

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

факультет прикладних інформаційних технологій та електроінженерії

(повна назва факультету)

кафедра автоматизації технологічних процесів і виробництв

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: «Розроблення системи управління транспортними платформами
на базі програмованого логічного контролера Siemens SIMATIC S7-300»

Виконав(ла): студент(ка) IV курсу, групи КА-41
спеціальності 151 «Автоматизація

та комп'ютерно-інтегровані технології»

(шифр і назва спеціальності)

	<u>Громовик М.З.</u>
(підпис)	(прізвище та ініціали)
Керівник	<u>Трембач Р.Б.</u>
(підпис)	(прізвище та ініціали)
Нормоконтроль	<u>Козбур В.Р.</u>
(підпис)	(прізвище та ініціали)
Завідувач кафедри	<u>Савків В.Б.</u>
(підпис)	(прізвище та ініціали)
Рецензент	<u>Митник М.М.</u>
(підпис)	(прізвище та ініціали)

Тернопіль
2022

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет прикладних інформаційних технологій та електроінженерії
(повна назва факультету)

Кафедра автоматизації технологічних процесів і виробництв
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Савків В.Б.

(підпис)

(прізвище та ініціали)

« ____ » _____ 2022р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавр

(назва освітнього ступеня)

за спеціальністю 151 «Автоматизація та комп'ютерно-інтегровані технології»

(шифр і назва спеціальності)

студенту Громовик Михайлу Зеновійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи «Розроблення системи управління транспортними платформами на базі програмованого логічного контролера Siemens SIMATIC S7-300»

Керівник роботи к.т.н., доцент Трембач Р.Б.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «27» січня 2022 року № 4/7-40

2. Термін подання студентом завершеної роботи 16 червня 2022 року

3. Вихідні дані до роботи Технічні характеристики периметру

4. Зміст роботи (перелік питань, які потрібно розробити)

1) аналітична частина; 2) проектна частина;

3) спеціальна частина; 4) Безпека життєдіяльності, основи охорони праці.

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

Презентація кваліфікаційної роботи 14 аркушів формату А4

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ	5
ВСТУП	6
1 АНАЛІТИЧНА ЧАСТИНА	8
1.1 Опис алгоритму управління транспортними платформами на базі ПЛК SIMATIC S7-300	8
1.2 Аналіз програмного забезпечення для побудови автоматизованої системи управління на базі S7 – 300	16
1.3 Загальні принципи програмування з допомогою LAD	18
2 ПРОЕКТНА ЧАСТИНА	20
2.1 Опис алгоритму функціонування системи	20
2.2 Етапи створення проекту S7	24
2.3 Тестування програми	29
2.4 Відладка створеної програми	30
2.5 Створення діаграми входів і виходів для двигуна	
3 СПЕЦІАЛЬНА ЧАСТИНА	35
3.1 Конфігурація і параметризація модулів	35
3.2 Характеристика компонентів мови LAD	41
3.3 Опис процесу управління транспортною платформою	46
4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	58
4.1 Значення охорони праці для забезпечення безпечних умов праці	58
4.2 Аналіз потенційних небезпек та шкідливостей виробничої сфери	59
4.3 Електромагнітне й іонізуюче випромінювання	61
4.4 Забезпечення нормальних умов праці	62
4.5 Розрахунок напруги дотику	67
ВИСНОВКИ	70
ПЕРЕЛІК ПОСИЛАНЬ	71

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

- ВДТ – відео термінал
- ГВК – гірниче виробництво кераміки
- ЕОС – електронно-обчислювальна система
- ЕОМ – електронно-обчислювальна машина
- ОЦ – обчислювальний центр
- ПЗ – програмне забезпечення
- ПЛК – програмований логічний контролер
- BСD – Binary Coded Decimal (двійково-десяткові числа)
- CPU – Central Processing Unit (центральний процесорний пристрій)
- DB – Data Block (блок даних)
- FB – Function Block (функціональний блок)
- FBD – Function Block Diagram (діаграма функціональних блоків)
- FC – Function (функція)
- FM – Function Modules (функціональний модуль)
- LAD – Ladder Logic (контактний план)
- MPI – Multipoint Interface (мультиточковий інтерфейс)
- NC – Normal Close Contact (нормально закритий контакт)
- NO – Normal Open Contact (нормально відкритий контакт)
- OB – Organization Block (організаційний блок)
- SCL – Structured Control Language (структурована мова управління)
- SDB – System Data Block (системні блоки даних)
- SFC – System Function (системна функція)
- SFB – System Function Blocks (системний функціональний блок)
- STL – Statement List (список операторів)
- VAT – Variable Table (таблиця змінних)

ВСТУП

Сучасний стан розвитку систем автоматизації характеризується широким впровадженням засобів автоматизації на основі мікропроцесора: розумних датчиків, пристроїв управління, функціональних блоків, пристроїв відображення інформації, панелей оператора та інших засобів.

Особливе місце в цьому списку займає промисловий мікропроцесорний контролер. Починаючи з початку 1970-х років, вони швидко завоювали репутацію серед професіоналів з автоматизації. Це пояснюється високою надійністю, програмуванням, відносною простотою використання та розширенням системи управління. Це заклало основу для впровадження нового покоління систем автоматизації — інтегрованої комп'ютерної системи керування.

В даний час пропонується велика кількість контролерів вітчизняного та зарубіжного виробництва. Контролер Siemens SIMATIC S7-300 дуже популярний на вітчизняних підприємствах.

SIMATIC S7-300 - це модульний програмований контролер, розроблений для систем автоматизації низької та середньої складності.

Модульна конструкція, природне охолодження, можливість використання локальних і розподілених структур вводу-виводу, широкі комунікаційні можливості та багато функцій, які підтримуються на рівні операційної системи, простота використання та обслуговування забезпечують економічні рішення для різних типів систем автоматичного керування.

Ефективному використанню контролерів сприяє використання декількох типів центральних процесорів з різною продуктивністю, широкий спектр модулів введення/виводу окремих і аналогових сигналів, функціональних модулів, комунікаційних процесорів.

Метою кваліфікаційної роботи є розробка програмного забезпечення для управління транспортною платформою на базі програмованого логічного контролера Simatic STEP-7 S7-300.

При автоматизації цього процесу спостерігається загальне вдосконалення системи, тому не потрібно постійного втручання людини в систему. Підвищується надійність і швидкість процесу.

1 АНАЛІТИЧНА ЧАСТИНА

Завданням роботи є розробка алгоритму маневрування транспортною платформою маневреного пристрою з використанням сучасної системи автоматизації SIMATIC на базі програмованого логічного контролера SIMATIC S7-300.

Алгоритм, розроблений для системи керування маневровою платформою на базі ПЛК SIMATIC S7-300, який безпосередньо відображений у цій роботі, може бути використаний на підприємствах для заміни існуючої системи керування релейним маневровим обладнанням.

1.1 Опис алгоритму управління транспортними платформами на базі ПЛК SIMATIC S7-300

Об'єктом управління є маневрені платформи, що транспортуються по дорозі шляхом зупинки транспортних візків, які приводять в рух редукторні станції на базі мотор-редукторів зі спеціальними механізмами для переміщення цих будівельних візків, які є барабанами. Сталь обмотана тросом, який безпосередньо з'єднаний з транспортним візком транспортної платформи. Рульове колесо сталевого троса, що переміщує захватне колесо, приводиться в рух черв'ячним редуктором.

Маневровий пристрій призначений для транспортування виготовленого виробу на спеціальній транспортній платформі (візок печі), у цьому випадку вивантажена транспортна платформа переміщується з печі на керамічну облицювальну цеглу, вивантажуючи її, потім спалюючи безпосередньо до кінця пакування, а потім перезавантаження. Продукція, що випускається, являє собою набір приводних станцій на базі мотор-редукторів, транспортні платформи (котельні візки), транспортні колійні системи, кінцеві вимикачі (датчики положення), практично маневрені пристрої.

Виходячи з аналізу, наша робота з управління транспортною платформою полягає в тому, щоб вчасно контролювати двигуни змінного струму, тобто вводити та виводити їх у правильний час, тим самим генеруючи рух транспортної платформи. Загалом можна сказати, що SIMATIC S7-300 дозволяє реалізувати програмований логічний контролер для вирішення цієї проблеми автоматизації та створення ефективного алгоритму системи управління.

Функціональна схема платформи маневреного транспортного засобу наведена на рисунку 1.1. Загальний принцип роботи маневрових пристроїв (транспортних майданчиків) існуючої системи керування на підприємстві такий: після випікання на вагонетці печі (транспортній платформі) керамічна облицювальна цегла виходить з однієї з двох печей і падає на одному з них транспортні маршрути.

На основі релейно-контактної схеми керування оператор, що знаходиться біля ПКП, розподіляє ці візки між іншими транспортними шляхами (рисунок 1.1) таким чином, щоб котли з продукцією, що виробляється на всіх транспортних маршрутах, рівномірно завантажувалися, а потім розвантажувалися. (транспортна палуба) Розвантаження, подальше пакування, після розвантаження порожні візки повертаються на резервний транспортний шлях для подальшого відвантаження керамічної цегли, а потім для випікання.

Рух пічних вагонеток здійснюється за допомогою перемінних мотор-редукторів рульових станцій, з'єднаних безпосередньо з інвалідним візком за допомогою сталевого троса, обмотаного навколо керма при запуску двигуна.

Для більш детального пояснення алгоритму управління транспортною платформою наведемо розроблену функціональну схему управління як окрему зону, яка послідовно визначає відповідні структурні елементи для кожної зони, і ця функціональна схема показана на рисунку 1.2.

Область 1 складається з наступних структурних елементів:

- D1-D6 (колія 1), D3-D7 (колія 2), D2-D8 (колія 3) - черв'ячні мотор-редуктори рульових станцій, що використовуються для приводу транспортної платформи (пічної тачки), що рухається вздовж транспортного засобу колія (1-я

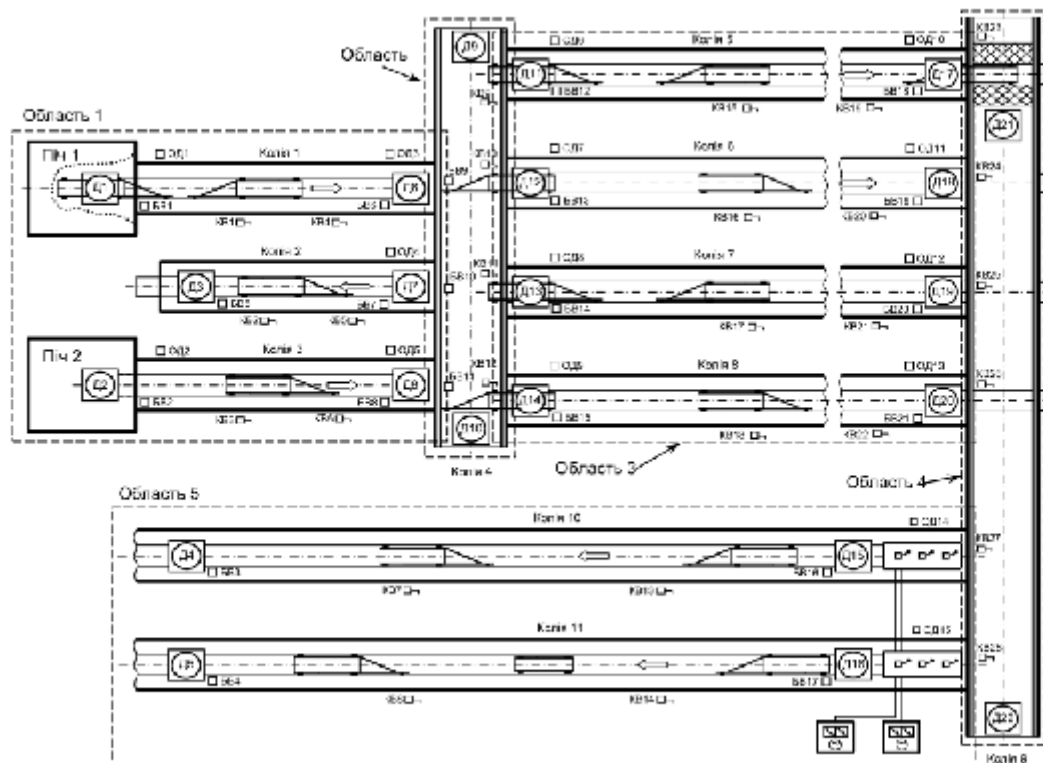


Рисунок 1.1– Функціональна схема управління маневровими пристроями із виділенням окремих областей управління

колія, 2-а колія, 3-я колія) і призначена для транспортування візка (транспортної платформи) від точки виходу котла до кінцевої точки розвантаження та пакування, а приводом будівельного візка є безпосередньо з'єднаний сталевий трос, привід станції на основі колеса, обмотаного сталевим тросом, для запуску черв'ячного мотор-редуктора і приводного мотор-редуктора, отже, в результаті взаємопов'язаної роботи пускових станцій на базі двигуна-редуктора, розташованого на протилежному боці транспортної дороги, буде здійснюватися переміщення будівельного візка по відповідній дорозі та перевезення пічної вагонетки;

- OD1-OD3 (колія 1) - оптичні датчики положення, розташовані на протилежній стороні шляху 1, виявляють наявність транспортної платформи при виході з фази 1 і наближенні до шляху 4;

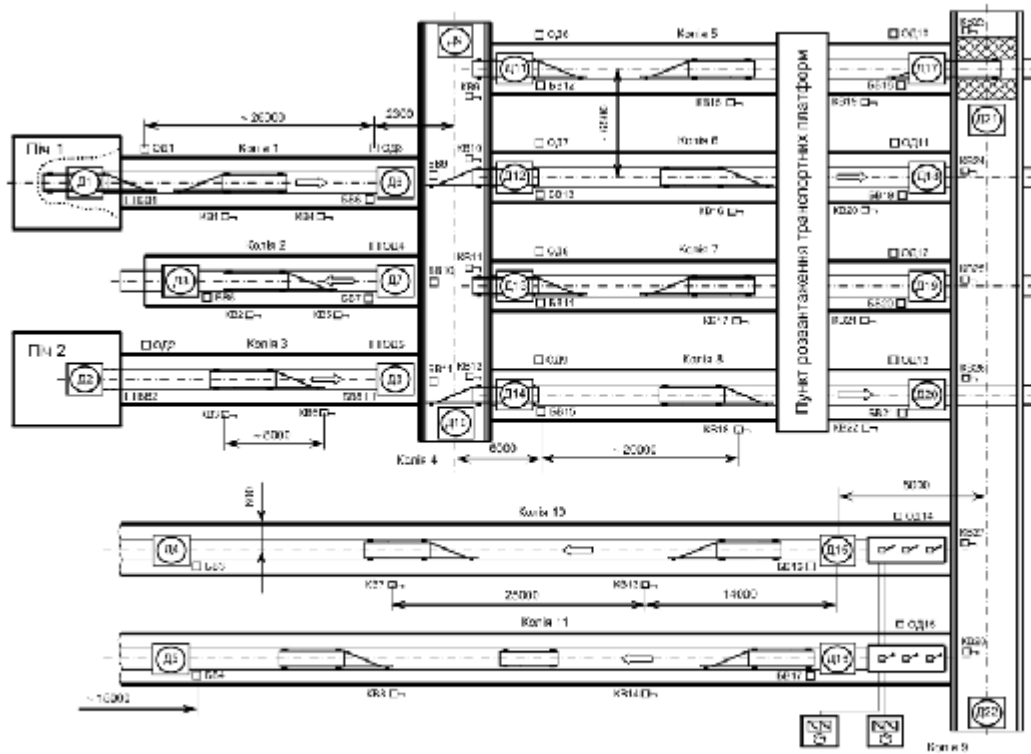


Рисунок 1.2 – Розроблена функціональна схема управління транспортними платформами

- OD4 (колія 2) - оптичний датчик положення виявляє наявність транспортної платформи в кінці смуги 2, коли вона знаходиться поблизу смуги 4;

- OD2-OD5 (колія 3) - оптичний датчик положення, розташований у полюсних точках маршруту 3 для корекції транспортної платформи при її виході з 2-ї печі та наближенні до полярного положення маршруту 3;

- KB1-KB4 (колія 1), KB2-KB5 (колія 2), KB3-KB6 (колія 3) - кінцевий вимикач (датчик положення контакту) для зупинки руху транспортної платформи. датчик.

