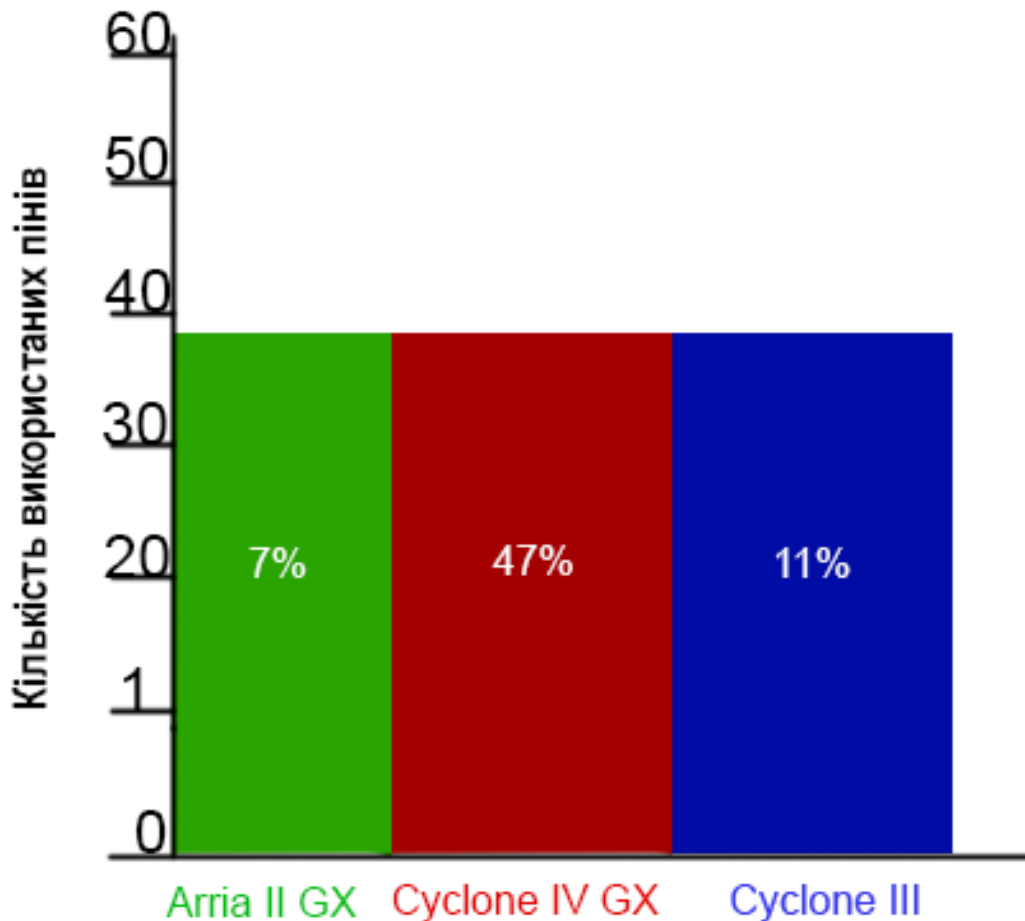


Рисунок 4.12 – Порівняння кількості пінів

Як і з логічними елементами, всі три ПЛІС (рис 4.13) використовують однакову кількість регістрів, що становить менше 1% від доступних. Тому за цим параметром суттєвої різниці немає.

Аналіз топології Cyclone III у контексті наданого звіту компіляції свідчить про реалізацію проєкту з низьким рівнем використання апаратних ресурсів. Більша частина потенціалу кристала залишається незадіяною, що відкриває дві основні можливості: використання меншої та, відповідно,

економічно вигіднішої ПЛІС, або ж суттєве розширення функціональності проєкту на даному кристалі.

