

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

**МЕТОДИЧНІ ВКАЗІВКИ**  
**для проведення комп'ютерного практикуму з курсу**  
**«Розподілені системи управління»**

для студентів спеціальності  
151 – «Автоматизація та комп'ютерно-інтегровані технології»  
денної та заочної форм навчання

Затверджено  
редакційно-видавничою  
радою університету,  
протокол № від р.

Харків  
НТУ «ХПІ»  
2018

Методичні вказівки для проведення комп'ютерного практикуму з курсу **«Розподілені системи управління»** для студентів спеціальності 151 – «Автоматизація та комп'ютерно-інтегровані технології» денної та заочної форм навчання. Уклад. Подустов М.О., Лисаченко І.Г., Бабіченко А. К., Шутинський О.Г. – Х. : НТУ «ХП», 2018. – 84 с.

Укладачі: М.О. Подустов  
І. Г. Лисаченко  
А. К. Бабіченко  
О. Г. Шутинський

Рецензент: проф. І. Л. Красніков  
проф. каф. автоматизації технологічних систем  
та екологічного моніторингу

Кафедра автоматизації технологічних систем  
та екологічного моніторингу

## ВСТУП

Ці методичні вказівки розроблені для проведення комп'ютерного практикуму зі студентами денної та заочної форм навчання спеціальності 151 - «Автоматизація та комп'ютерно-інтегровані технології» з дисципліни «Розподілені системи управління».

Методичні вказівки складаються з двох розділів. В першому розділі викладені основні правила роботи в середовищі програмування контролерів на базі гібридного компактного програмно-технічного комплексу (ПТК) *Freelance 800F* та наведена методика створення людино-машинного інтерфейсу (ЛМІ). Перед проведенням практикуму необхідно самостійно ознайомитися з матеріалом першого розділу. Другий розділ присвячений власне виконанню завдань комп'ютерного практикуму.

Вказівки вміщують необхідні теоретичні відомості для розроблення прикладного програмного забезпечення (ППЗ) в середовищі *Control Builder F (CBF)* [1] для промислових контролерів *AC700* та *AC800* [2] виробництва підрозділу «Автоматизація процесів та виробництв» групи компанії *ABB Group* (Швейцарія/Швеція) [3]. Також в методичних вказівках розглянуті питання створення систем людино-машинної взаємодії, тобто ППЗ для автоматизованого робочого місця (АРМ) оператора технологічної установки, яке сумісно з промисловим контролером є елементами розподілених систем управління (PCY). При цьому, функцію АРМ оператора виконує звичайний ПК з встановленим спеціальним програмним забезпеченням, а саме з виконавчим модулем ЛМІ *DigiVis*.

Виконання комп'ютерного практикуму дозволяє студентам більш ефективно засвоїти в подальшому теорію та практику застосування PCY у складі комп'ютерно-інтегрованих систем управління. Це забезпечується комплексним підходом до виконання окремих завдань практикуму. Тобто, спочатку студенти виконують типові завдання з розроблення ППЗ, яке реалізує різноманітні алгоритми керування та функції управ-

лінія, що притаманні інформаційно-управляючим компонентам РСУ. Далі студенти самостійно вирішують індивідуальні завдання.

Отримання навичок розроблення ППЗ побудовано за принципом поступового ускладнення. В комп'ютерному практикумі розглянуті питання дискретного керування установкою, використання таймерів та лічильників, оброблення сигналів від аналогових датчиків, використання складних законів регулювання (П, ІІ та ІІД), тощо.

Для розроблення, завантаження і налагодження ППЗ для програваних логічних контролерів (ПЛК) серії *AC700* або *AC800* до складу ПТК входить ПК з встановленим спеціальним програмним забезпеченням (СПЗ) – середовищем *Control Builder F*. Необхідно відмітити, що вказане СПЗ повністю відповідає вимогам стандарту стосовно технологічного програмування *IEC 61131-3*. При цьому, для налагодження проєктів користувача ППЗ може використовуватися як реальний контролер, який входить до комплекту демо-кейса (в подальшому – кейс) *FREELANCE 700F* моделі *TYPE 7-DC-RB-AC700F*, так і емулятор ПЛК, який може бути запущений на ПК користувача (так звана інжинірингова станція). До складу кейса також входять симулятори вхідних та вихідних сигналів, які дозволяють імітувати роботу системи управління технологічним об'єктом (процесом). Для створення РСУ є додатковий кейс з *TYPE 8-DC-RB-PB-S700* модулем промислової мережі *PROFIBUS* та модулями розподіленої системи збирання інформації.

Опис основних принципів роботи з контролерами серії *AC700F* і *AC800F* виробництва *ABB Group* наведено в [4], а технічну підтримку можна отримати за адресою в *Internet* [5].

*Насамкінець зазначимо, що обладнання та програмне забезпечення для комп'ютерного практикуму безкоштовно надано компанією *ABB Group* через свій підрозділ в Україні «АББ Україна» (м. Київ) відповідно до програми підтримки вищих навчальних закладів.*

**РОЗДІЛ 1**  
**ЗАГАЛЬНІ ВІДОМОСТІ ПРО АПАРАТНЕ ТА ПРОГРАМНЕ**  
**ЗАБЕЗПЕЧЕННЯ ПРОГРАМНО-ТЕХНІЧНОГО КОМПЛЕКСУ**  
*FREELANCE 800F ВІД ABB Group*

**1.1 Загальні відомості про кейс *FREELANCE AC700F***

Загальний вигляд кейса *FREELANCE 700F* моделі *TYPE 7-DC-RB-PB-AC700F* представлений на рис. 1.1. На цьому рисунку цифрами позначено розташування основних елементів кейса: 1 – модулі контролера; 2 – імітатор вхідних сигналів та вихідних елементів; 3 – блок живлення; 4 – пенал з додатковим приладдям.