Область 2 містить наступні структурні елементи:

- Д9-Д10 (колія 4) - черв'ячний мотор-редуктор (колія 5), колія 6, колія 7, колія 8, призначений для переміщення електровізка, що завантажує виготовлений виріб (пічний візок) на відповідну транспортну смугу по колії 4;

- BV9, BV10, BV11 (колія 4) - безконтактний перемикач (датчик положення), призначений для зупинки руху електромобілів на протилежному боці доріжок 1, 2 і 3, завантаження їх у вагонетку котла та руху електричний візок на його ділянці у разі зупинки перед дорогою;

- KB9, KB10, KB11, KB12 (колія 4) - кінцевий вимикач (датчик положення контакту) для подальшого зниження положення протилежної сторони 5, 6, 7 та 8 смуг руху вантажного електромобіля. Візок реагує на завантаженої гальванічної пластини, і це положення спричиняє закріплення датчика перед однією із смуг у разі прямого (механічного) контакту з датчиком.

Розглянемо основні структурні елементи області 3:

- OD6-OD10 (колія 5), OD7-OD11 (колія 6), OD8-OD12 (колія 7), OD9-D13 (колія 8) - оптичні датчики положення для виявлення та сигналізації наявності пічних вагонеток (транспортної платформи) на протилежному кінці відповідного транспортного маршруту;

- KB12- KB 18 (колія 5), KB 13- KB 19 (колія 6), KB 14- KB 20 (колія 7), KB15- KB 21 (колія 8) - безконтактний датчик положення, що використовується для визначення крайнього (кінцевого) діапазону руху для захоплення транспортної платформи, спрацювання яких відбувається при вході зупиняючого крісла в зону їх дії, що призводить до негайного припинення руху візків по транспортній дорозі;

- KB15-KB19 (колія 5), KB16-KB20 (колія 6), KB17-KB21 (колія 7), KB18-KB22 (колія 8) - кінцевий вимикач для зупинки руху транспортної платформи (контактний датчик положення) за відповідним маршрутом безпосереднього механічного контакту з ними;

- D11-D17 (колія 5), D12-D18 (колія 6), D13-D19 (колія 7), D14-D20 (колія 8) - транспортні маршрути.

Область 4 складається із таких структурних елементів:

- KB23, KB24, KB25, KB26, KB27, KB28 (колія 9) - контактні обмежувачі (датчики положення), що використовуються для визначення розташування електромобілів на протилежній стороні відповідної смуги (колії 5, колії 6, колії 7, колії 8, колії 15, колії 16);

- Д21-Д22 (колія 9) - мотор-редуктори для станцій керування, призначені для приводу електролафета вздовж транспортної колії 9.

Область 5 містить наступні структурні елементи:

- Д4-Д15 (колія 10), Д5-Д16 (колія 11) - черв'ячний двигун-редуктор транспортної станції розвантаженої транспортної платформи (печного візка) з подальшим завантаженням керамічною облицювальною цеглою;

- БВ3-БВ16 (колія 10), БВ4-БВ17 (колія 11) - безконтактний датчик положення для визначення кінцевої межі руху інвалідного візка;

- KB7-KB13 (колія 10), KB8-KB14 (колія 11) - контактні кінцеві вимикачі, призначені для зупинки руху транспортної платформи при безпосередньому контакті з цим датчиком.

Алгоритм управління транспортною платформою дозволяє запровадити систему управління на базі ПЛК SIMATIC S7-300, що дозволяє виконувати повний процес автоматизації маневрового пристрою без втручання в роботу людини-оператора. вантажні платформи на відповідних коліях.-ручна ланцюг контактної схеми. В принципі, система управління на базі SIMATIC S7-300 PLC дозволяє залучити лише одного висококваліфікованого оператора для моніторингу технологічного процесу транспортування та виконання технічного обслуговування.

Блок-схема системи управління транспортною платформою з використанням програмованого логічного контролера SIMATIC S7-300 показана на рисунку 1.3.



Рисунок 1.3 – Загальна структурна схема системи управління транспортними платформами на основі ПЛК SIMATIC S7-300

Після закінчення процесу випікання керамічної фінішної цегли завантажена платформа з готовим керамічним виробом відсувається за допомогою спеціального штовхача візка до виходу з 1-ї або 2-ї печі і перевіряється на наявність транспортної платформи з виробленим продуктом, піч з оптичним датчиком положення OD1 (OD2), потім сигнал F90 передається на станцію розподіленого вводу-виводу ET-200M через пристрій зв'язку (AS-інтерфейс) за допомогою функціонального модуля введення-виведення, а потім аналізується програмованим логічним контролером SIMATIC S7-300 (блок-схема системи управління) (показано на рисунку 1.2), у свою чергу, формує сигнал, який подається на перетворювач частоти MICROMASTER 420, який активує активатор двигуна-редуктора D6 (D8), по виділених транспортних смугах (колії 1, 2) і за відповідними сигналами повинні працювати транспортні платформи, споруджуються тролейбуси, рух продовжується до подачі сигналу про його зупинку. , в цьому випадку BV1 (BV2) має сигнал на ПЛК SIMATIC S7-300 при підключенні до безконтактного датчика положення, він аналізує це і посилає керуючий сигнал на частотний перетворювач двигуна-редуктора D1 (D2), який в свою чергу вимикає ці двигуни, зупиняє рух інвалідного візка по дорозі і автоматично захоплює коляску печі. Це тому, що датчик BV1 (BV2) визначає положення транспортного візка в певній зоні.

Після захоплення транспортних платформ ПЛК SIMATIC S7-300 отримує керуючий сигнал для запуску передач G6 (D8) рульових станцій, які транспортуватимуть захоплені транспортні платформи по смугах 1 і 3. , при підключенні до вбудованої транспортної платформи KB3 від датчика на ПЛК SIMATIC S7-300 формується сигнал, який зупиняє мотор-редуктор D6 (D8).

Після того, як тяговий візок звільнить транспортну платформу, ПЛК SIMATIC S7-300 запускає мотор-редуктор D1 (D2), переміщує візок у протилежному напрямку до виходу печі 1 (печі 2) і зупиняє рух візка. Коли ви маєте справу з датчиком BV1 (BV2), який переміщено на доріжці 1 (доріжка 3), тримайте його в присутності візка для печі з керамічної цегли.

При цьому інший транспортний тролейбус тримає тролейбус попереднього переміщення, тому що на одній колії є два тролейбуси, тому на одному транспортному маршруті можна одночасно перевозити два візки з виготовленим виробом.

Таким чином, коли раніше переміщений візок утримується другим утримуючим візком, можна утримувати візок, якщо перший візок знаходиться на виході з 1-ї печі (печі 2), а потім подавати сигнал від SIMATIC S7-300 PLC до перетворювач частоти. Двигун-редуктор D6 (D8) запускав привід цих приводних двигунів, в результаті чого побудовані візки транспортуються по 1-й смузі (3-й смузі).

Коли платформа контактує з обмежувачем KV4 (KV6), двигуни D6 (D8) автоматично зупиняються за сигналом, що відповідає SIMATIC S7-300 PLC, який потім відпускається негайно, коли завантажені візки ловляться.

У цьому випадку навантажений візок знаходиться в кінці колії 1 (колії 2), внаслідок чого ПЛК SIMATIC S7-300 припиняє переміщення інших транспортних платформ до тих пір, поки завантажений візок не залишить піч 1 (колія 2). Інвалідні візки можна перевозити з колії 1 на одну з чотирьох колій (колія 5, колія 6, колія 7, колія 8) до кінця колії 3.

ПЛК SIMATIC S7-300 постійно аналізує інформацію, що надходить від датчика розташування і генерує керуючий сигнал, який подається на відповідний активатор, тому SIMATIC S7-300 PLC регулярно обробляє та аналізує вхідну інформацію.

1.2 Аналіз програмного забезпечення для побудови автоматизованої системи управління на базі S7 – 300

Система автоматизації SIMATIC об'єднує всі підсистеми, що використовуються для вирішення проблем автоматизації, від рівня об'єкта до управління процесами, в однорідну архітектуру системи. Програмований

контролер серії S7-300 відповідає всім потребам автоматизації процесів і виробництва - модульний масштабований контролер призначений для використання як в системах з низьким попитом, так і з високою потужністю.

STEP 7 — програмне забезпечення для розробки системного програмного забезпечення SIMATIC. Програмне забезпечення для блокового програмування кроку 7 використовує такі мови програмування: STL (список операторів - список операторів або мнемонічний список), LAD (покрокова логіка або діаграма кроків - план контактів; представлення схоже на релейну логічну схему; багатокрокова діаграма), FBD (функціональна блок-схема - схема або функціональний план функціональних блоків) і пакет SCL (structured management language - структурована мова управління) [2].

Інструмент SIMATIC Manager є основним інструментом у КРОКУ 7. Під час першого запуску з'явиться вікно Майстра проекту.

Як правило, програмне забезпечення ЦП складається з операційної системи та спеціальної програми.

Операційна система — це сукупність усіх системних ресурсів, які використовують ці ресурси, а також усіх інструкцій та пояснень, які керують цим процесом. Він включає в себе такі функції, як резервне копіювання даних у разі збою живлення та активація класів пріоритету. Операційна система є компонентом ЦП, і користувач не має доступу до режиму запису. Однак ви можете перезавантажити картку пам'яті, наприклад, якщо оновити програмне забезпечення.

Пристрої SIMATIC S7 мають пріоритети. У деяких класах пріоритету ви можете змінити пріоритет під час встановлення параметрів ЦП. У таблиці наведено найнижчий і найвищий з можливих класів; Кожен процесор має свої власні пріоритети.

Параметр CPU визначає, який із можливих класів пріоритету буде використаний. Пріоритетним класам, які не зачіпаються, слід надати нульовий пріоритет. Для всіх використовуваних пріоритетних класів програмуються

відповідні організаційні підрозділи; інакше ЦПІ згенерує ОВ 85 ("Помилка програми") або перейде в режим STOP [2].

Достатня кількість пам'яті має бути виділено для тимчасових локальних даних (L-стек) для кожної вибраної категорії пріоритету.

Програму можна розділити на будь-яку кількість розділів, щоб її було легше читати та розуміти. Мова програмування STEP 7 підтримує цю концепцію та надає необхідні функції. Кожна частина програми повинна бути незалежною, технологічно та оперативно обґрунтованою. Ці частини програми називаються «блоками». Блок — це частина програми, яка визначається його функцією, структурою та проблемами.

1.3 Загальні принципи програмування з допомогою LAD

Програма складається з окремих елементів LAD, з'єднаних послідовно або паралельно. Контактна схема схожа на електричну. План контактів або програмування зв'язку починається з лівої несучої або лівої шини живлення. Вам потрібно вибрати місце, куди ви хочете вставити елемент за посиланням, і вибрати потрібний програмний елемент:

- за допомогою відповідної функціональної клавіші;
- за допомогою натиснення відповідної кнопки у полі функції;
- з каталогу програмних елементів.

Каталог програмного забезпечення підтримує програмування LAD і FBD і надає наявну графіку, а також окремі контейнери блоків, кілька програмованих версій і блоки, які вже знаходяться в доступних бібліотеках.

Ланка закінчується котушкою або прямокутним блочним елементом. Більшість програмних елементів повинні виділяти елементи пам'яті. Найпростіший спосіб зробити це – спочатку створити всі елементи програмного забезпечення, а потім позначити їх.

У цій частині роботи виконується аналіз програмного забезпечення для розробки програмного забезпечення для управління транспортною платформою

PLC S7-300 на основі Step7. Це програмне забезпечення відрізняється від звичайних мов програмування (C++, Delphi, Java та ін.) тим, що в якості оператора програмування використовується план комунікації (логічні елементи), що значно спрощує розробку програми.

2 ПРОЕКТНА ЧАСТИНА

2.1 Опис алгоритму функціонування системи

Маневруючий пристрій призначений для пакування цегли після випікання в печі. Схема показана на рисунку 2.1. Технологічний процес виглядає наступним чином. Після випікання на інвалідному візку цеглини падають на одну з доріжок. Оператор, що стоїть на пульті керування, розподіляє ці коляски між чотирма коліями так, щоб усі колії були рівномірно завантажені. Потім цеглу вивантажують з візка на конвеєр, упаковують, а порожній візок повертають на тротуар для наступного завантаження та зважування цегли.

Вагонетки приводяться в рух по черзі електродвигунами. Принцип дії двигуна змінного струму заснований на взаємодії магнітного потоку, створюваного ротором (рухомою частиною), обертового магнітного поля, створюваного обмоткою статора (нерухомою частиною двигуна). Електродвигун, швидкість обертання ротора якого дорівнює швидкості магнітного поля статора, називається синхронним двигуном.

Електродвигун, швидкість обертання ротора якого менша за швидкість магнітного поля статора, називається асинхронним двигуном. Синхронні двигуни потужністю до сотень Вт використовуються для автоматики, коли необхідно підтримувати постійну швидкість. Їх ротори виготовляють із постійних магнітів, циліндри з магнітотвердих сплавів з широкими петлями гістерезису або профільні циліндри з магнітного матеріалу з короткозамкненими котушками для полегшення запуску двигуна.

Отже, синхронний двигун працює асинхронно, поки ротор не досягне синхронної швидкості. Двигуни змінного струму надійніші за двигуни постійного струму, довговічні, мають широкий діапазон обертів валу, низьке тертя та відсутність іскор. При цьому вони менші за двигуни постійного струму

за пусковим моментом і ККД, мають більший розмір і більш високу частоту напруги живлення.