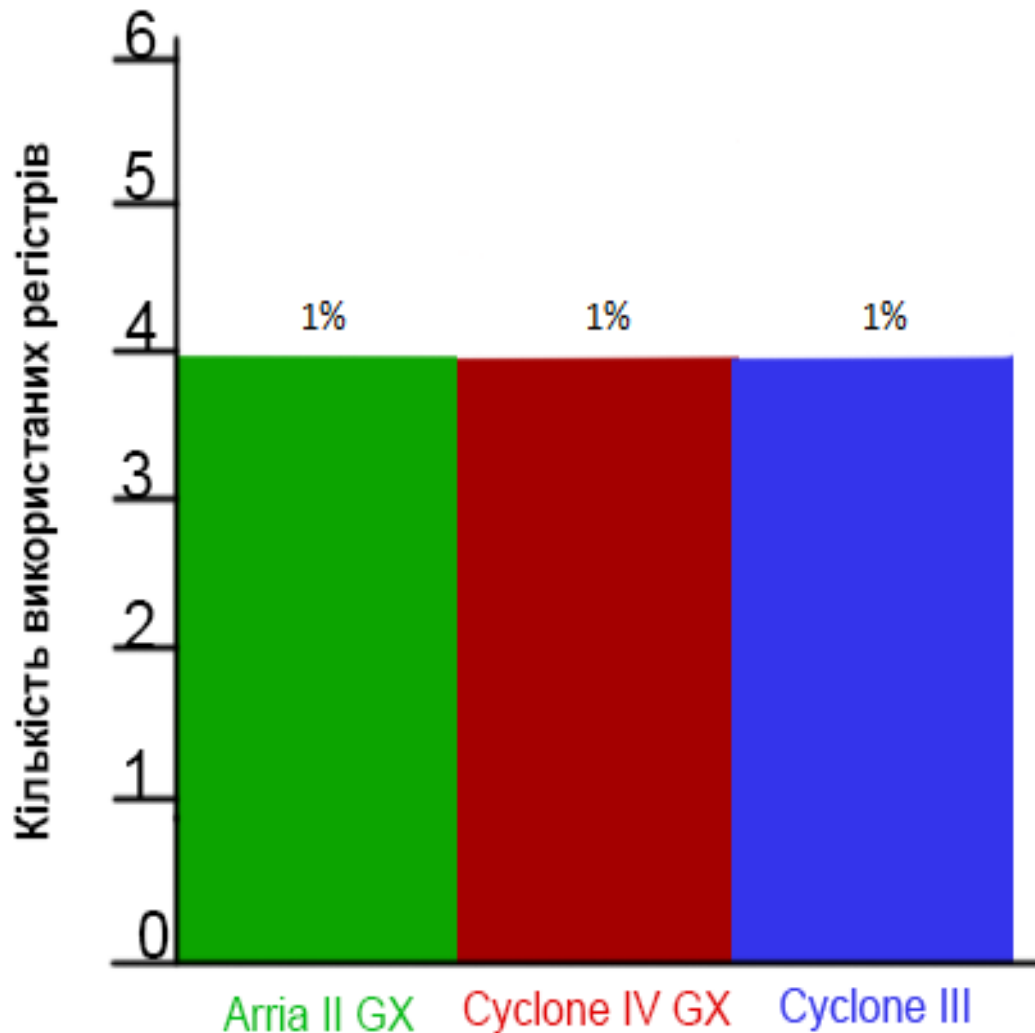


Рисунок 4.13 – Порівняння кількості регістрів

Як і з логічними елементами, всі три ПЛІС (рис 4.13) використовують однакову кількість регістрів, що становить менше 1% від доступних. Тому за цим параметром суттєвої різниці немає.

Аналіз топології Cyclone III у контексті наданого звіту компіляції свідчить про реалізацію проєкту з низьким рівнем використання апаратних ресурсів. Більша частина потенціалу кристала залишається незадіяною, що

відкриває дві основні можливості: використання меншої та, відповідно, економічно вигіднішої ПЛІС, або ж суттєве розширення функціональності проєкту на даному кристалі.