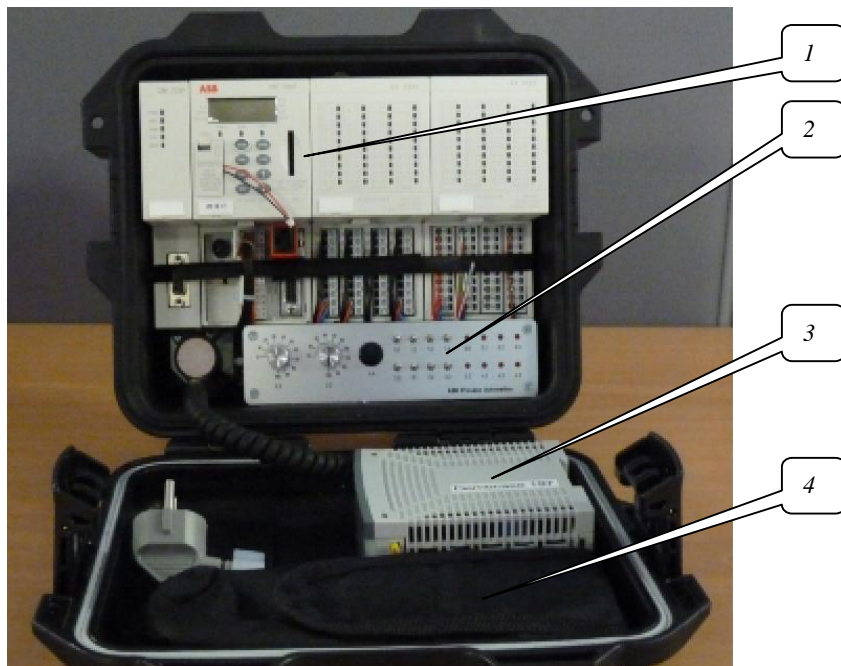


Рис.1.1 – Загальний вигляд кейса.

На рис. 1.2 показана узагальнена структурна схема та склад кейса, який входить до ПТК. Контролер має блочно-модульну конструкцію, яка складається зі спеціальної термінальної бази для модуля процесора (*CPU Terminal base* типу *TB711*); модуля центрального процесора (*CPU* типу *PM783*); термінальної бази для модулів введення/виведення сигналів (*I/O Terminal Unit*) і різноманітних сигнальних та функціональних модулів (*S700 I/O MODULE*), які монтуються на базу (для 3, 5 та 8 модулів). Також на термінальних базах присутні послідовні інтерфейси для мережного обміну (*RS-232/485* з протоколом *ModBus*); для діагностики та конфігурування (*RS-232/485 Diagnostics*) і *Ethernet* для завантаження програм користувача (*RJ45* з протоколом *TCP/IP*).

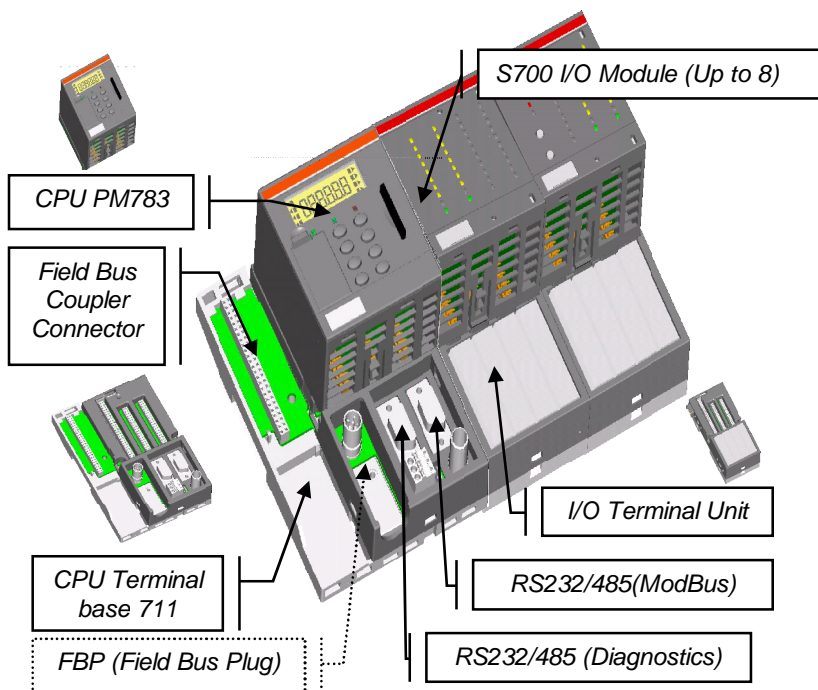


Рис. 1.2 – Апаратна частина ПТК FREELANCE 700 F.

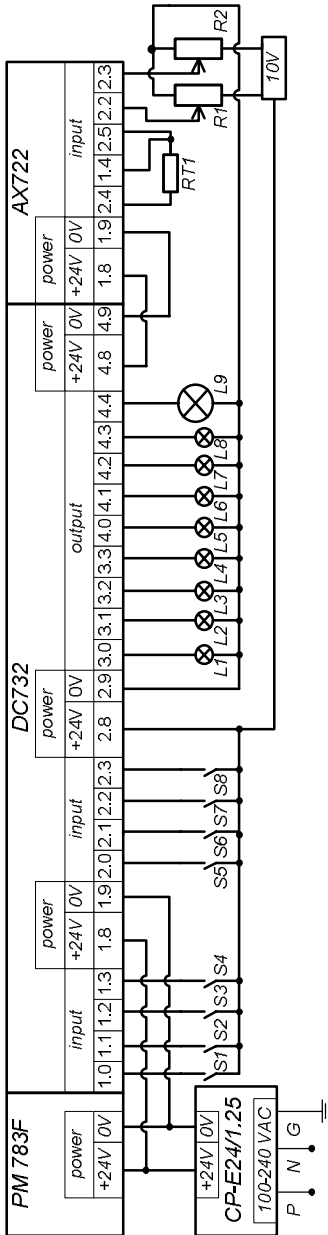


Рис. 1.3 – Принципова електрична схема кейса AC700F.