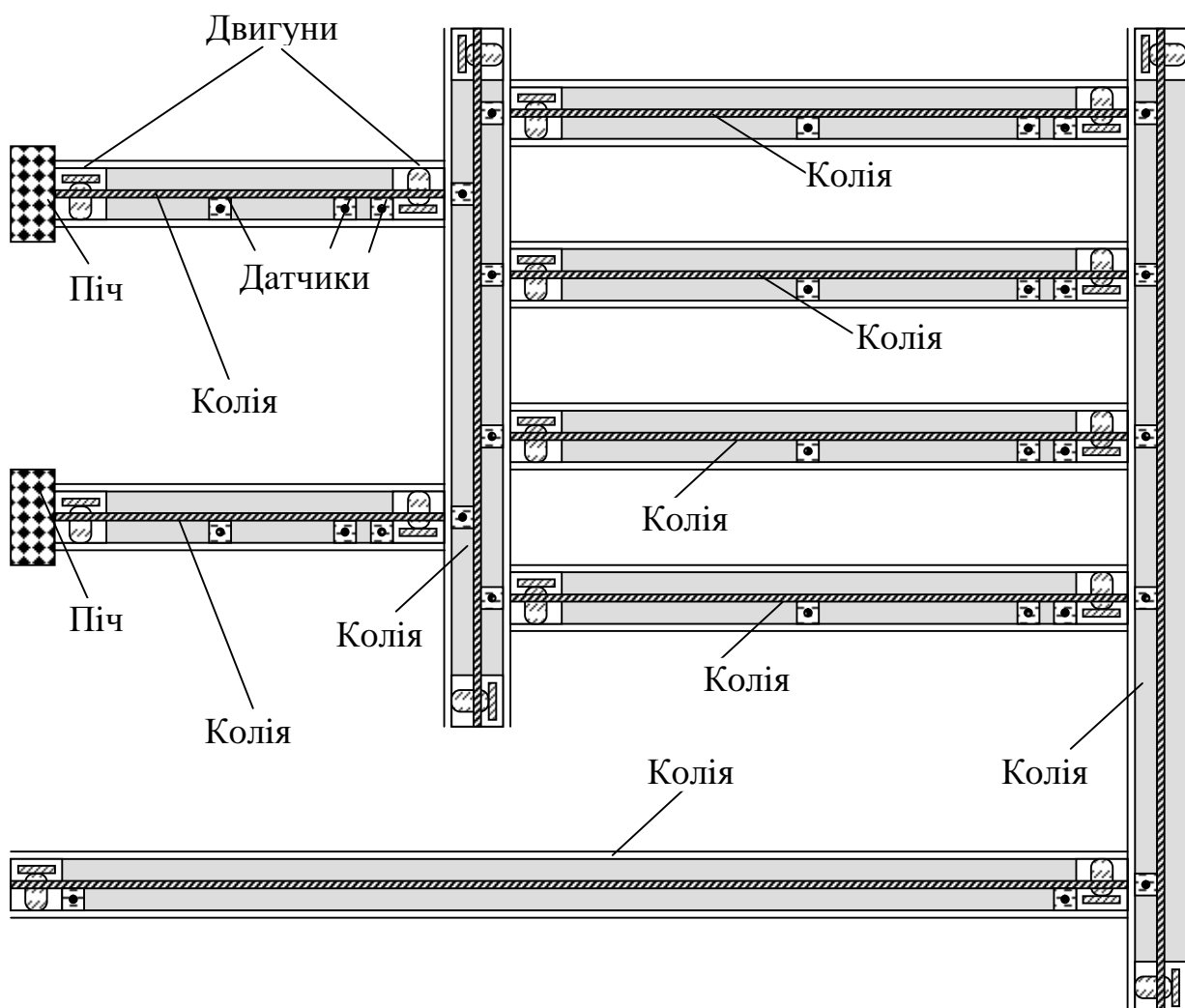


Рисунок 2.1 – Схема технологічного процесу

Асинхронні двигуни змінного струму мають кращі динамічні характеристики, ніж синхронні двигуни, мають широкий діапазон регулювання швидкості та зазвичай використовуються у високошвидкісних регульованих електроприводах.

Залежно від кількості фаз асинхронні двигуни поділяють на однофазні, трифазні та двофазні. Однофазний двигун малопотужний і використовується як нерегульований привід вентилятора. Трифазні двигуни мають високу потужність (до тисячі Вт), але мають складну схему керування (вимагає

використання трифазного генератора), тому використовуються лише для нерегульованих електроприводів. Двофазні двигуни відносно легко регулюють швидкість і крутний момент, тому їх часто використовують в системах автоматичного керування.

Для транспортування візків використовуються технічні асинхронні двигуни змінного струму, як показано в таблиці 2.1.

Таблиця 2.1 – Технологічні параметри двигуна змінного струму

Тип двигуна	Кількість обертів	ККД	$\cos \varphi$	Потужність	Напруга	Сила струму
Асинхронний	675об/хв	75	0,73	1,5 кВт	380 В	4,1А

Управління двигуном здійснюється за допомогою панелі керування з кнопками «Пуск» і «Стоп» (S1 і S2).

Функціональна схема автоматики показана на рисунку 2.2.

Алгоритм роботи такий.

Наявність цегляного візка визначають за допомогою фотодатчика F1 (F2). Після появи візка сигнал передається на контролер S7-300, який відповідно до поставленого завдання формує сигнал запуску на активатор M1 (двигун) (M3). Двигун працює до тих пір, поки не отримає сигнал зупинки від кільця. Після зупинки двигуна M1 (M3) M2 (M4) сигналізує двигуну про запуск, що переміщує візок до кінця колії. Після двічі запуску двигуна M1 (M3) контролер генерує сигнал для запуску двигуна M5.

Двигун M5 зупиняє тягач, який рухає візок по дорозі, де встановлений цегляний візок, і зупиняє його за допомогою кільця. Потім двигун M2 (M4) знову запускається, і інвалідний візок виїжджає на пішохідний перехід (смуга 3). Після отримання сигналу зупинки двигуна на доріжці 3 (рисунок 2.1)

контролер формує сигнал запуску двигуна М1 (М3). Потім він визначає, яка з наступних 4 смуг вільна, і включає двигун М5 або М6, залежно від отриманого сигналу.

Вагонетка доставляється на одну з вільних колій (колії 4, 5, 6, 7) і вмикається відповідний двигун (М7, М8, М9, М10) до 3-ї смуги, потім в'їжджає вагонетка на 3-тю колію, зупиняє двигун (М7, М8, М9, М10) кільцем і сигналізує контролеру запуснути двигун на протилежному боці дороги (М12, М13, М14, М15).

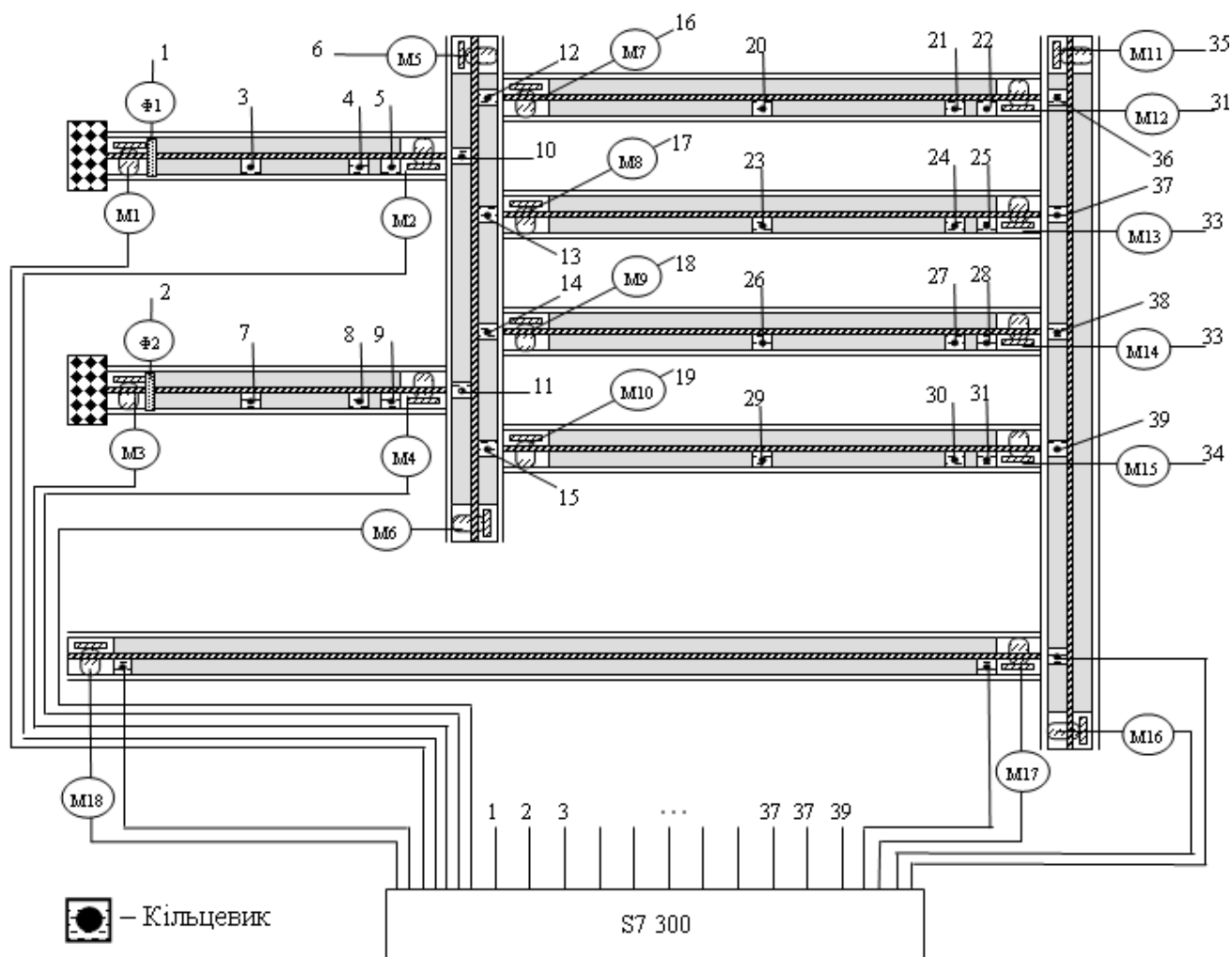


Рисунок 2.2 – Функціональна схема управління

Двигуни зупиняються кільцем посередині шляху розвантаження цегли. Після розвантаження цегли в кінці дороги (M12, M13, M14, M15) подається сигнал запуску двигуна. Порожній візок відвезуть до кінця дороги, а потім переведуть на маршрут 9 для випікання в цеглині та печач.

2.2 Етапи створення проекту S7

2.2.1 Конфігурація станцій

Для створення програмованих параметрів керування використовуються інструменти конфігурації обладнання. Конфігурація виконується в автономному режимі без підключення ЦП. Ви також можете використовувати цей інструмент для позначення та параметризації модулів. Конфігурація обладнання починається з вибору станції, перейдіть до меню Правка -> Відкрити об'єкти або двічі клацніть об'єкт Hardware у відкритому контейнері SIMATIC 300 Station (рисунок 2.3).

Після налаштування скористайтеся командою Station -> E Consistency Check, щоб переконатися, що введена інформація правильна.

Команда Station -> Save записує всі дані значень параметрів проекту в таблицю конфігурації жорсткого диска [3].

Команда Station -> Save and Compile не тільки зберігає таблиці конфігурації, але й записує скомпільовані дані в об'єкт системних даних в окремому контейнері Blocks.

Апаратна конфігурація генерує належним чином скомпільовану контрольну суму станції та зберігає її в системних даних. Та сама конфігурація системи має ту саму контрольну суму.

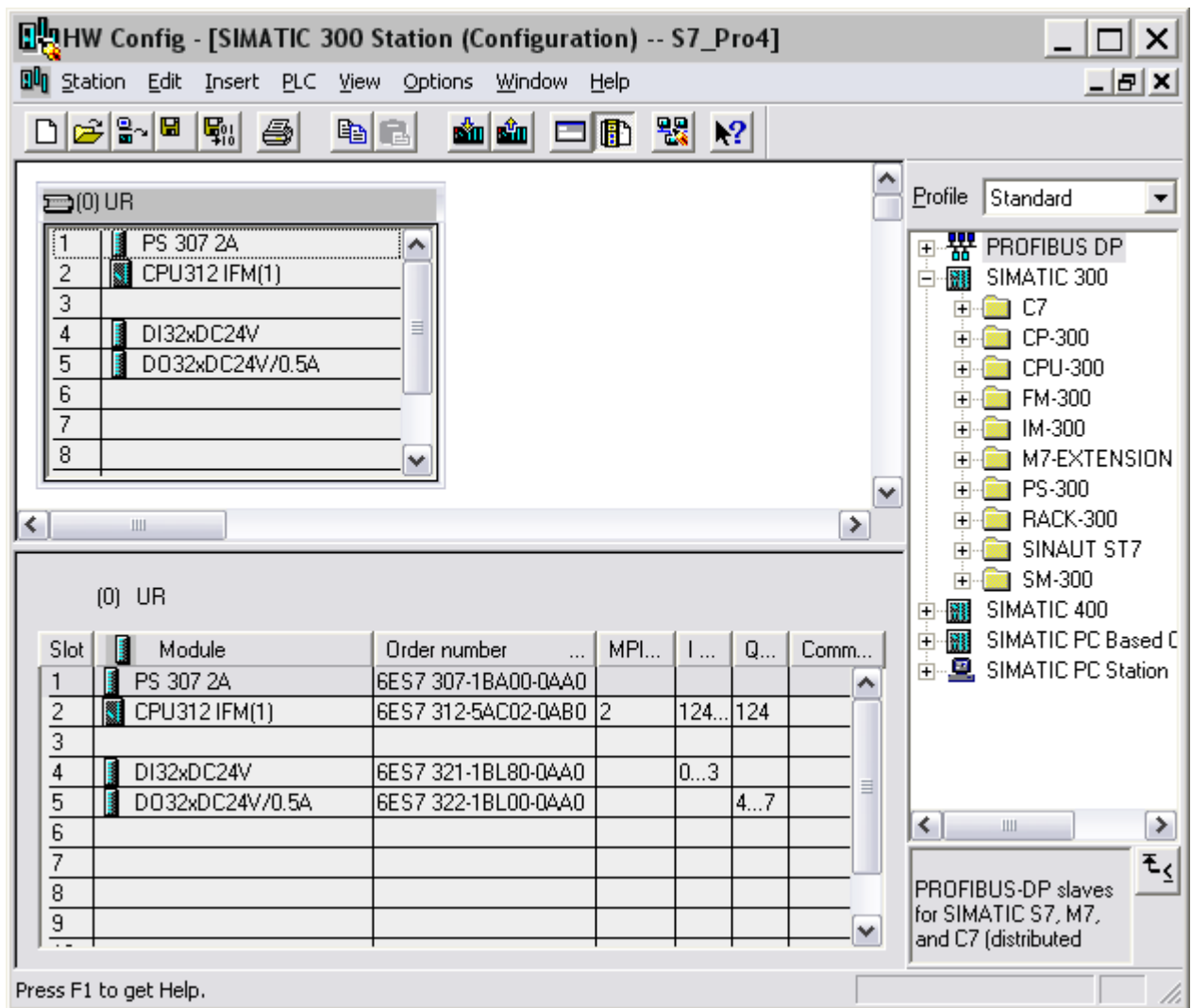


Рисунок 2.3 – Вікно станції в Hardware Configuration

Конфігурація починається з вибору рейки маршрутизатора з каталогу модулів, наприклад, під «SIMATIC 300», утримуйте її, перетягніть мишу у верхню частину вікна станції та помістіть її кудись у цьому вікні (перетягніть і опустіть). Показано порожню таблицю конфігурації для центральної полиці. Потім виберіть потрібні модулі з каталогу модулів і помістіть їх у відповідні слоти, як описано вище. Знак «Паркувати заборонено» вказує на те, що обраний модуль не можна помістити в слот.

Для однорядної конфігурації станції S7-300 слот 3 залишається порожнім; Це для інтерфейсного модуля полиці розширення.

Під час налаштування модулів інструмент налаштування апаратного забезпечення автоматично призначає початкову (початкову) адресу модулям. Цю адресу можна побачити у властивостях об'єкта відповідних модулів у нижній частині вікна станції.