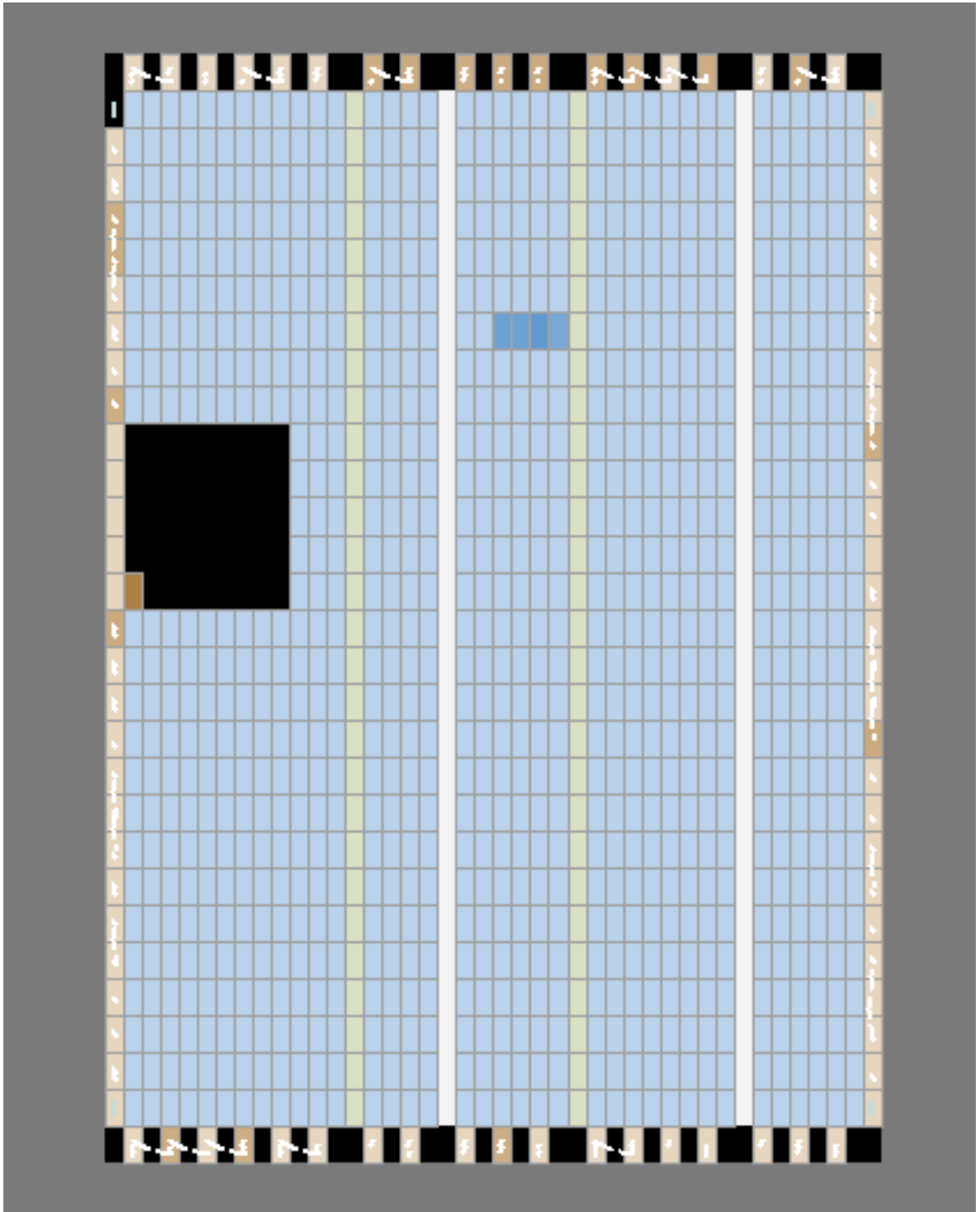


Рисунок 4.14 – Топологія Cyclone III

Cyclone IV GX є надмірно потужним для даного проєкту. Хоча

використання логіки та регістрів мінімальне, використання майже половини доступних виводів (47%) є значним недоліком (рис 4.15). Це означає, що кристал використовується неефективно з точки зору вартості та потенціалу розширення. Використання майже половини виводів при мінімальному використанні інших ресурсів робить його неефективним з точки зору вартості та можливостей розширення. Швидкий час компіляції не компенсує неефективне використання ресурсів.