Додатково до складу ПЛК може входити шина для підключення інтерфейсного модуля (*Field Bus Coupler Connector*) та інтерфейсний з'єднувач польовий шини (*FBP – Field Bus Plug*).

Процесова станція (ПЛК) в кейсі моделі *TYPE 7-DC-RB-PB-AC700F* (див. рис. 1.1) складається з модуля центрального процесорного пристрою (ЦПП) типу *CPU PM783F*, модуля дискретних входів/виходів типу *DC732F*, модуля аналогових входів/виходів типу *AX722F*, комунікаційного модуля типу *CM772F* та блоку живлення *CP-E 24/1.25*. Крім того, до складу кейсу входять імітатор дискретних та аналогових вхідних сигналів і вихідних елементів, кабель *RJ-45* з *crossover* адаптером для безпосереднього підключення ПЛК до ПК інтерфейсом *Ethernet*. Якщо передбачається підключення ПЛК до концентратора або маршрутизатора, то адаптер не використовують. Принципова електрична схема кейса представлена на рис.1.3.

**Принцип роботи кейса AC700F.** Вмикання ПЛК відбувається відразу після підключення блоку живлення (БЖ) до електричної мережі (100...240 VAC). На схемі перемикачі *S1...S8* підключені до дискретних входів *Input 1.0...1.3* та *Input 2.0...2.3* модуля *DC732F*. Кожен з перемикачів комутує напругу +24 В від внутрішнього джерела живлення сигнального модуля. При замиканні перемикача на відповідному дискретному вході модуля формується сигнал *TRUE*. Таким чином, можна імітувати сигнали дискретних датчиків (*ON/OFF*), керувати режимами роботи установки (*Ручний/Автоматичний*) або параметром технологічного процесу (*Більше/Менше*). Для індикації режимів роботи або аварій використовуються світлодіоди червоного світлу, які підключені до виходів *Output 3.0...3.3* та *Output 4.0 ... 4.3* модуля *DC732*. До виходу *Output 4.4* модуля підключена лампа зеленого світла, яка імітує роботу нагрівача. До складу імітатора входять два потенціометра, які формують аналоговий сигнал *0...10 В* постійного струму. Вони підключені до входів *Input 2.2* та *2.3* модуля *AX722F*. До складу також входить термометр опору з НСХ ТСП-100, який підключений по трьох провідною схемою до аналогових входів *Input 2.4* та *1.4, 2.5* модуля *AX722F*.

## **1.2 Загальні відомості про кейс S700 Remote I/O**

Загальний вигляд кейса *FREELANCE 700F* моделі *DC-RB-PB-S700 Remote I/O* представлений на рис. 1.4. На цьому рисунку цифрами позначено розташування основних елементів кейса: 1 –мережний модуль для віддаленого доступу та сигнальні модулі; 2 – імітатор вхідних сигналів та вихідних елементів; 3 – блок живлення; 4 – пенал з додатковим приладдям.

Модулі монтуються на спеціальну термінальну базу. Ліворуч в термінальну базу встановлюється сигнальний модуль з інтерфейсом *PROFIBUS*, далі встановлюються звичайні сигнальні модулі видаленого введення/виведення серії *S700*. Термінальна база може мати 3, 5 або 8

слотів для підключення сигнальних модулів. В комплекті до кейсу включений інтерфейсний кабель *PROFIBUS*, за допомогою якого кейс *AC700F* (ПЛК) з'єднується з кейсом *S700* (*RTU*). Крім того, до складу кейсу *S700* входять імітатор дискретних та аналогових вхідних сигналів і вихідних елементів подібно до кейсу *AC700F*, який розглянутий вище.

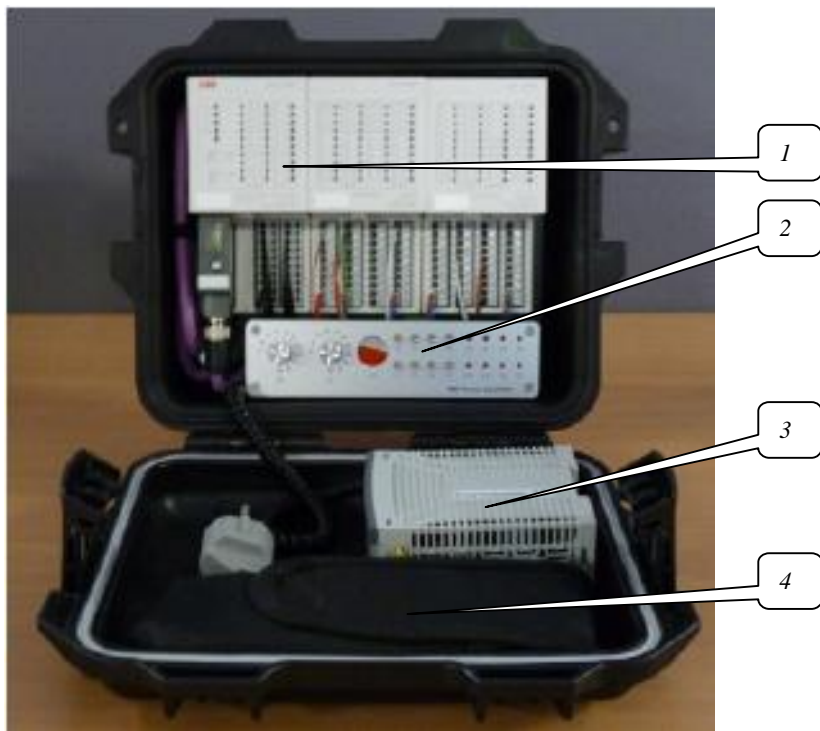


Рис.1.4 – Загальний вигляд кейса *S700*.