2.2.2 Конфігурація мережі

Мережева система S7 є основою системи SIMATIC. Необхідними об'єктами є підмережі та модулі з комунікаційними можливостями на станціях. За допомогою SIMATIC-менеджера ви можете створювати нові підмережі та станції в ієрархії проекту. Можуть бути додані модулі (ЦП), які потім можуть обмінюватися даними через інструмент налаштування обладнання. Потім за допомогою інструмента «Конфігурація мережі» в таблиці підключень визначте зв'язки з'єднання цих модулів.

Конфігурація мережі дозволяє графічно представити параметри мережі та їх вузли. Ви можете використовувати параметри мережі для створення всіх необхідних підмереж і станцій; потім розподіліть станції до підмережі та параметризуйте характеристики вузлів модулів із підтримкою зв'язку [3].

На рисунку 2.4 показано приклад вікна конфігурації мережі. У верхній частині цього вікна знаходяться всі раніше створені підмережі та станції (вузли) з налаштованими підключеннями. Якщо у верхній частині вікна вибрано модуль із «комунікаційними можливостями», наприклад, CPU S7-400, таблиця підключень буде відображена в нижній частині вікна.

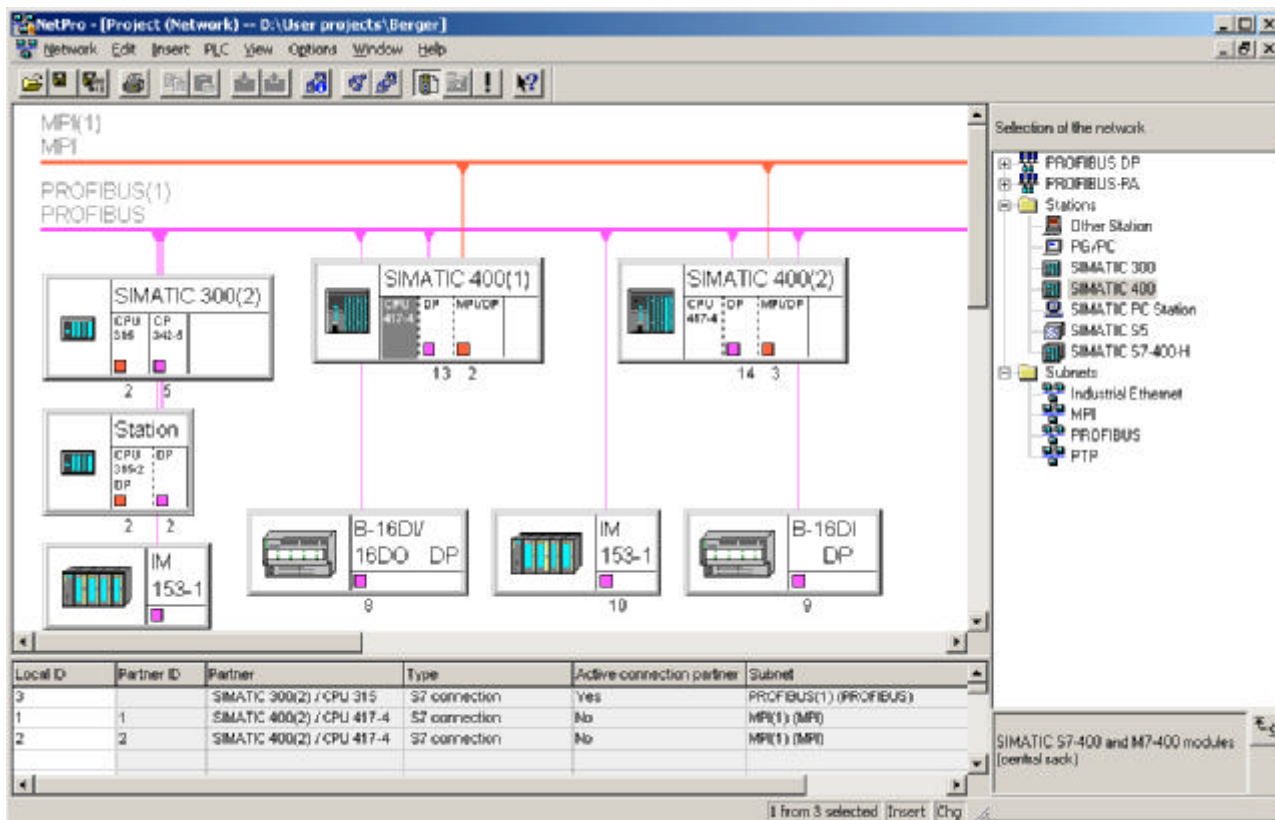


Рисунок 2.4 – Приклад Network Configuration

Робота в конфігурації мережі починається з вибору підмережі. За допомогою миші перетягніть у вікно мережі, щоб вибрати підмережу з каталогу. Підмережа вікна зображується горизонтальною лінією. Недійсна позиція вказується забороненим покажчиком миші [3].

Без підключення до мережі аналогічні операції спочатку виконуються на необхідних станціях. Станції залишаються «порожніми». Подвійне клацання станції відкриє інструмент налаштування мережі, який дозволяє налаштувати станцію для під'єднаних до мережі модулів.

2.2.3 Створення програми S7

У папці програм S7 була створена спеціальна програма. Цю папку можна включити в ієрархію проекту ЦП або створити незалежно від ЦП. У папці S7 програма містить об'єкти Symbols, Source Files і Blocks (рис. 2.5).

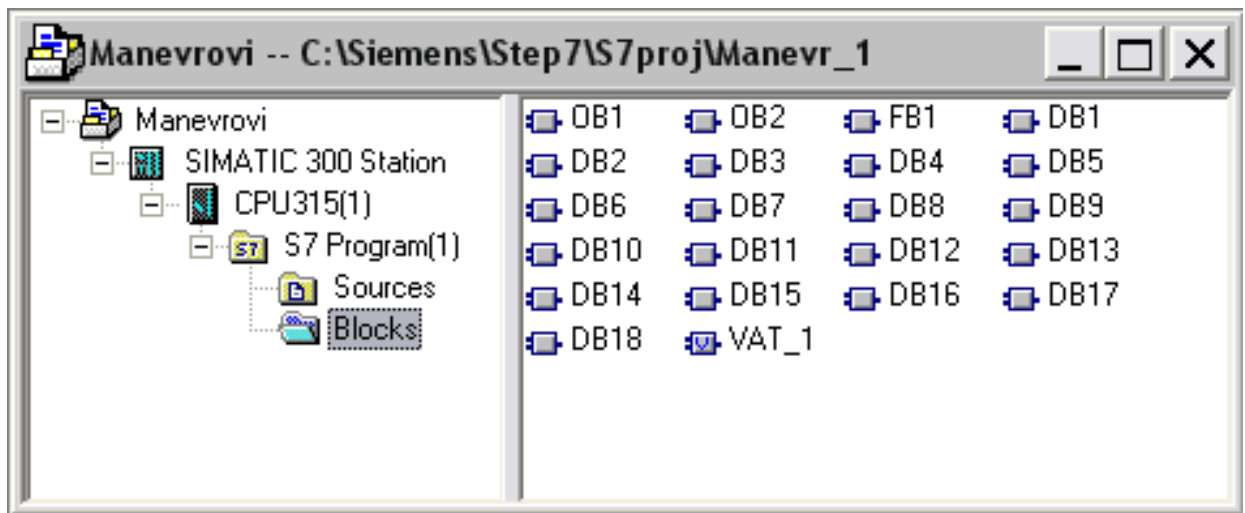


Рисунок 2.5 – Об’єкти для програмування

При створенні програми вводьте її поетапно блоками (блоками). негайно перевірте синтаксис введеної інформації. Блок записується і зберігається в контейнері Block і компілюється одночасно. За допомогою покрокового програмування ви можете редагувати блоки ЦП в режимі онлайн під час виконання завдань.

При створенні програми основна робота виконується за адресою; Це входи, виходи, таймери та блоки. Вони можуть давати як абсолютні, так і символічні адреси. Замість абсолютної адреси імені використовується маркування символів. Використання символічних адрес робить програму більш читабельною [4].

Символьні адреси включають локальні та глобальні символи. Локальні символи доступні лише у визначених блоках. Глобальний символ присутній у всій програмі і має однакове значення для всіх блоків. Глобальні символи визначаються в таблиці символів (об’єкт Symbols в контейнері програми S7).

Глобальні символи починаються з літери і можуть містити до 24 символів.

У таблиці символів можна назвати такі адреси та об’єкти.

- вхід, вихід, периферійний вхід і периферійний вихід;
- маркери, таймери, лічильники;

- кодові блоки OB, FB, FC, SFC, SFB і блоки даних DB;
- таблиці змінних (variable table) VAT.

Адреси даних і блоки даних входять до числа локальних адрес; пов'язані символи визначаються в розділі опису блоку даних для глобальних блоків даних і в розділі опису функціональних блоків, наприклад, блоків даних.

При створенні програми S7 менеджер SIMATIC також створює порожню таблицю символів Symbols. Він визначає персонажів світу та дає їм абсолютну адресу (рис. 2.6). S7 може мати лише одну таблицю символів.

	Symbol	Address	Data type	Comment
1	Cycle Execution	OB 1	OB 1	Циклічний блок
2	Dat_1_Kol_3	I 1.2	BOOL	Датчик 1 для колії 3
3	Dat_1_Kol_8	I 3.2	BOOL	Датчик 1 для колії 8
4	Dat_2_Kol_3	I 1.3	BOOL	Датчик 2 для колії 3
5	Dat_2_Kol_8	I 3.3	BOOL	Датчик 2 для колії 8
6	Dat_3_Kol_3	I 1.4	BOOL	Датчик 3 для колії 3
7	Dat_3_Kol_8	I 3.4	BOOL	Датчик 3 для колії 8
8	Dat_4_Kol_3	I 1.5	BOOL	Датчик 4 для колії 3

Рисунок 2.6 – Приклад таблиці символів

2.3 Тестування програми

Після підключення до ЦП і завантаження користувальницької програми є можливість протестувати її повністю або частково, наприклад, окремими блоками.

Якщо процесор переходить у стан STOP через помилку тестування, першим кроком у визначенні причини зупинки є запуск діагностичного буфера. ЦП поміщає всі повідомлення в буфер діагностики, включаючи причину зупинки та помилки, які до неї призвели. Щоб відобразити діагностичний буфер, перейдіть до онлайнного режиму з позначкою S7-application і виберіть

вкладку Diagnostic buffer за допомогою команди меню PLC -> Module information. Остання подія (це номер 1) призвела до переходу в стан STOP, наприклад, «STOP, тому що ОВ помилки програмування не завантажено» (STOP, тому що помилка програми не завантажена).

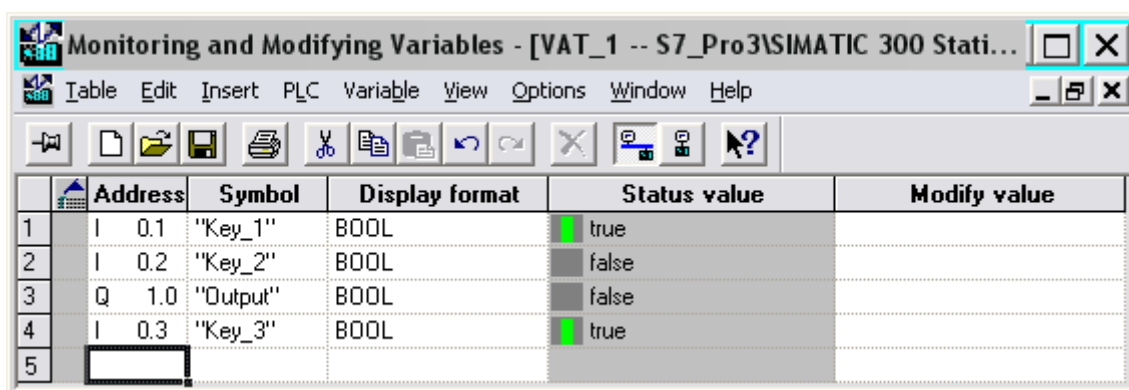
Помилка, яка викликала СТОП, була описана в попередньому повідомленні, наприклад «FC не завантажено» [4].

2.4 Відладка створеної програми

Важливим компонентом налагодження програми користувача є спостереження та зміна змінних за допомогою таблиці змінних VAT. Тут можна відобразити стан сигналу або значення простої змінної типу даних.

Щоб спостерігати за змінними, створіть таблицю змінних VAT, яка містить змінні та відповідний формат даних. Ви можете створити до 255 таблиць змінних (від VAT 1 до VAT 255) і назвати таблицю символів. Максимальний розмір таблиці змінних становить 1024 рядки довжиною до 255 символів (рисунок 2.7).

Таблицю змінних VAT можна створити окремо, вибравши Blocks спеціальну програму та команду меню Insert -> S7 Block ->E Variable Table, або вибравши команду меню програми S7, PLC ->Monitor/Modify Variables. .



	Address	Symbol	Display format	Status value	Modify value
1	I 0.1	"Key_1"	BOOL	true	
2	I 0.2	"Key_2"	BOOL	false	
3	Q 1.0	"Output"	BOOL	false	
4	I 0.3	"Key_3"	BOOL	true	
5					

Рисунок 2.7 – Приклад таблиці змінних

2.5 Створення діаграми входів і виходів для двигуна

Для управління транспортною платформою у виробничому процесі використовується один тип двигуна. Кожен двигун управляється власним «блоком двигуна», однаковим для всіх пристроїв. Цей пристрій потребує чотирьох входів: запуск, зупинка, помилка та скидання сервісного екрана. Логічний блок також потребує чотирьох виходів: два для вказівки стану роботи двигуна, один для вказівки на помилку і один для вказівки на те, що двигун потрібно обслуговувати.

Діаграма входів і виходів двигуна зображена на рисунку 2.8.



Рисунок 2.8 – Діаграма входів/виходів для двигуна

Кожен процес вимагає інтерфейсу для зв'язку з оператором, що забезпечує участь людини в процесі. Для включення, зупинки та моніторингу процесу необхідна станція оператора. Станція оператора оснащена наступним обладнанням.

- перемикач управління фазою процесу. Використовуйте перемикач, щоб вимкнути електродвигун для технічного обслуговування.
- лампи індикатора пристрою для індикації стану процесу.
- вимикач аварійної зупинки.

Ієрархія виклику блоку, завантажена в програму малювання на рисунку 2.9

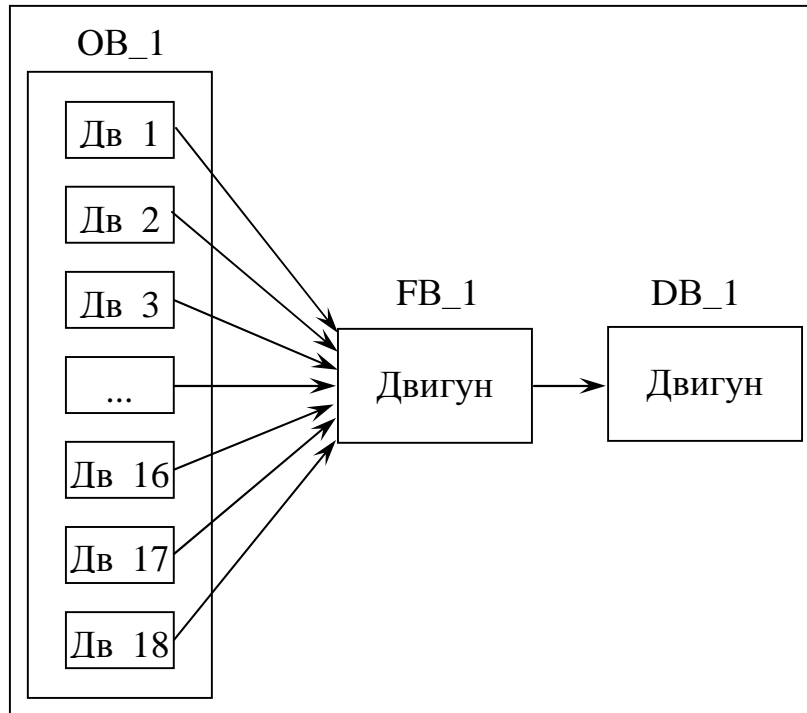


Рисунок 2.9 – Ієрархія викликів блоків програми

Блок OB1 підключається до операційної системи ЦП і містить основну програму. У OB1 блоки FB1 завантажуються і передають спеціальні параметри, необхідні для управління процесом.