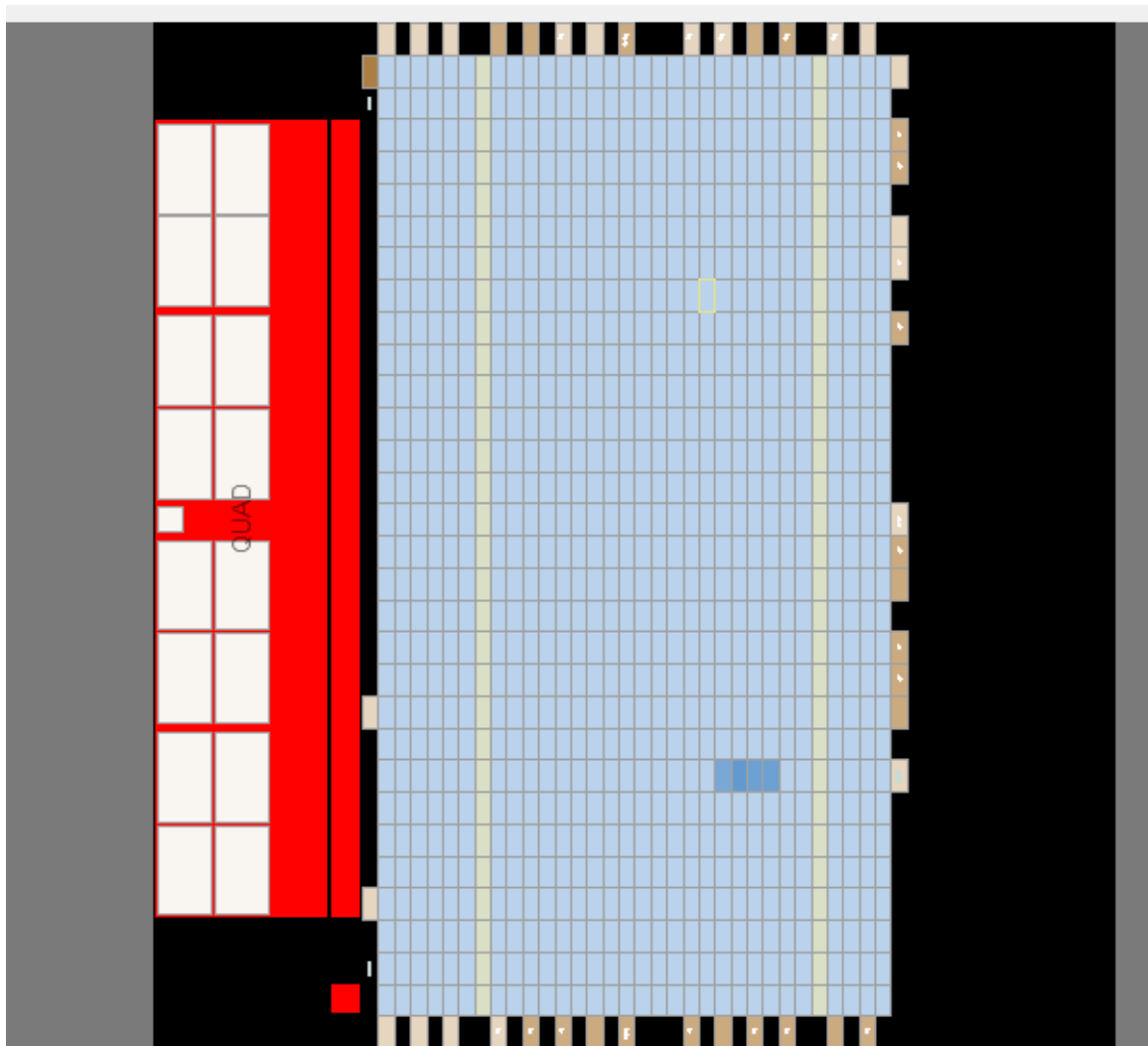


Рисунок 4.15 – Топологія Cyclone IV GX

На основі наданих даних (рис 4.16), Arria II GX виглядає найбільш ефективним вибором для цього даного проєкту. Хоча різниця у використанні логічних елементів не дуже велика, Arria II GX має значно більший запас по

кількості виводів. Це важливо, якщо в майбутньому планується розширення функціональності проєкту або підключення додаткових зовнішніх пристроїв. Cyclone IV GX, навпаки, використовує майже половину доступних виводів, що може стати обмеженням у майбутньому. Cyclone III займає проміжне положення.