До складу кейса *S700* входять такі модулі:

- модуль з інтерфейсом *PROFIBUS* та дискретними сигналами введення/виведення *DC705F*;
- модуль аналогового введення/виведення *AX721F*;
- модуль дискретного введення/виведення *DX731F*.

На рис. 1.5 представлена принципова електрична схема кейса *S700*.

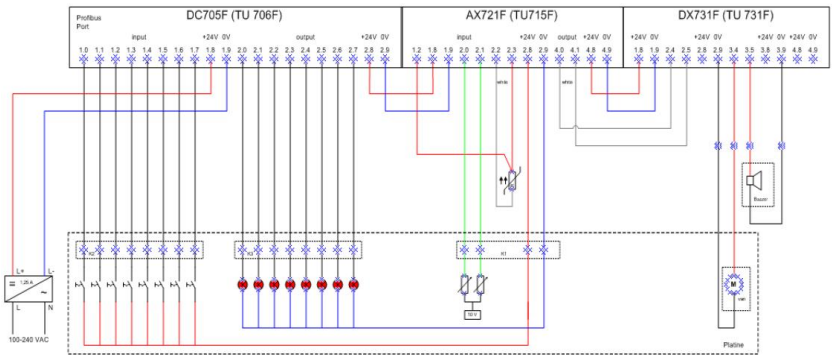


Рис. 1.5 – Принципова електрична схема кейса *S700*.

**Принцип роботи кейса.** Вмикання модулів відбувається відразу після підключення блоку живлення (БЖ 220/24V) до електричної мережі (100...240 VAC). На схемі перемикачі *SI...S8* підключені до дискретних входів *Input 1.0...1.7* модуля *DC705F*. Кожен з перемикачів комутує напругу +24 В від внутрішнього джерела живлення сигнального модуля. При замиканні перемикача на відповідному дискретному вході модуля формується сигнал *TRUE*. Таким чином, можна імітувати сигнали дискретних датчиків (*ON/OFF*). Для індикації стану дискретних виходів світлодіоди червоного світлу, які підключені до виходів *Output 2.0...2.7* модуля *DC705F*. До входів *Input 2.0* та *2.1* модуля *AX721F* за однопровідною схемою підключені два потенціометра, які формують аналоговий сигнал *0...10 В* постійного струму. Також до входів *Input 1.2, 2.2* та *2.3* модуля *AX721F* за 3-х провідною схемою підключений термістор. До дискретних виходів *Output 2.9* та *3.4* модуля *DX731F* підключений електродвигун постійного струму. До дискретних виходів *Output 3.5* та *3.9* модуля *DX731F* підключений електричний зумер.

### 1.3 Архітектура побудови та технічні характеристики ПТК *FREELANCE 800F*

Архітектура ПТК є дворівневою та складається з рівня контролера (польового) і рівня оператора (ЛМІ). На малюнку 1.6 представлена повна архітектура ПТК *FREELANCE 800F*. В ПТК всі роботи з розроблення ППЗ виконуються за допомогою єдиного програмного середовища – *Control Builder F*, який працює спільно із засобом візуалізації та управління *DigiVis*. При цьому функції інженерної та операторської станції можуть виконуватися як на одному комп'ютері, так і окремо.

На польовому рівні можливо як пряме підключення польових засобів автоматизації до ПЛК, так і мережне – з використанням модулів віддаленого введення/виведення (*RTU*). Додатково до складу системи управління може входити панель оператора (ПО) або ПК, який виконує функцію засобу ЛМІ. Однак на даному етапі ознайомлення з ПТК буде розглянуто лише взаємодія ПЛК, в якому датчики і виконавчі механізми підключені безпосередньо до модулів введення/виведення. Надалі буде розглянута верхня складова ПТК – середовище *DigiVis*, яке в повному обсязі виконує функції ЛМІ, навіть SCADA-системи. Далі також будуть розглянуті мережні можливості ПТК, тобто інтерфейси та протоколи обміну. Для цього додатково до кейсу буде підключений інший кейс моделі *TYPE 8-DC-RB-PB-S 700*, який є системою віддаленого введення/виведення типу *S700-I/O*.

Операторській рівень складається з стандартного ПК (Operator Station – *VIS*) з ОС *Microsoft Windows* та встановленим додатком *DigiVis*. Оператор може одночасно спостерігати за параметрами технологічного процесу та за ходом виконання послідовних функціональних програм. Також на операторському рівні знаходиться інжинірингова станція – стандартний ПК або ноутбук з операційною системою *Microsoft Windows* та встановленим додатком *Control Builder F*. Тут розробляється та тестується ППЗ.