Усі двигуни можна керувати за допомогою функціонального блоку FB1, оскільки вимоги (увімкнення, вимкнення тощо) однакові.

Приклади DB 1 включають фактичні параметри керування двигуном та статичні дані.

Символи використовуються в звичайних програмах і повинні бути визначені в таблиці символів за допомогою кроку 7.

2.6 Створення FB електродвигуна

Двигун FB виконує такі логічні функції:

- є входи запуску та зупинки;

- кілька замків дозволяють пристрою працювати. Стан блокування зберігається в тимчасових локальних даних (L-стек) ОВ1 («Активація двигуна», «Активація клапана») і логічно поєднується з введенням пуску та зупинки під час обробки FB для двигуна;

- сигнали зворотного зв'язку від пристроїв повинні з'явитися протягом зазначеного часу. В іншому випадку помилка або помилка вважається такою, що сталася. Ця функція зупиняє двигун;

- повинен задаватися час відповіді сигналу, тривалість або період помилки необхідно вказати;

- якщо натиснути кнопку пуску та розблокувати двигун, пристрій буде продовжувати працювати до тих пір, поки не будуть натиснуті кнопки запалювання та зупинки.

При використанні багатьох типів двигунів FB необхідно вказати назви загальних вхідних і вихідних параметрів. У звичайному процесі двигун FB вимагає наступного:

- він повинен отримати сигнал від операторської станції про зупинку та запуск двигуна;

- для цього потрібен сигнал відповіді від двигуна, що означає, що двигун працює.

Ці вимоги можна визначити як вхід і вихід FB. У таблиці 2.2 показані параметри FB двигуна в нашому звичайному процесі. Програмний код FAB1 LAB наведено в Додатку А.

Таблиця 2.2 – Параметри для FB електродвигуна

Адреса	Опис	Ім'я	Тип	Початкове значення	Коментарі
0.0	In	Start	BOOL	FALSE	Включення двигуна
0.1	In	Stop	BOOL	FALSE	Зупинка двигуна
0.2	In	Maint	BOOL	FALSE	Обслуговування двигуна
0.3	In	Fault	BOOL	FALSE	Несправність
2.0	Out	Start_Dv	BOOL	FALSE	Стан включення двигуна
2.1	Out	Stop_Dv	BOOL	FALSE	Стан виключення двигуна
2.2	Out	Maint_Dv	BOOL	FALSE	Сповідання про обслуговування двигуна
2.3	Out	Fault_Dv	BOOL	FALSE	Виявлення помилки двигуна

Загальний вигляд створеного блоку зображений на рисунку 2.10

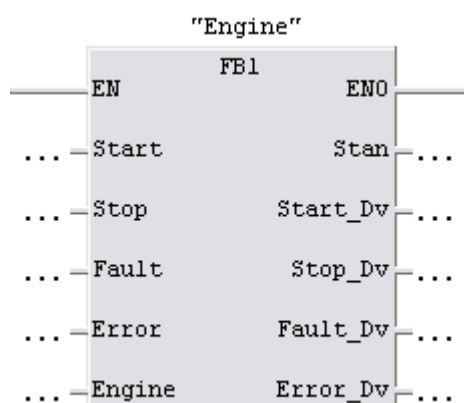


Рисунок 2.10 – Загальний вигляд функціонального блоку двигуна

Розділ коду програми управління транспортними платформами представлений в додатку Б.

3 СПЕЦІАЛЬНА ЧАСТИНА

3.1 Конфігурація і параметризація модулів

Інструмент налаштування апаратного забезпечення використовується для налаштування контролера програмування. Налаштування апаратного забезпечення починається з вибору станції, перейдіть до меню «Редагувати» Відкрити об'єкти» або двічі клацніть об'єкт «Устаткування» у відкритому контейнері станції SIMATIC 300 (станція SIMATIC 300). Виконайте основні налаштування за допомогою Параметри > Налаштувати.

Для даної роботи була створена конфігурація, яка показана на рисунку 3.1.

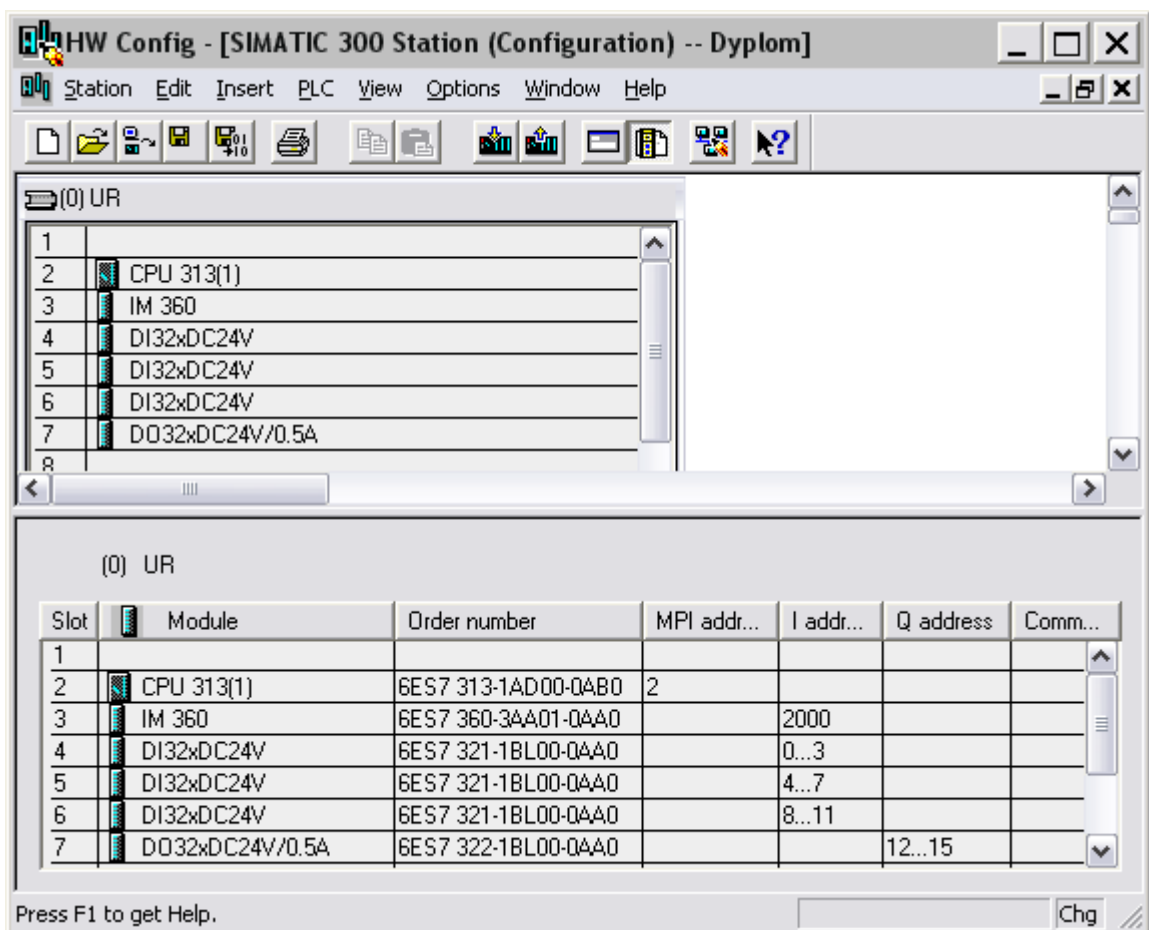


Рисунок 3.1 – Таблиця конфігурації станції

CPU 313 (рисунок 3.2) використовується як стаціонарний CPU - компактний CPU з 10 дискретними входами і 6 окремими виходами, який використовує мікрокарту пам'яті. Вбудовані функції обчислення швидкості. Наявність встановлених входів і виходів дозволяє безпосереднє підключення до об'єкта управління [6].

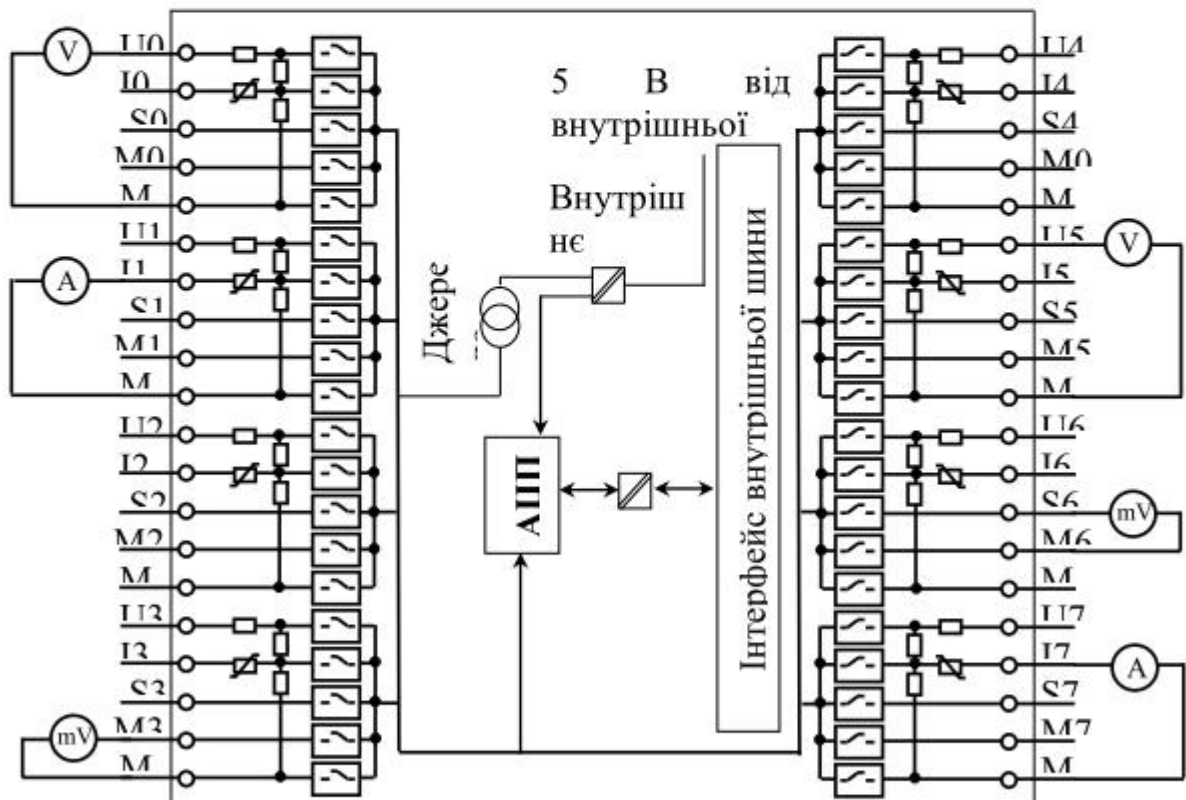


Рисунок 3.2 – Загальний вигляд процесора

Для аналогових модулів входу використовується модуль SM 331. Модулі введення аналогового сигналу використовуються для перетворення аналогового вхідного сигналу контролера в аналого-цифровий і для генерування числового значення, яке використовується центральним процесором під час виконання програми. До входу модуля можна підключати датчики, термопари та термометри опору з інтегрованими вихідними сигналами напруги або струму [6]. Загальний вигляд модуля та схема його підключення показано на рисунку 3.3.



а)



б)

а) Загальний вигляд модуля

б) Схема підключення модуля

Рисунок 3.3 – Загальний вигляд та схема аналогового модуля входу

Як модуль дискретного входу використовується модуль SM 321. Модулі введення дискретного сигналу використовуються для перетворення окремого

вхідного сигналу контролера у внутрішній логічний сигнал. До входу модуля можуть бути підключені контактні та безконтактні датчики [6].

Модулі можуть бути розміщені в пластиковій коробці (рисунок 3.4 а).

На передній панелі:

- зелений світлодіод, що вказує на стан вхідного ланцюга;
- червоний світлодіод, що вказує на помилку;
- передній роз'єм закритий роз'ємом і захисною кришкою;
- пази в захисному кожусі для встановлення етикеток із зовнішнім маркуванням ланцюга.

маркуванням ланцюга.

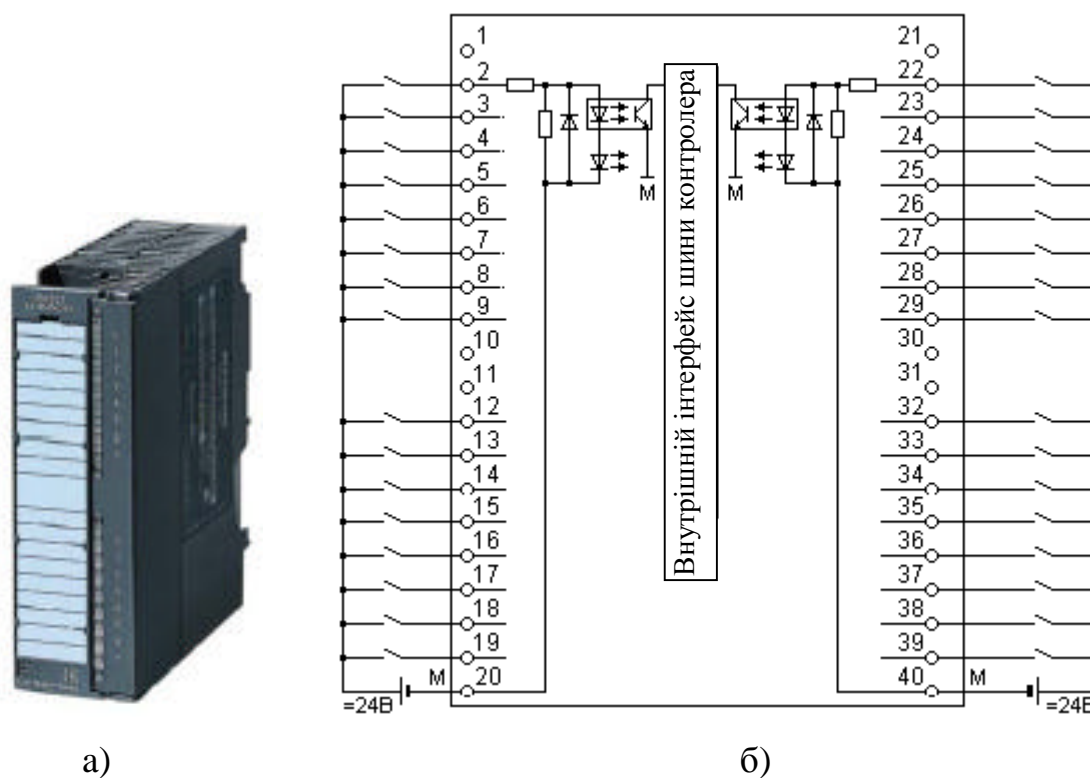


Рисунок 3.4 – Модуль SM 321

Модуль SM 332 використовується як модуль аналогового виходу. Модулі виводу аналогового сигналу призначені для перетворення внутрішнього

цифрового значення контролера в цифрово-аналоговий і формування аналогового вихідного сигналу. До виходу модуля можна підключити активатор, керований одним сигналом напруги або струму.