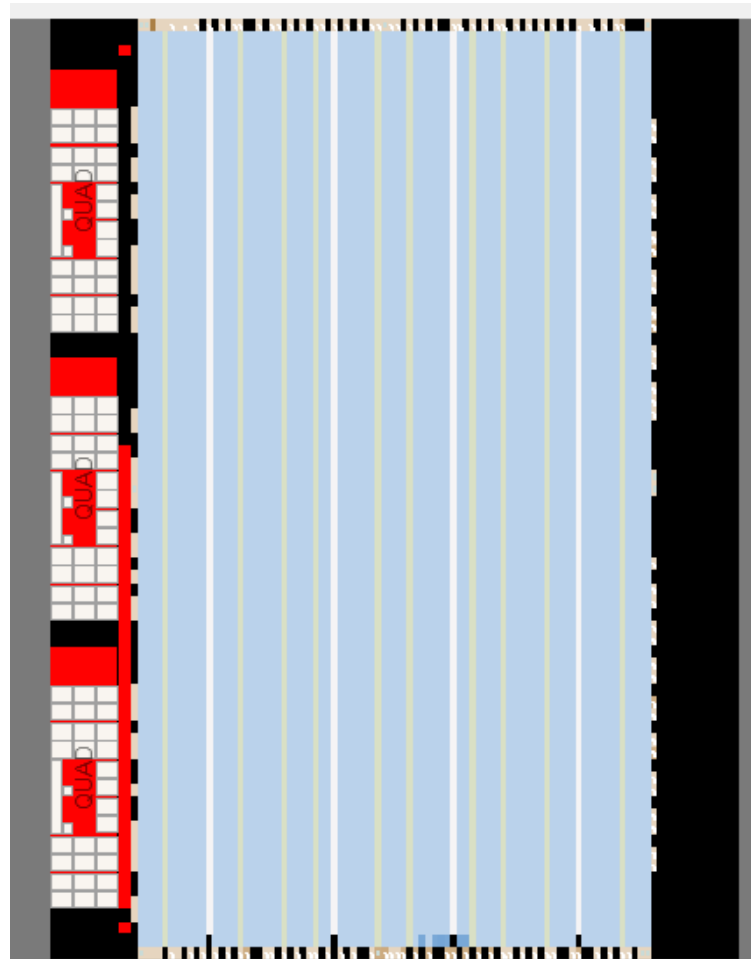


Рисунок 4.16 – Топологія Arria II GX

Arria II GX може бути ефективним вибором для АСС з огляду на запас по виводах та потенціал розширення. Однак, для серійного виробництва або якщо вартість є критичним фактором, слід ретельно зважити всі фактори, включаючи вартість, швидкодію, споживання енергії, доступність та потенційні майбутні вимоги, та можливо розглянути Cyclone III як більш економічний варіант. Остаточне рішення має базуватися на комплексному аналізі всіх вимог до проєкту.

ВИСНОВКИ

В рамках дипломного проєкту була розроблена система автоматизованого сортування пакунків на основі ПЛІС, яка забезпечує ефективне керування процесом сортування за рахунок підвищення продуктивності, зниження кількості помилок та оптимізації витрат ресурсів.

Для розробки програмного забезпечення було обрано мову опису апаратури VHDL та середовище розробки ActiveHDL. Перед початком розробки було проведено комплексне дослідження існуючих систем автоматизованого сортування пакунків, включаючи аналіз їхньої архітектури, компонентів.

Дослідження систем автоматизованого сортування пакунків на основі ПЛІС може мати значний вплив на склади та підприємства, приносячи ряд переваг та покращуючи ефективність їхньої роботи. Впровадження систем автоматизованого сортування пакунків на основі ПЛІС є ефективним рішенням для складів та підприємств будь-якого масштабу. Завдяки високій продуктивності, надійності та гнучкості, такі системи значно покращують ефективність логістичних процесів. Використання ПЛІС дозволяє створювати індивідуальні рішення, адаптовані до конкретних потреб підприємства, при цьому мінімізуючи витрати на програмне забезпечення та обслуговування.