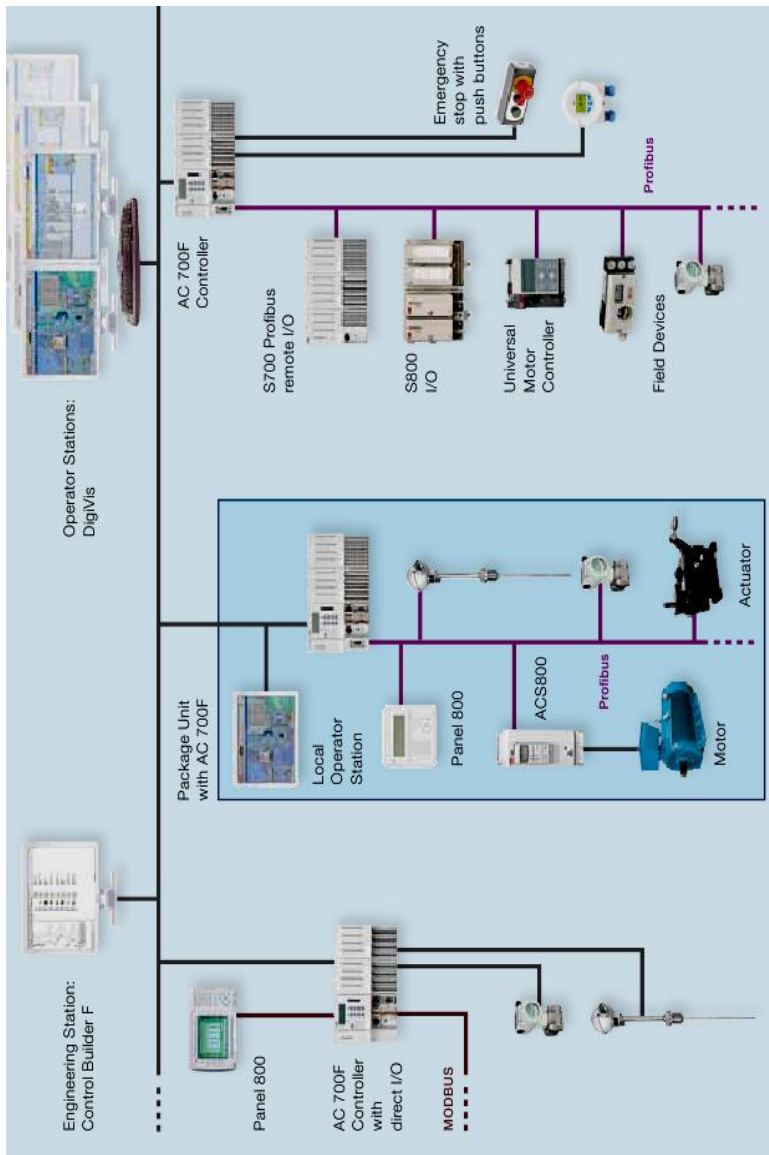


Рис. 1.6 – Архітектура ПТК *FREELANCE 800*.

На польовому рівні головним пристроєм є ПЛК (*Process Station – D-PS*). Спільне використання функціональних можливостей, середовища розроблення та модульність комплексу забезпечують легкість і гнучкість застосування, масштабованість та власне роботу РСУ.

Також на польовому рівні система управління може складатися з одного або декількох ПЛК типу *AC700F* або *AC800F*, які можуть бути розширені різними модулями введення/виведення серії *S700* за допомогою польових шин. При цьому є можливість конфігурування як стандартної системи, так і системи з резервуванням (*Redundant process station – D-PS/RED*). Контролер типу *AC 700F* може здійснювати контроль та управління технологічним процесом обробляючи до 300 сигналів.

Об'єднання декількох ПЛК на польовому рівні здійснюється за допомогою мережі *Ethernet*. Операторський та польовий рівні об'єднуються в РСУ за допомогою стандартної мережі *Ethernet*. При цьому середовищем передачі даних може бути кручена пара типу *UTP cat.5*.

Тому, у зв'язку з наведеним вище, ПТК *FREELANCE* називають гібридною компактною системою управління технологічним процесом.

Отже, розглянемо склад, технічні характеристики та порядок застосування компонентів ПТК на прикладі кейса моделі *TYPE 7-DC-RB-PB-AC700F*.

Модуль центрального процесорного пристрою – *CPU AC700 CPUPM783F* представлений на рис. 1.2. В модулі застосований потужний 32-розрядний процесор *RISC*-архітектури виробництва компанії *Motorola* типу *Motorola Power PC (MPC8247)*. Час циклу сканування ПЛК може становити значення до 2 мс. На платі ЦПУ встановлена оперативна пам'ять об'ємом 4 *MB* (типу *S-RAM*) або напівпостійна пам'ять об'ємом 16 *MB* (типу *SD-RAM*) з резервною батареєю для збереження даних. В кейсі на термінальну базу з одним центральним процесором можна підключити 2 модуля введення/виведення. В нижній частині мо-

дуля процесора знаходиться термінальний блок послідовного інтерфейсу *SER (COM1)* з пружинними затискачами для швидкого підключення сполучних провідників. Залежно від обраної конфігурації даний інтерфейс забезпечує з'єднання або по *RS-232*, або по *RS-485* зі швидкістю обміну від *1200 bps* до *38400 bps*. Підтримується обмін в режимі «Головний/Підлеглий» (*Master/Slave*) за протоколом *ModBus ASCII/RTU*. Для діагностики стану ПЛК передбачений спеціальний порт *DIAG (COM2)* з інтерфейсом обміну *RS-232* та з'єднувачем типу *SUB-D* типу *F*. На борту модуля центрального процесора є мережний інтерфейс *Ethernet (RJ45)* для завантаження програм користувача та взаємодії з програмою *DigiVis* для реалізації ЛМІ, тобто візуалізації технологічних процесів. Ліворуч від процесорного модуля в кейсі встановлено комунікаційний модуль *CM772F* для обміну по шині *Profibus Master* за протоколом *DP-V0/V1* зі швидкістю передачі даних від *9,6 kBit/s* до *12 MBit/s*.