Щільність модулів становить 12 біт. Вони можуть створювати запити на переривання для надсилання діагностичних повідомлень. За потреби з модуля можна отримати додаткову діагностичну інформацію.

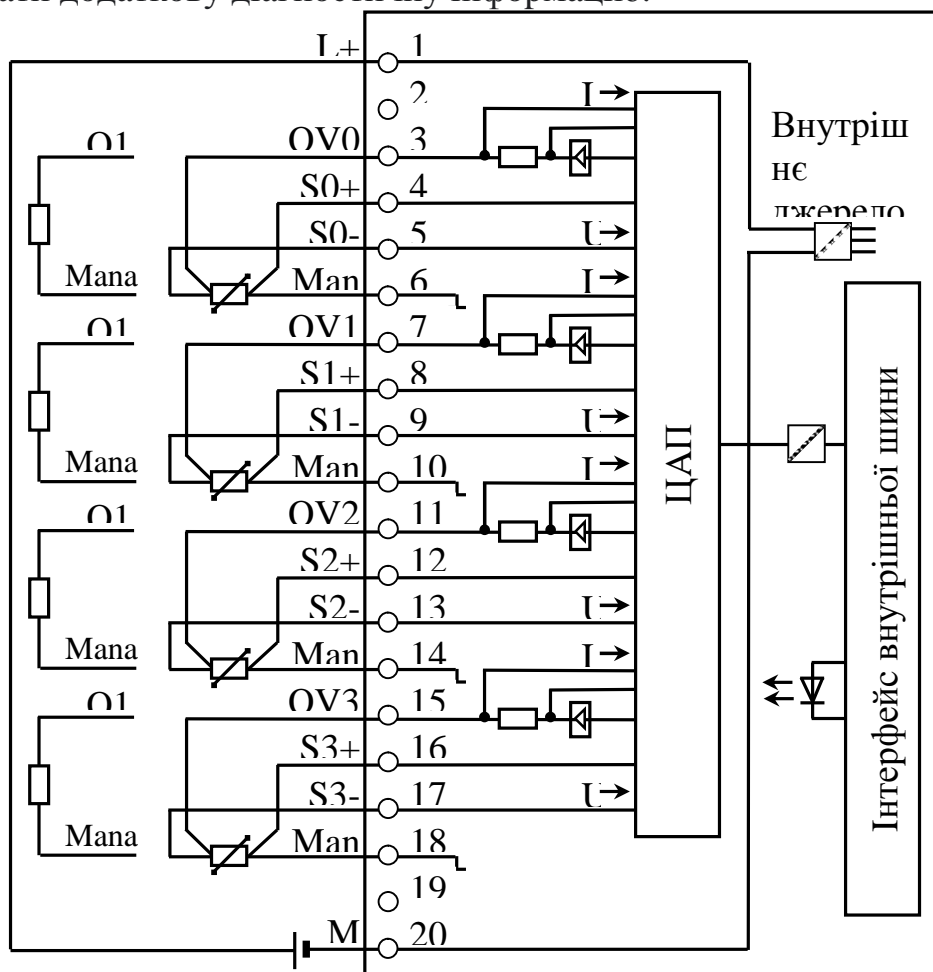


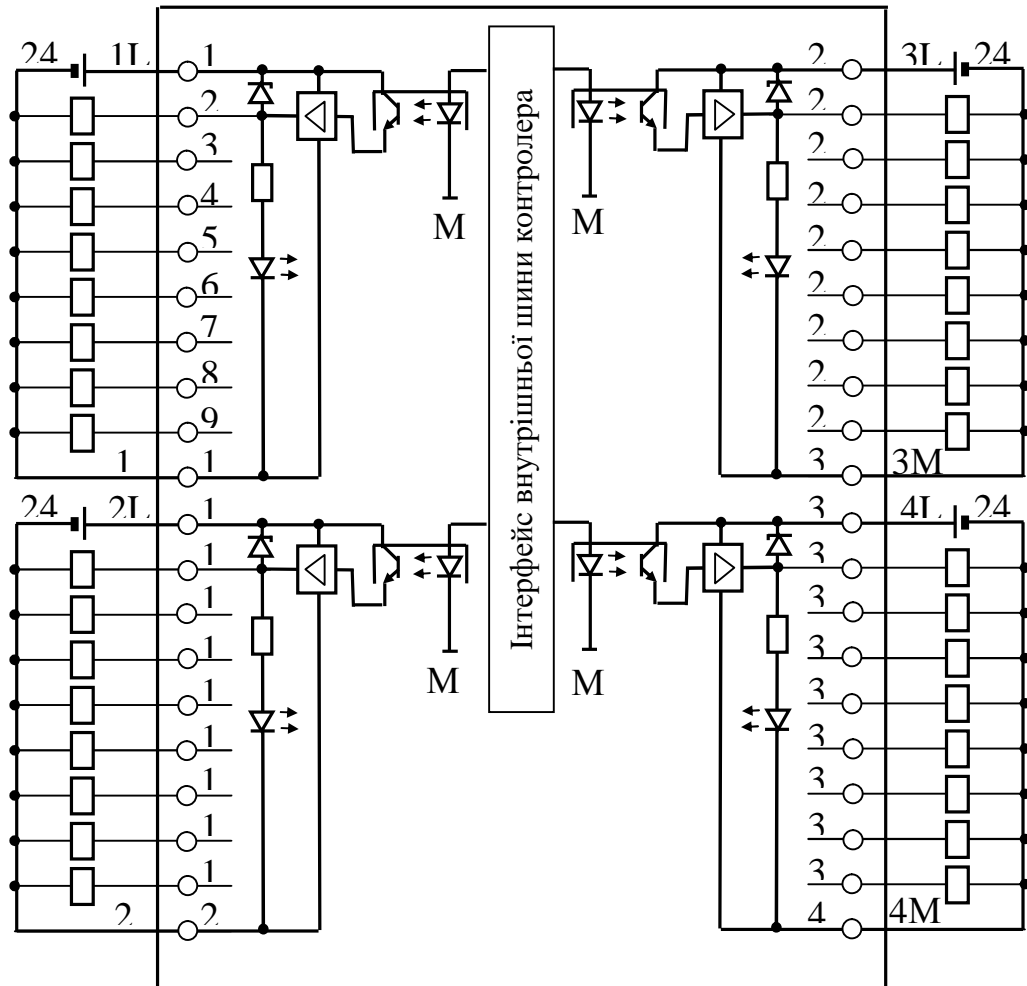
Рисунок 3.5 – Схема підключення модуля SM332

Модулі виведення дискретного сигналу призначені для перетворення внутрішнього логічного сигналу контролера в окремий вихідний сигнал. Вихід модулів може бути підключений до приводу або їх комутаційного пристрою.

Як модуль виведення дискретного сигналу використовується модуль SM 322. Модулі виводу дискретного сигналу використовуються для перетворення внутрішнього логічного сигналу контролера в його вихідний дискретний

сигнал. Клієнт або пристрої та їх комутаційні пристрої можуть бути підключені до виходу модулів.

Схема підключення модуля зображено на рисунку 3.6.



Рисунку 3.6 – Схема підключення модуля

Останнім елементом цієї системи є блок живлення. Вибрано джерело живлення PS 307 (10A). Блок живлення PS 307 призначений для створення вихідної напруги = 24 В, необхідної для живлення кількох модулів ЦП і контролера SIMATIC S7-300.

Блок живлення PS 307 використовує вхідну напругу ~ 120 / 230 В. Цей блок живлення можна використовувати для живлення внутрішньої схеми контролера та для живлення вхідних і вихідних ланцюгів [7].

Модуль монтується з лівого боку на стандартну профільну рейку DIN S7-300. Праворуч розташований модуль ЦП або інтерфейсний модуль ІМ 360 (на стійці розширення). Підключення до CPU або інтерфейсного модуля ІМ 360 здійснюється за допомогою електричного роз'єму, який постачається з кожним блоком живлення.

3.2 Характеристика компонентів мови LAD

3.2.1 Послідовні та паралельні схеми

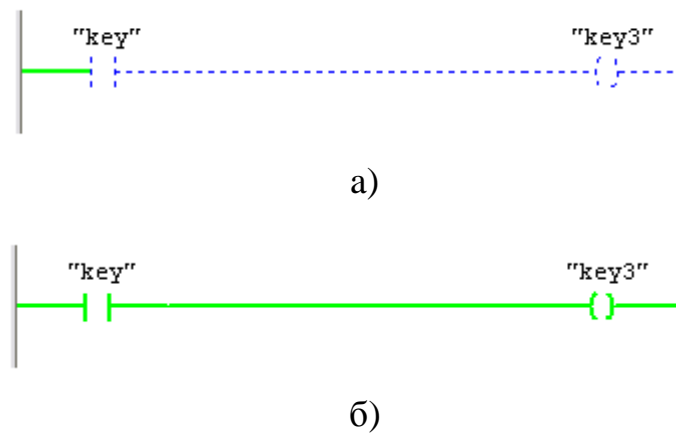
Стани двійкового сигналу групуються в LAD за послідовним і паралельним контактним з'єднанням.

Послідовне з'єднання відповідає функції І, а паралельне – функції АБО. LAD використовує тільки NO контакти ("1" для статусу сканованого сигналу) і NC контакти ("0" для статусу сканованого сигналу). Якщо відсканований двійковий операнд має стан сигналу «1», контакт NO активується, замикається, і «тече струм».

На рисунку 3.7 показаний приклад, коли датчик підключений до входу і сканується за допомогою NO контакту. Якщо датчик відкритий, вхідний «ключ» містить «0» і струм через розривний контакт не проходить (рисунок 3.7 а). Контактори, керовані виходом «ключ 3», не входять у комплект.

Якщо датчик активований, вхідний «ключ» знаходиться в стані сигналу «1». Струм з лівої живильної шини тече через розмикаючий контакт до котушки, а контактор підключається до виходу «ключ3» і активується (рисунок

3.7 б). NO-контакт сканує вхід в очікуванні стану сигналу «1», а потім NO або NC замикається незалежно від того, який контакт, вхід є датчиком.



а) Вхід „key” не активований (поданий „0”)

б) Вхід „key” активований (подана „1”)

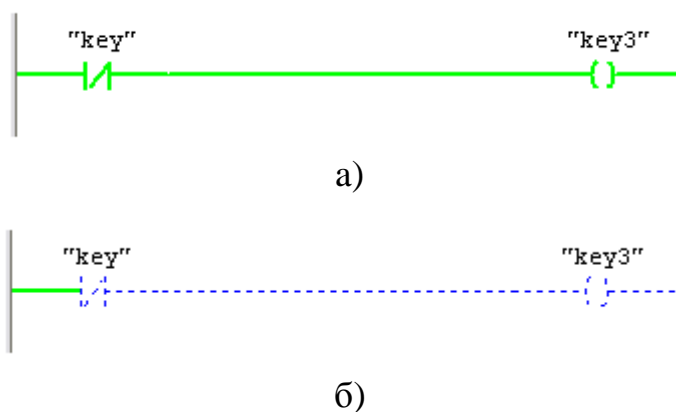
Рисунок 3.7 – Приклад застосування NO контакту

Якщо двійковий операнд має стан сигналу «0», струм протікає через NC-контакт. Якщо стан сигналу «1», мережевий контакт «розмикається» і потік струму переривається. У прикладі на малюнку 3.8, якщо датчик не закритий, струм протікає через NC-контакт (вхід «ключ» знаходиться в стані сигналу «0»). Через котушку і вихід контактора «ключ3» протікає струм – увімкнено (рисунок 3.8 а). Якщо датчик активований, вхід «ключ» буде в стані сигналу «1» і розмикається контакт NC.

Струм переривається і вихідний контактор «ключ3» відключається від навантаження (рисунок 3.8 б). Мережевий контакт перевіряється, щоб переконатися, що вхід знаходиться в стані нульового сигналу, а вхід знаходиться в закритому стані, незалежно від того, який контактний датчик, НЗ чи НЗ контакт.

3.2.2 Лічильник

Лічильники дозволяють використовувати ЦП для обчислювальних задач. Лічильник може реєструвати збільшення (прямий підрахунок) і зменшення (підрахунок). Поле рахунку містить три цифри (від 000 до 999). Лічильники розташовані в системній пам'яті ЦП; Кількість лічильників визначається версією ЦП.



а) Вхід „key” не активований (поданий „0”)

б) Вхід „key” активований (подана „1”)

Рисунок 3.8 – Приклад застосування НС контакту

Лічильник буде просканований, зчитуючи його статус (нульове або відмінне від нуля значення рахунку) або поточне значення лічильника.

За допомогою лічильника можна виконати наступні операції:

- встановлення лічильників та коригування значень лічильників;
- прямий рахунок;
- повернення рахунку;
- дахин скидання лічильника;
- прочитати стан (бінарного) лічильника;
- інструмент двійкового читання (числового) значення;
- прочитати значення лічильника в двійковому (десятковому) вигляді.

Елемент блоку лічильника містить усі функції обліку у вигляді функціональних входів і функціональних виходів (рисунок 3.9). Абсолютна або символічна адреса лічильника повинна бути на елементі блоку. Елемент блоку визначає тип лічильника як розділ заголовка (S CUD означає "лічильник вгору і вниз", "лічильник вперед і лічильник"). Перша точка входу повинна бути; для інших входів і виходів ви можете залишити пункт призначення.

Є три виконання блокових елементів лічильника: лічильник прямого і зворотного рахунку (up-down counter, S CUD), лічильник тільки прямого рахунку (up counter, S CU) і лічильник зворотного рахунку (down counter, S CD).

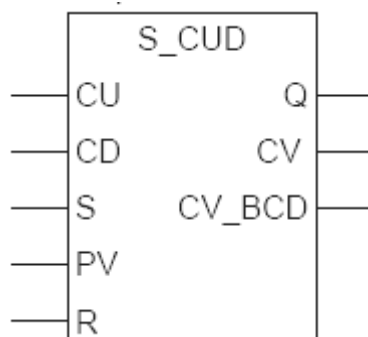


Рисунок 3.9 – Представлення лічильника на мові LAD

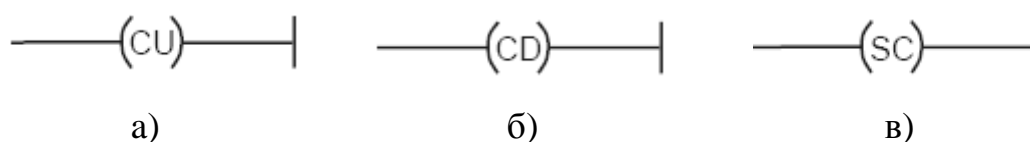
Визначення входів і виходів лічильників наведено в таблиці 3.1.

Також можна запрограмувати лічильник за допомогою окремих елементів (рисунок 3.10). У цьому випадку монтаж і розрахунок виконуються за допомогою котушки. Котушка встановлення лічильника містить операцію розрахунку (SC = Калібрувати лічильник); Під котушкою наведено значення лічильника у форматі WORD, яке використовується для встановлення лічильника. Щоб скинути лічильник, перевірте стан лічильника за допомогою котушки скидання та NO або NC контакту.