Розроблена система автоматизованого сортування пакунків характеризується виділенням окремих блоків, призначених для виконання елементарних підпрограм. Використання даної блочної структури дозволяє спростити процес масштабування системи сортування до необхідних розмірів шляхом комбінації елементарних блоків зі своїми підпрограмами, модернізуючи лише систему загального керування. Це дозволяє скоротити часові витрати на етапі проєктування шляхом використання однотипних структурних елементів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Steven S. Skiena The Algorithm Design Manual Second Edition. New York, 1997. 730 p.
2. Corman T., Lazerson C.E, Stein C., Rivest R.L. Introduction to Algorithms Fourth Edition. London, 2022. 1290 p.
3. Christopher M. Logistics & Supply Chain Management Fourth Edition. London, 2011. 276 p.
4. Hamdy A. Taha Operations Research: An Introduction Eighth Edition. Upper Saddle River, 2006. 813 p.
5. Vandeput N. Data Science for Supply Chain Forecasting. Berlin, 2021.310 p
- 6 Попович П.Г., Ковальчук О. В., Теорія автоматичного керування. Київ, 2007. 655 с.
- 7 Ладанюк А.П., Заєць Н.А., Власенко Л.О. Сучасні технології конструювання систем автоматизації складних об'єктів: монографія. Київ: Видавництво Ліра-К, 2016. 294 с.
- 8 Anil K. Maini Digital Electronics: Principles, Devices and Applications. New York, 2007. 752 p.
- 9 Що таке WMS система. URL : <https://uislab.com/uk/chto-takoe-wms-sistema/> (дата звернення: 25.10.2024).
- 10 WMS система: як це працює?. <https://ssk.ua/ua/blog/wms-sistema-kak-eto-rabotaet-501>(дата звернення: 25.10.2024).
- 11 Automated parcel sorting: How to Transform Your CEP Business. . URL : <https://www.beumergroup.com/knowledge/cep/automated-parcel-sorting-an-introductory-guide/> (дата звернення: 26.10.2024).
- 12 What is a sorting system? URL : <https://equinoxmhe.com/what-is-a-sorting-system/> (дата звернення: 27.10.2024).
- 13 Які застосовуються системи сортування товарів на складі та як підібрати оптимальне рішення. URL : <https://kapelou.com/blog/statti/systemy->

sortuvannia-tovariv (дата звернення: 24.10.2024).

14 Types of Sortation Systems. URL: <https://www.elementlogic.net/us/insights/types-of-sortation-systems/> (дата звернення: 28.10.2024).

15 Sorting Systems Package & Parcel Sorting Solutions for Logistics & Distribution Centers. URL : <https://www.semiphoton.com/sorting-systems/> (дата звернення: 28.10.2024).

16 What is a Sorter System? URL : <https://www.beevision.ai/what-is-the-sorter-system> (дата звернення: 28.10.2024).

17 Parcel sorting system at DHL logistics. URL: <https://sortec-online.de/en/parcel-sorting-system-at-dhl-logistics/> (дата звернення: 28.10.2024).

18 Automated sorting solutions. URL: <https://www.ulmahandling.com/en/automated-intralogistics/solution-automated-sorting-systems> (дата звернення: 29.10.2024).