Зміст функцій щодо створення ППЗ, які передбачені в ПТК, відповідає базовому обсягу, що визначений в стандарті *IEC61131-3*. Більш того, для спеціальних завдань можна додавати функціональні блоки, які створені користувачем. Під час конфігурування потужність та швидкість контролерів можна легко адаптувати до вимог технічного завдання з автоматизації. Програми в контролері виконуються на основі багатозадачної операційної системи реального часу (ОСРЧ), що забезпечує гнучкість стратегій для алгоритмів оброблення вхідних даних.

Операційна система контролерів має два типи *задач (TASK)*: системні та задачі користувача. Системні задачі контролюють операційну систему, наприклад, у випадку «холодного» запуску або у разі появи помилки модуля. Задачі користувача виконують прикладні програми користувача.

Для задач користувача є два режими виконання:

- до восьми задач, з часом окремих циклів від 5 мс до 24 годин;

- оброблення лише однієї задачі з максимально можливою швидкістю (режим *PLC*).

Системні задачі викликаються автоматично. Вони виконуються один раз у випадку виникнення таких подій:

- *RUN* (РОБОТА);
- *STOP* (ЗУПИНКА);
- *COLD START* («ХОЛОДНИЙ» запуск);
- *WARM START* («ГАРЯЧИЙ» запуск, після відновлення живлення);
- *REDUNDANCY TOGGLE* (Перемикання резервування);
- *ERROR* (ПОМИЛКА).

Для управління задачами та режимами роботи ПЛК, а також для відображення стану ПЛК на передній панелі процесорного блока є кнопки та ЖК-дисплей.

Для складних додатків можна налаштувати вісім циклічних задач із зазначенням пріоритетів і регульованим часом циклу, а також циклічні задачі (режим *PLC*), які працюють з максимально можливою швидкістю. Ця багатозадачність дозволяє розробити додатки, які відповідають усім вимогам до керування технологічним процесом, і в той же час вирівнюють навантаження на процесор. Це сприяє економії ресурсів ПЛК, необхідних для програми користувача.

Комунікаційний модуль шини *Profibus CM772F* дозволяє здійснювати обмін даними з іншим ПЛК або модулями віддаленого введення/виведення по польовій шині *PROFIBUS DP*. Модуль монтується зліва від процесора на тій же термінальній базі. Якщо комунікаційний модуль відсутній, то замість нього необхідно встановити спеціальний модуль-«заглушку». Обмін даними між процесором та шиною-з'єднувальним вимикачем здійснюється за допомогою шини позаду процесора (інтерфейс вимикача), яка вбудована в термінальну базу. Обмін даними реалізований через двохпортову оперативну пам'ять.

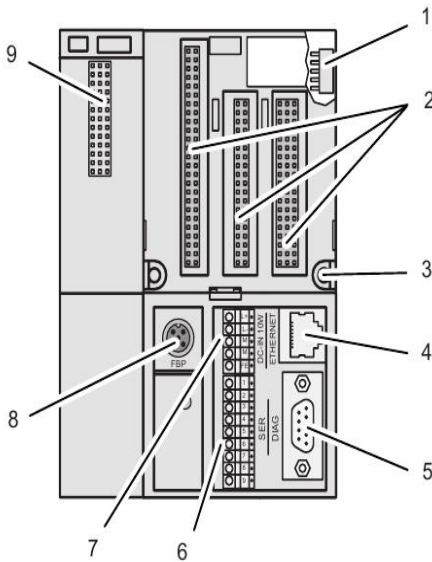


Рис. 1.7 – Схема термінальної бази.

Термінальна база процесора типу *TB711F* слугує для монтажу модулів та їх електричного з'єднання між собою. На рис. 1.7 показано розміщення слотів та з'єднувачів. Цифрами на схемі позначені наступні елементи:

- 1 – з'єднувач задньої шини введення/виведення;
- 2 – з'єднувачі для центрального процесора;
- 3 – отвори для монтажу на стіні або в шафі;
- 4 - інтерфейс *Ethernet*;

- 5 – послідовний діагностичний інтерфейс *DIAG (COM2)*;
- 6 – послідовний інтерфейс для обміну даними *SER (COM1)*;
- 7 – основні клеми для живлення ПЛК напругою 24 В пост. струму;
- 8 – з'єднувач шини *Feldbus* (резервний);
- 9 – з'єднувач типу *PROFIBUS*-ведучий пристрій (захищений за допомогою модуля-«заглушки», якщо не використовується).

Призначення контактів з'єднувача *SER (COM1)* показано на рис. 1.8. Для підключення живлення до ПЛК використовується 5-піновий з'єднувач з пружинними затискачами. Схема підключення живлення показана на рис. 1.9.

Сигнальні модулі системи *S700* можуть використовуватися як для безпосереднього введення/виведення спільно з контролером *AC700F*, так і в якості елемента ПСУ у будь-якому сегменті мережі *Profibus-DP*.

В табл. 1.1 представлені усі сигнальні модулі системи *S700*. Позначена сірим фоном підгрупа (стовпець 2, поз. 1...7) може використовуватися

тися в якості модулів безпосереднього введення/виведення спільно з контролером AC700F. Усі модулі (стовпець 3, поз. 1...15) можуть використовуватися в якості модулів віддаленого вводу/виводу в мережі Profibus-DP. У заголовку таблиці прийняті наступні скорочення: безп. – безпосереднє підключення, від. – віддалене підключення.