Таблиця 3.1 – Входи і виходи лічильника

Найменування	Тип даних	Опис
CU	BOOL	Вхід прямого підрахунку
CD	BOOL	Вхід зворотного підрахунку
S	BOOL	Вхід установки
PV	WORD	Вхід перед установки
R	BOOL	Вхід скидання
CV	WORD	Текуче значення в двійковому вигляді
CV_BCD	WORD	Текуче значення в двійково-десятковому вигляді
Q	BOOL	Стан лічильника

Налаштуйте лічильник, коли RLO змінюється з "0" на "1" на вході обмотки блоку або блоку S перед блоком. Для встановлення лічильника завжди потрібен плюсовий фронт. «Калібрувати лічильник» означає, що лічильник було скинуто. Діапазон значень - від 0 до 999.



- а) Прямий підрахунок котушок (метрів)
- б) Підрахунок котушки (метра).
- в) Встановіть лічильник і визначте значення лічильника

Рисунок 3.10 - Відображення лічильника як самостійного елемента

Лічильник скидається, коли струм протікає на вході скидання або котушці скидання (коли RLO дорівнює «1»). У цьому випадку показання

лічильника з НЗ контактом буде показувати «0», а перевірка NC контакту – «1». Скидання лічильника встановлює його облікове значення на «нуль». Вхід R блоку лічильника може бути вільним.

У цьому розділі описано основні компоненти конфігураційної таблиці, що використовується для побудови проектної станції S7, та компоненти мови LAD, що розробляла програмне забезпечення для керування транспортною платформою.

3.3 Опис процесу управління транспортною платформою

3.3.1 Характеристики програмного продукту SIMATIC WinCC

Система SCADA (моніторинг і збір даних) SIMATIC WinCC (центр управління Windows) — це комп'ютерна система з людино-машинним інтерфейсом, яка працює під керуванням операційних систем Windows і забезпечує широкий спектр систем управління будівлею.

Система підходить для роботи з усіма типами продуктів SIMATIC, оснащена потужним інтерфейсом для зв'язку з процесами, забезпечує парольний доступ до управління процесами, має високу продуктивність. Базова конфігурація системи дуже гнучка і може використовуватися для створення системи управління різного призначення.

Зв'язок із сторонніми системами управління через PROFIBUS здійснюється через інтерфейс FMS. Наступне програмне забезпечення використовується для організації обміну даними через PROFIBUS на SIMATIC WinCC V5.0 SP2.

- ДП-5613;

- ФМС-5613.

Функція зв'язку S7 підтримується програмним забезпеченням S7-5613 / Windows NT4.0 і Windows 2000 Pro.

SIMATIC WinCC Mapping Systems [Windows Control Center - Windows Control Center] — це система управління та контролю, яка використовує персональний комп'ютер як основу для системи, яка надійно відстежує певні процеси в стандартному середовищі Windows NT.

У продуктах SIMATIC HMI (HMI = Human Machine Interface) WinCC поділяється на системи управління та керування класом SCADA, вартість і рівні продуктивності з потужними функціями автоматизованого керування процесами. WinCC унікальний тим, що він надзвичайно відкритий.

Він легко взаємодіє зі стандартними та користувацькими додатками, створюючи досить прості візуальні рішення, які точно відповідають практичним вимогам. Завдяки відкритим інтерфейсам системні інтегратори можуть розробляти власні програми, навмисно створюючи системні розширення на основі WinCC.

WinCC — це система обробки зображень на базі Windows NT, заснована на ноу-хау Siemens, лідера продуктів для автоматизації процесів, і Microsoft, провідного світового розробника комп'ютерного програмного забезпечення. WinCC — це найсучасніша система моніторингу та керування для простих і складних завдань, яку можна легко інтегрувати в широкий спектр корпоративних додатків, від інтеграції до рішень MES та ERP.

Основними перевагами системи відображення WinCC є:

- використання інноваційних програмних технологій: WinCC базується на останніх розробках і методах в індустрії програмного забезпечення, і в тісній співпраці з Microsoft користувачі можуть впевнено сподіватися на інновації в майбутньому;

- вбудовані функції потужної системи обробки зображень: базова система WinCC є звичайною частиною WinCC, з усіма компонентами та функціями, необхідними для вирішення складних проблем із зображенням, редакторами зображень, сценаріїв, сигналізації, графіки та звітів;

- Розширення від простих завдань до складних: WinCC - це модульний модуль автоматизації з гнучким масштабуванням для простих застосувань у машинобудуванні.

- Масштабування за допомогою галузевих і технологічних опцій і доповнень: WinCC вже розробив ряд додаткових пакетів для задоволення конкретних потреб різних галузей і технологій, наприклад управління водними ресурсами на основі відкритого програмного інтерфейсу. База даних, яка зберігає дані. Доступ до бази даних WinCC можна отримати без проблем за допомогою бази даних SQL або за допомогою мови структурованих запитів через драйвер ODBC.

За допомогою цих методів доступу WinCC відкриває свої дані, наприклад, для інших програм і баз даних Windows і повністю сумісна з концепцією фабрики або підприємства;

- Високопродуктивні стандартні інтерфейси, такі як OLE, ActiveX та OPC: Стандартні інтерфейси, такі як DDE та OLE для обміну даними між додатками Windows, є важливою частиною WinCC, оскільки вони постійно інтегрують керування ActiveX та операції на стороні клієнта OPC сервер;

- Відкритий API з функцією WinCC і доступом до даних: надає всім модулям WinCC відкритий інтерфейс програмування. Це дозволяє встановлювати функції та функції проектування WinCC в програмі користувача;

- багатомовне програмне забезпечення для проектування: програмне забезпечення WinCC повністю розроблене для багатомовного використання;

- канали зв'язку контролера від усіх відомих виробників: охоплення WinCC включає всі важливі канали зв'язку, підключені до контролера SIMATIC S5 / S7 / 505, а також такі канали, як Profibus DP, DDE і OPC. Існує багато каналів зв'язку, доступних як варіанти або доповнення.

- Сумісність SIMATIC WinAC: поєднання ПЛК і систем обробки зображень на одному ПК є концепцією майбутнього. WinCC і WinAC

забезпечують потужні комп'ютерні, повністю інтегровані рішення автоматизації від Siemens;

- Повністю інтегрована автоматизація (TIA): TIA WinCC є комбінацією різних продуктів Siemens і є центральним підрозділом T.I.A., T.I.A. спільне проектування, програмування, зберігання даних та обмін інформацією;

Компоненти системи керування та керування SIMATIC PCS 7: SIMATIC PCS 7 — це система контролю й керування процесами в T.I.A. PCS 7 поєднує переваги рішень автоматизації на основі контролерів для автономного виробництва з перевагами рішень для контролю та управління процесами для безперервного виробництва.

- Інтеграція рішень MES та ERP: завдяки стандартному інтерфейсу SIMATIC WinCC є частиною всього ландшафту інформаційних технологій підприємства, від автоматизованих виробничих процесів до оптимізації процесів на рівні управління підприємством та підготовки адміністративних даних для управління підприємством (MES = Manufacturing) системи продуктивності та ERP = планування організаційних ресурсів).

3.3.2 Функціональні можливості системи WinCC

З допомогою надзвичайно потужних функцій проектування, які надає система SIMATIC WinCC (рис.3.11) можна радикально скоротити час на проектування. Спільно з іншими компонентами SIMATIC, система має в своєму складі такі додаткові функції, як діагностика і обслуговування процесу – новий напрям у візуалізації. При проектуванні таких функцій, між собою взаємодіють всі інструментальні засоби для проектування SIMATIC. WinCC надає всі основні функціональні можливості для візуалізації і управління процесом.



Рисунок 3.11 – Структура система візуалізації WinCC

З цією метою WinCC використовує ряд редакторів і інтерфейсів, за допомогою яких ці функції можуть бути індивідуально спроектовані для будь-якого додатку.

Графічне вікно системи візуалізації WinCC зображено на рисунку 3.12.

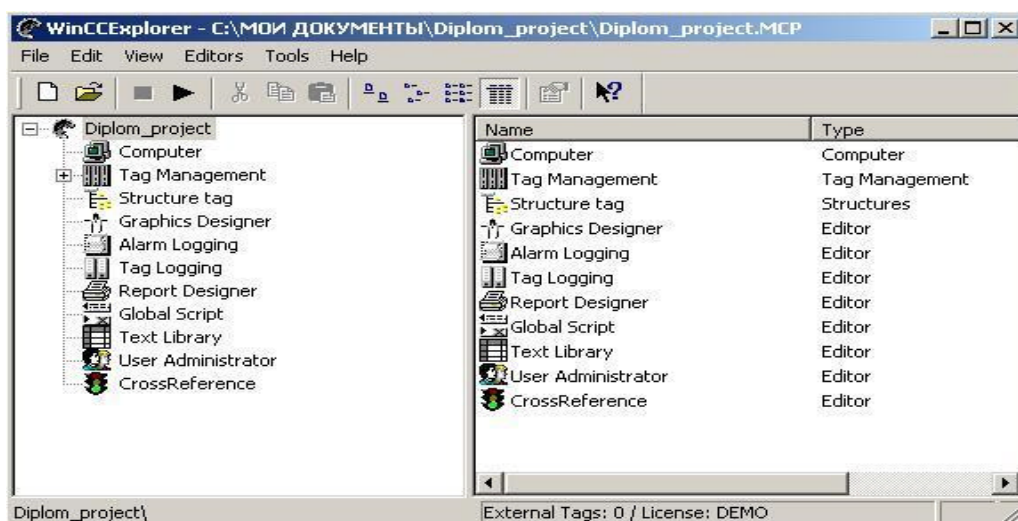


Рисунок 3.12 – Представлення графічного вікна системи візуалізації WinCC

Таблиця 3.2 – Основні редактори та інтерфейси WinCC

Редактори і інтерфейси WinCC	Задача або проектована функція етапу виконання
WinCC Explorer (Провідник WinCC)	Централізоване управління проектом для швидкого доступу до всіх його даних і централізована настройка
Graphics Designer (графічний дизайнер)	Графічна система для візуалізації і управління через графічні об'єкти, причому всі їхні властивості можуть бути зроблені динамічними
Alarm Logging (Реєстрація аварійних повідомлень)	Система повідомлень для реєстрації і архівації подій з можливостями відображення і управління відповідно до DIN19235; вільно вибрані класи повідомлень, відображення і протоколювання
Tag Logging (Реєстрація тегів)	Архівація значень змінних процесу для реєстрації, стиснення і збереження вимірних значень, наприклад, для відображення у вигляді графіків і таблиць та подальшої обробки
Report Designer (Дизайнер звітів)	Система звітів для керованого часом або подіями документування повідомлень, управляючих дій і поточних даних про процес у вигляді призначених для користувача звітів або проектною документації у вільно вибраному форматі
User Administrator (адміністратор користувача)	Інструмент для зручного управління користувачем і його повноваженнями
Global Scripts (глобальні сценарії)	Функції обробки з безмежними можливостями шляхом використання вбудованого компілятора ANSI-C
Канали зв'язку	Для обміну даними з пристроями управління (протоколи SIMATIC, Profibus DP, сервер DDE і OPC в об'ємі поставки)
Стандартні інтерфейси	Для відкритої інтеграції інших додатків Windows (ODBC/SQL, ActiveX, OLE, DDE, OPC і інше)
Інтерфейси для програмування	Для індивідуального доступу до даних і функцій WinCC (C-API) і для вбудовування в програми користувача

3.3.3 Принцип відкритості системи WinCC

SIMATIC WinCC (рис 3.13) являється одним із світових лідерів, що стосується використання найсучасніших програмних технологій в системах візуалізації. Тут йдеться про такі новітні технології Microsoft, як OX або

ActiveX, OLE і COM (DCOM), а також про вбудовування бази даних і розробку сценаріїв за допомогою стандартної мови програмування ANSI-C.

Разом з вбудовуванням новітніх технологій, вже програмна структура WinCC гарантує абсолютну відкритість. Це стає можливим завдяки модульній структурі WinCC і завчасній підготовці програмних інтерфейсів для окремих модулів.

Система WinCC надає можливості відкритості даних і системних функцій, завдяки чому не ставиться ніяких меж для нововведень при розв'язанні задач автоматизації.

Зокрема, система візуалізації WinCC містить :

- вбудовування технологій Microsoft;
- відкриті комунікації OPC;
- повна свобода програмування;
- використання стандартної мови сценаріїв ANSI-C;
- C-API для професійного використання;
- відкрита база даних;
- доступ до даних через SQL/ODBC.



Рисунок 3.13 – Принцип відкритості системи WinCC

3.3.4 Візуалізація процесу управління транспортними платформами маневрових пристроїв з використанням системи WinCC

Для здійснення візуалізації процесу управління транспортними платформами використовуючи можливості програмного середовища системи WinCC

створимо новий проект WinCC (рис 3.14), після чого в діалоговому вікні вибирається тип проекту, який створюється, відповідно Single-User (однокористувацький), Multi-User (багатокористувацький).

Після вибору типу проекту задається його назва у відповідному діалоговому вікні, а також назва і шлях до папки, де будуть розміщуватися основні файли створюваного проекту.

Після створення нового проекту з'являється графічне вікно WinCC з іменем даного проекту і вказуванням шляху локального розміщення фалів даного проекту.

Для виконання процесу візуалізації управління транспортними платформами використаємо графічний дизайнер Graphics Designer скориставшись відповідною командою пункту головного меню чи безпосередньо запустивши його із графічного вікна навігації.

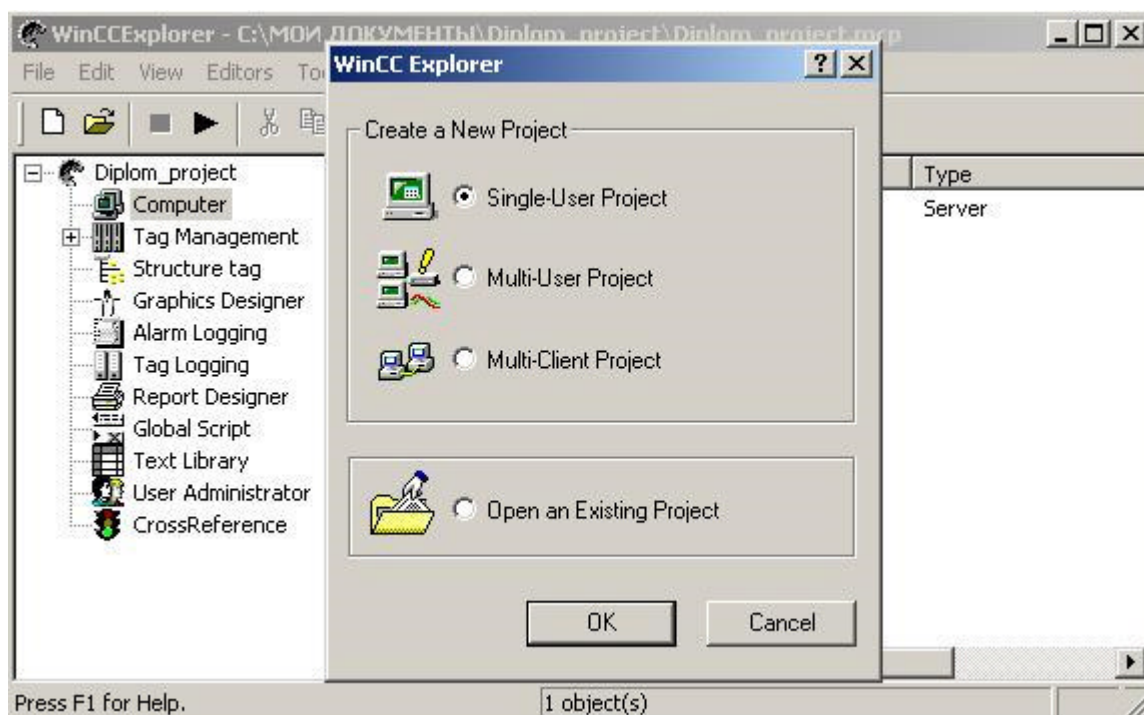


Рисунок 3.14 – Представлення діалогового вікна графічного вікна WinCC вибору типу створюваного проекту

Процес візуалізації можна виконувати послідовно вибираючи необхідні графічні компоненти із глобальної стандартної бібліотеки WinCC, де знаходяться вже готові і розсортовані по своєму типу і призначенню компоненти. Графічний дизайнер WinCC надає можливість також ручного створення необхідних компонентів з використанням панелей інструментів. Візуалізацію процесу управління транспортними платформами виконуємо з використанням графічного дизайнера на основі існуючих бібліотечних графічних об'єктів, а також поєднуючи при цьому відповідні засоби палітри об'єктів.