ДОДАТОК А ЛІСТИНГИ ПРОГРАМ

Лістинг 4.1 – VHDL код СКАПСІ

```
library IEEE;
use IEEE.std_logic_1164.all;

entity RALLY is
  port(
    x1 : in STD_LOGIC;
    x2 : in STD_LOGIC;
    x3 : in STD_LOGIC;
    x4 : in STD_LOGIC;
    x5 : in STD_LOGIC;
    x6 : in STD_LOGIC;
    x7 : in STD_LOGIC;
    x8 : in STD_LOGIC;
    x9 : in STD_LOGIC;
    x10 : in STD_LOGIC;
    x11 : in STD_LOGIC;
    x12 : in STD_LOGIC;
    x13 : in STD_LOGIC;
    x14 : in STD_LOGIC;
    x15 : in STD_LOGIC;
    x16 : in STD_LOGIC;
    CLK : in STD_LOGIC;
    RESET : in STD_LOGIC;
    a1 : out STD_LOGIC;
    a2 : out STD_LOGIC;
    a3 : out STD_LOGIC;
    a4 : out STD_LOGIC;
    a5 : out STD_LOGIC;
    a6 : out STD_LOGIC;
    a7 : out STD_LOGIC;
    a8 : out STD_LOGIC;
    a9 : out STD_LOGIC;
    a10 : out STD_LOGIC;
    a11 : out STD_LOGIC;
    a12 : out STD_LOGIC;
    a13 : out STD_LOGIC;
    a14 : out STD_LOGIC;
    a15 : out STD_LOGIC;
    a16 : out STD_LOGIC;
    a17 : out STD_LOGIC;
    a18 : out STD_LOGIC;
    a19 : out STD_LOGIC;
    a20 : out STD_LOGIC
  );
end RALLY;
```

```

--}} End of automatically maintained section

architecture RALLY of RALLY is

signal D,Q : std_logic_vector(5 downto 0);
signal T: std_logic_vector(41 downto 0);

begin
  process(CLK, RESET)
    begin
      if (RESET='0') then
        Q<="000000";
      elsif (CLK'event and CLK='1') then
        Q<=D;
      end if;
    end process;

    t(0) <= (not(q(4)) and not(q(3)) and not(q(2)) and not(q(1))
and not(q(0)));
    t(1) <= (not(q(4)) and not(q(3)) and not(q(2)) and not(q(1))
and q(0) and x1);
    t(2) <= (not(q(4)) and not(q(3)) and q(2) and q(1) and
not(q(0)));
    t(3) <= (not(q(4)) and q(3) and not(q(2)) and not(q(1)) and
q(0));
    t(4) <= (not(q(4)) and not(q(3)) and not(q(2)) and not(q(1))
and q(0) and not(x1));
    t(5) <= (not(q(4)) and not(q(3)) and not(q(2)) and q(1) and
q(0));
    t(6) <= (not(q(4)) and not(q(3)) and not(q(2)) and q(1) and
not(q(0)) and x2);
    t(7) <= (not(q(4)) and not(q(3)) and not(q(2)) and q(1) and
not(q(0)) and not(x2));
    t(8) <= (not(q(4)) and not(q(3)) and q(2) and not(q(1)) and
not(q(0)) and x3 and x5);
    t(9) <= (not(q(4)) and not(q(3)) and q(2) and not(q(1)) and
q(0) and x4 and x5);
    t(10) <= (not(q(4)) and q(3) and not(q(2)) and not(q(1)) and
not(q(0)) and not(x7));
    t(11) <= (not(q(4)) and q(3) and not(q(2)) and q(1) and
not(q(0)) and x8 and x10);
    t(12) <= (not(q(4)) and q(3) and not(q(2)) and q(1) and q(0)
and x9 and x10);
    t(13) <= (not(q(4)) and q(3) and q(2) and q(1) and not(q(0))
and not(x12));
    t(14) <= (q(4) and not(q(3)) and not(q(2)) and not(q(1)) and
not(q(0)) and x13 and x15);
    t(15) <= (q(4) and not(q(3)) and not(q(2)) and not(q(1)) and
q(0) and x14 and x15);
    t(16) <= (q(4) and not(q(3)) and q(2) and not(q(1)) and
not(q(0)) and not(x16));
    t(17) <= (not(q(4)) and not(q(3)) and q(2) and not(q(1)) and