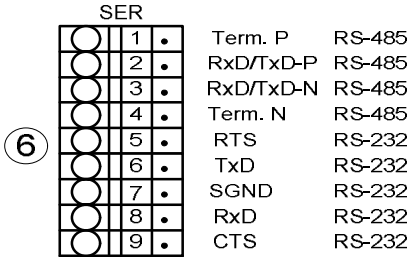


Рис. 1.8 – Схема підключення до інтерфейсів RS-232 та RS-485.

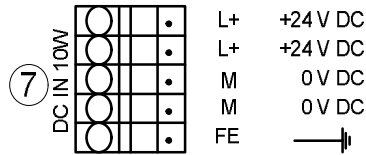


Рис. 1.9 – Схема підключення живлення ПЛК.

Таблиця 1.1 – Сигнальні модулі системи FREELANCE S700

№ з/п	Тип зв'язку		Тип модуля	Типі та кількість каналів	Вхідні канали	Вихідні канали
	пр.	від.				
1	2	3	4	5	6	7
1	Модулі безпосереднього введення	Модулі віддаленого введення	DC 732F	16 каналів цифр. введення 16 каналів цифр. виведення	24В пост. струму однопровідний	24В пост. струму, 0,5А
2			AI 723F	16 каналів аналог. введення АЦП – 12 біт + знак	0...10В, -10...+10В, 0/4...20мА Pt100/1000, Ni1000, цифр. вхід	-

1	2	3	4	5	6	7
3			<i>AX 722F</i>	8 аналог. каналів введення 8 аналог. каналів виведення (2 x 4), АЦП 12 біт + знак	аналогічно <i>AI 723F</i>	канали 0...3: -10...+10В, 0/4...20 мА канали 4...7: -10...+10В
4			<i>AO 723F</i>	16 каналів аналог. виведення (2 x 8), АЦП 12 біт + знак	-	канали 0...3 та 8...11: -10...+10В, 0/4...20 мА канали 4..7 и 12...15: -10...+10В
5			<i>DX 722F</i>	8 каналів цифр. введення, 8 каналів цифр. виведення (реле)	24В пост. струму	24В пост. струму, 110/230В змінного струму
6			<i>DX 731F</i>	8 цифр. каналів введення, 4 канали цифр. виведення (реле)	110/230В змінного струму	24В пост. струму, 110/230В змінного струму
7			<i>AI 731F</i>	8 аналог. каналів введення АЦП 15 біт + знак	8 <i>AI</i> : термопара, термометр опору, мВ/В, мА, кОм та 24В пост. струму 2-, 3- та 4- пров. підкл., 24В пост. струму, 5Вт	-

1	2	3	4	5	6	7
8			<i>DC 705F</i>	інтерфейс <i>FBP</i> та 8 каналів цифр. введення, 8 каналів цифр. виведення	24В пост. струму, однопров. підкл.	24В пост. струму, 0,5А
9			<i>DI 724F</i>	32 канали цифр. введення	24В пост. струму	24В пост. струму, однопров. підкл.
10			<i>DC 722F</i>	16 цифрових каналів введення та виведення	24В пост. струму (можливе 3-х провідне підкл.)	24В пост. струму, 0,5А
11			<i>DC 723F</i>	24 цифрових каналів введення	24В пост. струму (можливе 2-х або 3-х провідне підкл.) 24В пост. струму, 0,5А	-
12			<i>AX 721F</i>	4 канали аналог. введення та 4 канали аналог. виведення, АЦП 12 біт + знак	аналогічно <i>AI 723F</i>	-10...+10В, 0/4...20 мА
13			<i>AC 722F</i>	8 каналів змінного струму АЦП 12 біт + знак	аналогічно <i>AI 723F</i>	аналогічно <i>AX 722F</i>

1	2	3	4	5	6	7
14			<i>CD 722F</i>	2 лічильника, 2 ШИМ-канала, 2 канали цифр. введення, 8 каналів пост. струму	<i>RS-422</i> , <i>5-V-TTL</i> , вих. 2-транз каскад 24В, 1 <i>SIN-канал</i> введення пікової напр., інтерфейс <i>SSI</i>	ШИМ, 24В пост. струму, 0,5А
15			<i>DA 701F</i>	16 цифр. кан. введення, 8 цифр. кан. виведення, 4 кан. ан. введ., 2 кан. ан. вивед.	24В пост. струму для цифр. каналів, аналог. канали. аналогічно <i>AI 731F</i>	24В пост. струму, 0,5А, аналог. канали аналогічно <i>AI 731F</i>

## 1.4 Основні принципи роботи в середовищі *Control Builder F*

### 1.4.1 Початок роботи в середовищі *Control Builder F*

Розроблення ППЗ для системи управління на основі ПТК *FREELANCE 800F* починається з запуску програми з ім'ям *Control Builder F* (далі *CBF*). Для цього використовуйте або ярлик на робочому столі ПК, або натисніть ЛКМ на системну кнопку *Пуск* і послідовно позначте курсором рядки меню *Все програми* → *ABB Industrial IT* → *Freelance V9.2* → *Control Builder F*. Взагалі, виконавчим файлом програмного середовища є файл з ім'ям *CBF.EXE*, який знаходиться в папці зі шляхом доступу до нього «*C:\Program Files\ABB Industrial IT\Freelance\exe\\*.\**», для випадку стандартної інсталяції програми. Якщо використовується демо-версія програми, то в процесі її запуску з'явиться вікно, яке попереджає про це. В цьому вікні кількість днів її активності з можливих 100 діб. Далі відкриється стартове вікно програми, екранна форма якого показана на рис. 1.10. Це вікно складається з декількох зон: інформаційної верхньо-

го рядка ОС *Microsoft Windows* з кнопками керування віконним режимом (згордання, розгордання або закриття вікна); рядок основного меню з командами *Проект*, *Настройки*, *Справка*; панелі інструментів з кнопками-піктограмами, що дублюють основні команди меню; робочого простору та нижнього рядка з інформацією про шляхи до папок з раніше створеними проектами. За умовчанням шлях до папки з проектами такий: «*C:\Program Files\ABB Industrial IT\Freelance\proj\\*.\*.»*.