Процес завантаження графічного редактора WinCC показаний на рисунку 3.15. Після завантаження графічного дизайнера WinCC, можна здійснювати процес візуалізації з допомогою наявних засобів палітри об'єктів (рис. 3.16).

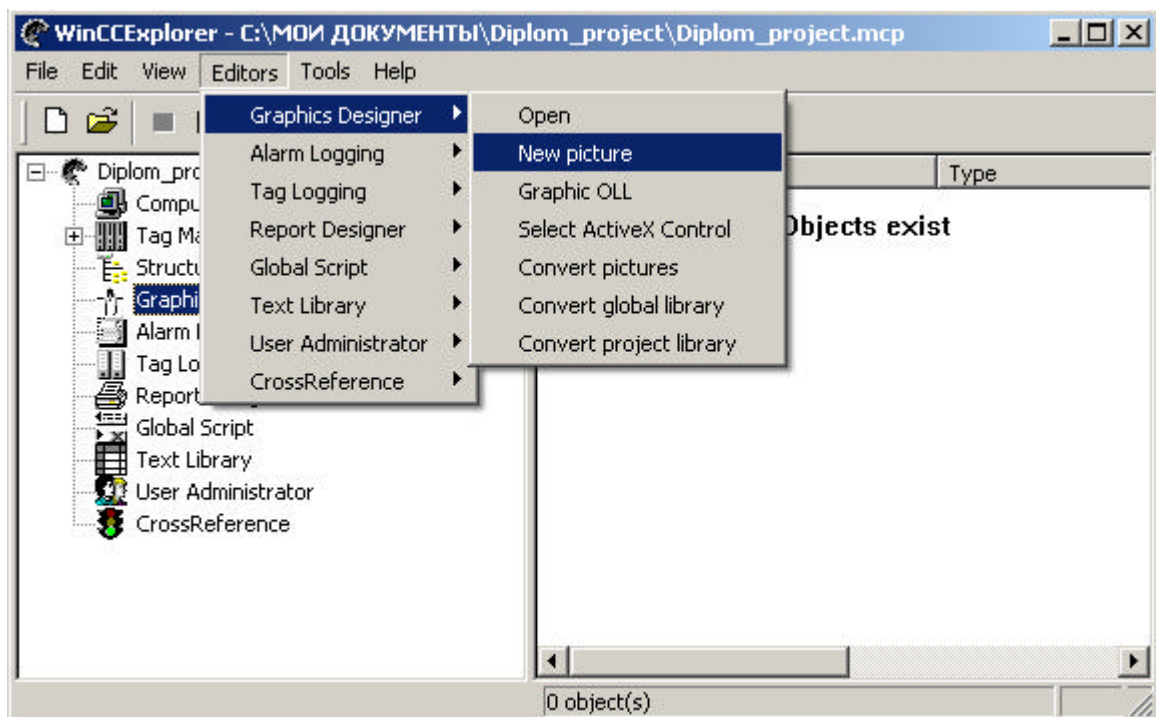


Рисунок 3.15 – Завантаження графічного дизайнера WinCC для виконання візуалізації процесу управління

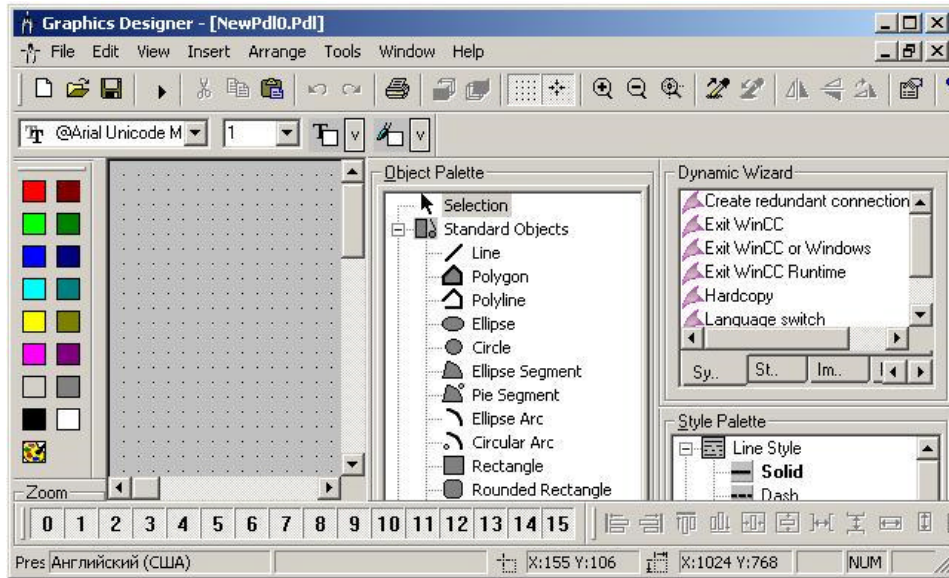


Рисунок 3.16 – Зображення вікна графічного дизайнера WinCC

Оскільки в графічному дизайнері системи візуалізації WinCC доступною являється бібліотека стандартних графічних об'єктів (компонентів), то це значною мірою прискорює процес виконання візуалізації (рис 3.17).

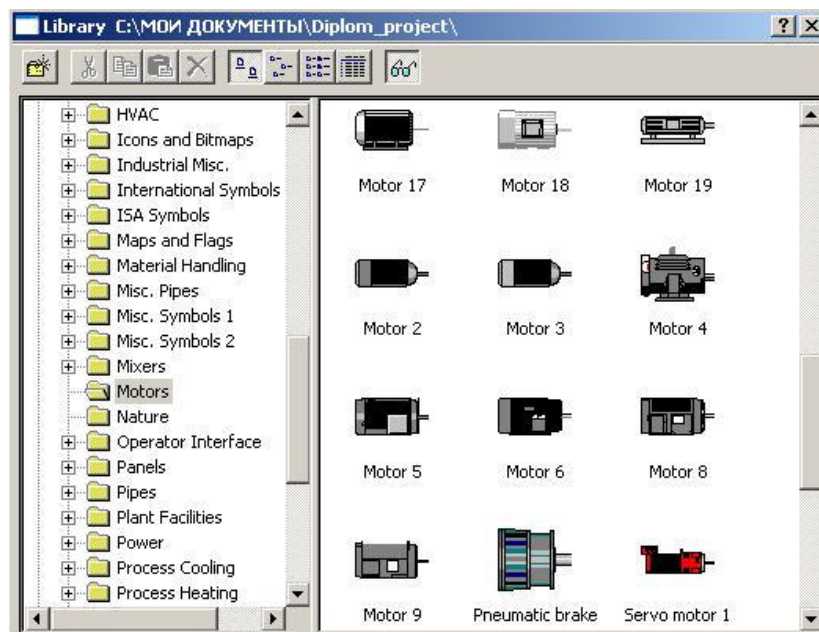


Рисунок 3.17 – Вибір стандартних бібліотечних компонентів WinCC для реалізації процесу візуалізації

Графічні об'єкти в основному призначені для створення системних кадрів. Вигляд об'єкта в режимі Run-Time може змінюватися, якщо об'єкт зробити динамічним, задавши його атрибути із змінними процесу. Graphics Designer передбачає також, використання так званих інтелектуальних об'єктів - Smart Objects, так наприклад об'єкт OLE дозволяє вбудовувати будь-які елементи управління ОСХ або ActiveX.

Динаміка в зображенні графічного об'єкта здійснюється шляхом безпосередньої прив'язки змінних процесу до властивостей об'єкта. Для оптимізації цієї частини проектування WinCC надає можливість перетягувати змінні з панелі символів методом буксирування (Drag and Drop) на властивості об'єктів і, таким чином, зв'язувати їх з властивостями.

Якщо об'єкт певним чином підключений до процесу, то можлива прив'язка безпосередньо до об'єкту, минувши вибір властивості, завдяки чому відбувається економія часу на проектування.

Візуалізований процес управління транспортними платформами маневрових пристроїв із позначенням основних структурних компонентів системи управління, зокрема давачів положення транспортної платформи (пічної вагонетки), привідних станцій на основі двигунів-редукторів зображений на рисунку 3.18.

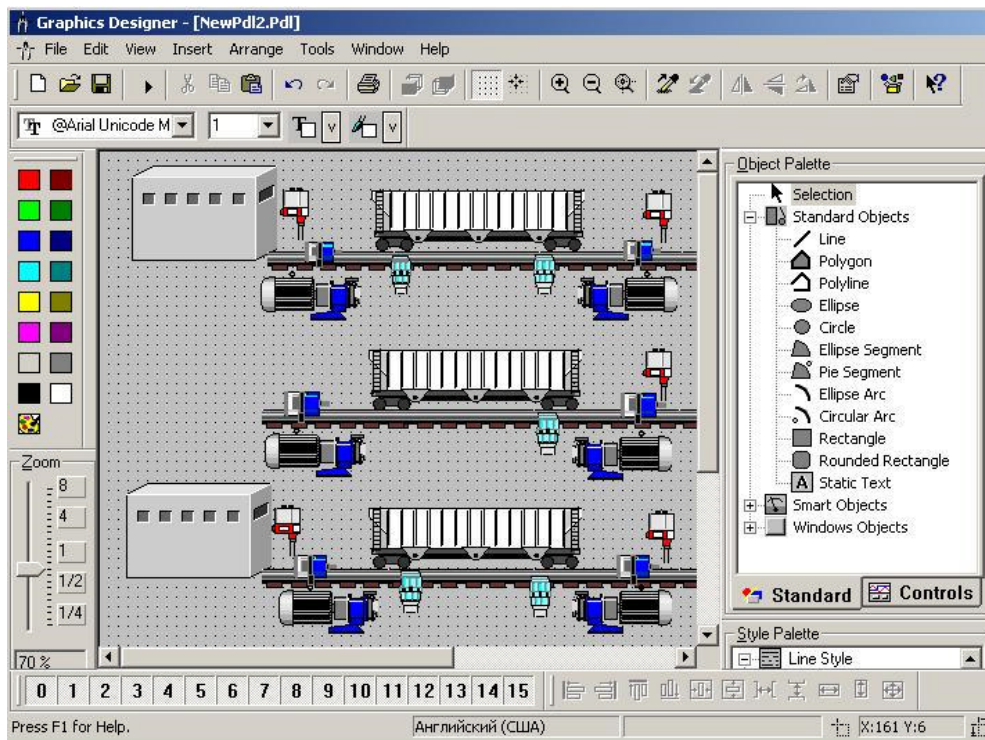


Рисунок 3.18 – Візулізація процесу функціонування транспортних платформ маневрових пристроїв з використанням засобів системи WinCC

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи було розроблено систему автоматизації технологічного процесу управління транспортними платформами за допомогою контролера S7-300 в Simatic Step7. Дана система дає можливість передавати дані на вищий рівень, тим самим організовуючи управління об'єкта на рівні підприємства.

Крім того використання в проекті автоматизації управління транспортними платформами контролера Simatic S7-300 дає змогу гнучко змінювати контролер як на програмному рівні так і на апаратному. Завжди є можливість його нарощувати, монтувати нові модулі, підключати його до інших аналогічних систем, крім цього тут можна використовувати панель оператора.

Дана система автоматизації характеризується високою надійністю, хорошими технічними показниками, відсутністю невизначених станів чи конфліктних ситуацій. Все це робить цю систему автоматизації економічно вигідною, через малу кількість споживаної потужності і хорошими технічними показниками.

ПЕРЕЛІК ПОСИЛАНЬ

1. Виницький М.Ю. Програмовані логічні контролери. К.–2003-135с.
2. Ганс Бергер. Автоматизация с помощью программ Step7 LAD и FBD. М.– 2004-678с.
3. LAD для S7-300 и S7-400, программирование. Справочное руководство. М. –2003-244с.
4. Ганс Бергер. Программирование с помощью Step7 V5.0. М.–2003-294с.
5. Конфигурация аппаратуры и проектирование соединений с помощью Step7 V5.0. Руководство. М.–2003-164с.
6. Simatic WinCC, версия 5. Оптимальная визуализация процессов. Каталог ST50.rus www.simatic.s7-300.ru
7. SIMATIC Комплексная автоматизация производства. Каталог ST70-2001.rus часть 1.
8. Шумилов Ю.А., Чебанюк В.К. – Асинхронные двигатели с улучшенными виброакустическими показателями, 1991.
9. Жидецький В.Ц., Джигирей В.С., Мельников О.В. Основи охорони праці. – Львів: Афіша, 2000.
10. Денисенко Г. Ф. Охрана труда. – М.: Химия, 1983.
11. Зимодро А.Ф., Скибинский Г.Л. Основы автоматики. – Л.: Энергоатомиздат, 1994. – 160 с.
12. Щербина Я. Я., Щербина И. Я. Основы противопожарной защиты. – М.: Вища школа, 1985.
13. Методичні вказівки до виконання кваліфікаційної роботи бакалавра спеціальності 151 «Автоматизація та комп'ютерноінтегровані технології»./ В.Б. Савків., Ю.Б. Капаціла, Р.І. Михайлишин// -: Тернопіль, Тернопільський національний технічний університет імені Івана Пулюя, 2021, – 46с.
14. https://knowledge.allbest.ru/programming/2c0a65635b3ac68a4d43a89421206d26_1.html
15. http://4ua.co.ua/manufacture/ra2ac69b4c53b89421316c27_0.html

16. <https://ukrbukva.net/page,19,52521-Avtomatizirovannoe-upravlenie-elektronnoiy-prohodnoiy.html>
17. <http://ukrefs.com.ua/page,16,91581-Razrabotka-sistemy-operativno-dispetcherskogo-kontrolya-i-upravleniya-kanala.html>
18. http://www.dgma.donetsk.ua/docs/kafedry/avp/metod/ПСА_Конспект_M3_УКР.pdf
19. https://plc4me.com/download-simatic_wincc_7-5-siemens-hmi-scada-software-real-100
20. <https://library.sumdu.edu.ua/uk/doslidnyku/prohramne-zabezpechennia/analiz-danykh-ta-vizualizatsiia/vizualizatsiia-ta-hrafichnyi-dyzain.html>
21. <https://victana.lviv.ua/knyhy/konspekty-lektsii/133-kros-platformente-prohramuvannia-ta-khmarni-servisy/570-lektsiia-22-rozodileni-modeli-promizhnoho-rivnia-dlia-windows-2016-r>
22. https://support.industry.siemens.com/cs/attachments/1144587/GraphicsDesigner_V502_r.pdf