```

```

not(q(0)) and x3 and not(x5));
    t(18) <= (not(q(4)) and not(q(3)) and q(2) and not(q(1)) and
q(0) and x4 and not(x5));
    t(19) <= (not(q(4)) and not(q(3)) and q(2) and not(q(1)) and
not(q(0)) and not(x3));
    t(20) <= (not(q(4)) and not(q(3)) and q(2) and not(q(1)) and
q(0) and not(x4));
    t(21) <= (not(q(4)) and q(3) and not(q(2)) and q(1) and q(0)
and not(x9));
    t(22) <= (q(4) and not(q(3)) and not(q(2)) and not(q(1)) and
q(0) and not(x14));
    t(23) <= (not(q(4)) and q(3) and not(q(2)) and q(1) and
not(q(0)) and x8 and not(x10));
    t(24) <= (not(q(4)) and q(3) and not(q(2)) and q(1) and q(0)
and x9 and not(x10));
    t(25) <= (not(q(4)) and not(q(3)) and q(2) and q(1) and q(0)
and not(x6));
    t(26) <= (not(q(4)) and q(3) and q(2) and not(q(1)) and
not(q(0)));
    t(27) <= (not(q(4)) and q(3) and q(2) and q(1) and q(0));
    t(28) <= (not(q(4)) and not(q(3)) and q(2) and q(1) and q(0)
and x6);
    t(29) <= (not(q(4)) and q(3) and q(2) and not(q(1)) and q(0)
and x11);
    t(30) <= (not(q(4)) and q(3) and q(2) and not(q(1)) and q(0)
and not(x11));
    t(31) <= (q(4) and not(q(3)) and not(q(2)) and q(1) and
not(q(0)));
    t(32) <= (not(q(4)) and q(3) and not(q(2)) and not(q(1)) and
not(q(0)) and x7);
    t(33) <= (not(q(4)) and q(3) and q(2) and q(1) and not(q(0))
and x12);
    t(34) <= (q(4) and not(q(3)) and not(q(2)) and not(q(1)) and
not(q(0)) and x13 and not(x15));
    t(35) <= (q(4) and not(q(3)) and not(q(2)) and not(q(1)) and
q(0) and x14 and not(x15));
    t(36) <= (not(q(4)) and q(3) and not(q(2)) and q(1) and
not(q(0)) and not(x8));
    t(37) <= (q(4) and not(q(3)) and not(q(2)) and not(q(1)) and
not(q(0)) and not(x13));
    t(38) <= (q(4) and not(q(3)) and not(q(2)) and q(1) and
q(0));
    t(39) <= (q(4) and not(q(3)) and q(2) and not(q(1)) and
not(q(0)) and x16);
    t(40) <= (q(4) and not(q(3)) and q(2) and not(q(1)) and
q(0));

    process(T)
    begin
        D(0) <= ( t(0) or t(6) or t(8) or t(19) or t(18) or
t(32) or t(36) or t(23) or t(24) or t(33) or t(34) or t(37) or
t(35) or t(39));
        D(1) <= ( t(4) or t(6) or t(7) or t(17) or t(8) or

```

```

t(18) or t(9) or t(20) or t(2) or t(28) or t(10) or t(3) or t(11)
or t(36) or t(12) or t(30) or t(33) or t(13) or t(14) or t(34) or
t(15) or t(35) or t(22) or t(31) or t(16) or t(40));
    D(2) <= ( t(1) or t(7) or t(5) or t(17) or t(8) or
t(19) or t(18) or t(9) or t(20) or t(10) or t(11) or t(23) or
t(12) or t(21) or t(24) or t(30) or t(33) or t(13) or t(14) or
t(15) or t(31) or t(38) or t(39) or t(16) or t(40));
    D(3) <= ( t(25) or t(28) or t(32) or t(36) or t(23) or
t(21) or t(24) or t(26) or t(30) or t(33) or t(27) or t(31) or
t(40));
    D(2) <= ( t(7) or t(8) or t(9) or t(10) or t(11) or
t(12) or t(29) or t(13) or t(14) or t(34) or t(37) or t(15) or
t(35) or t(22) or t(38) or t(39) or t(16));

    a1 <= ( RESET and ( t(0) ) );
    a2 <= ( RESET and ( t(1) or t(2) or t(3) ) );
    a4 <= ( RESET and ( t(4) or t(5) ) );
    a3 <= ( RESET and ( t(6) ) );
    a20 <= ( RESET and ( t(7) or t(8) or t(9) or t(10) or
t(11) or t(12) or t(13) or t(14) or t(15) or t(16) ) );
    a7 <= ( RESET and ( t(17) or t(18) ) );
    a5 <= ( RESET and ( t(19) ) );
    a6 <= ( RESET and ( t(20) or t(21) or t(22) ) );
    a12 <= ( RESET and ( t(23) or t(24) ) );
    a8 <= ( RESET and ( t(25) or t(26) or t(27) ) );
    a10 <= ( RESET and ( t(28) ) );
    a15 <= ( RESET and ( t(29) ) );
    a13 <= ( RESET and ( t(30) or t(31) or t(40) ) );
    a9 <= ( RESET and ( t(32) ) );
    a14 <= ( RESET and ( t(33) ) );
    a17 <= ( RESET and ( t(34) or t(35) ) );
    a11 <= ( RESET and ( t(36) ) );
    a16 <= ( RESET and ( t(37) ) );
    a18 <= ( RESET and ( t(38) ) );
    a19 <= ( RESET and ( t(39) ) );
end process;
end RALLY;

```