На даному етапі можливі три варіанти подальших дій: 1) Створення нового проекту; 2) Відкриття раніше створеного проекту; 3) Імпорт проекту з архівного файлу з розширенням «*\*.CSV»*.

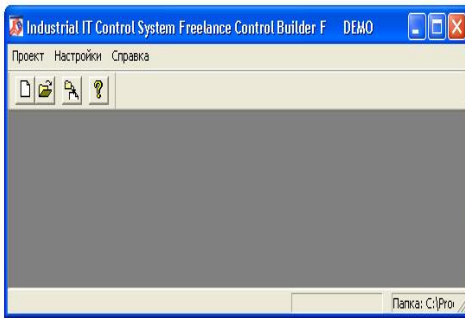


Рис. 1.10 – Стартове вікно програми.

*файла*: введіть ім'я нового проекту (наприклад, *project\_1*) і натисніть ЛКМ на кнопку *Сохранить*. Далі відкриється вікно конфігурації з екранної формою для заповнення загальних відомостей про файл проекту *Конфигурирование: Заголовок проекта: Тема проекта*. Ви можете заповнити порожні поля форми пізніше, тому ЛКМ натисніть на кнопку *OK*. В результаті відкриється бланк проекту, який показаний на рис. 1.11. На цьому бланку відображена основна інформація о проекті.

Якщо необхідно відкрити раніше створений проект, використовуйте стандартний формат *Открыть...* в меню *Проект*. У вікні, виберіть

Для створення нового проекту в рядку меню *Проект* виберіть команду *Новый...* або ЛКМ клікніть по інтуїтивно зрозумілою (відповідно до концепції роботи ОС *Microsoft Windows*) кнопці (*Создать...*) на панелі інструментів. У вікні з ім'ям *Создать проект* в поле *Имя*

ЛКМ потрібний проект і натисніть ЛКМ по кнопці *Открыть*. Усі проекти за замовчуванням зберігаються в папці зі шляхом «C:\Program Files\ABB Industrial IT\Freelance\proj\\*.»». Однак, якщо проект перебуває в іншому місці, то за стандартною процедурою за допомогою навігаційних кнопок на панелі інструментів та клікання ЛКМ по вибраному об'єкту у вікні *Открыть проект* можна вказати шлях до нього.

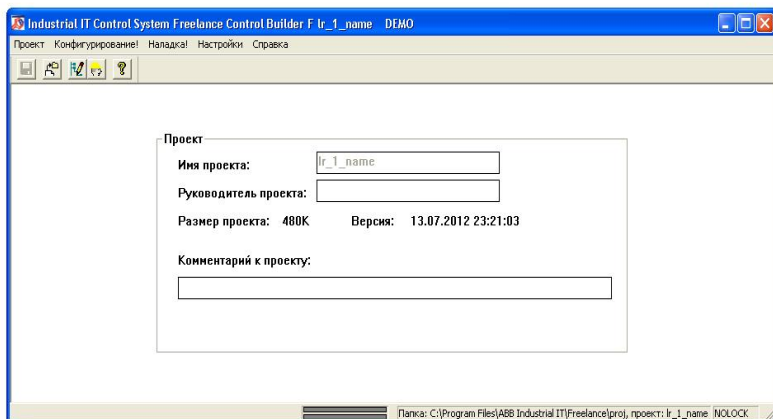


Рис. 1.11 – Бланк нового проекту.

Для відкриття проекту з архіву в форматі «\*.CSV» у меню *Проект* передбачена команда *Импорт...*, вибір якої активує вікно *Импорт проекта*. Тут теж необхідно вказати шлях до архівного файлу з проектом.

Після відкриття проекту або вилучення його з архіву відкриється вікно з бланком обраного проекту (див. рис. 1.11), подібно до випадку створення нового проекту.

З рис. 1.11 видно, що рядок меню і панель інструментів після відкриття проекту відрізняються від попередніх вікна (див. рис. 1.10). Рядок меню доповнився командами *Конфигурирование!* та *Наладка!* для вибору режимів роботи програми, а також відповідні їм кнопки на панелі інструментів. Крім того, розшириться перелік команд в меню *Проект* и

*Настройка.* Більш докладно функціональне призначення команд меню дивіться в інтерактивній довідці до програмі або в документації [1, 2, 3]. В середовищі *CBF* також працює контекстно-залежне меню, яке викликається за допомогою кліка ПКМ, яке дублює команди меню і функції кнопок на панелі інструментів. Крім того, якщо підвести курсор до кнопки інструменту, то через деякий час з'явиться або коментар, або команда, яка відповідає цьому інструменту.

Далі, необхідно перейти або до конфігурації проекту (якщо проект новий), або до його налагодження (якщо завантажений старий проект), використовуючи відповідні команди в меню або кнопки на панелі інструментів.

#### 1.4.2 Конфігурування структури апаратної частини проекту

Проектування системи управління для випадку створення нового проекту почніть з конфігурації «дерева» проекту. Для цього в рядку меню ЛКМ виберіть команду *Конфигурирование!* або клікніть ЛКМ по відповідній кнопці на панелі інструментів. У результаті активується режим конфігурування дерева проекту, екранна форма якого показана на рис. 1.12.

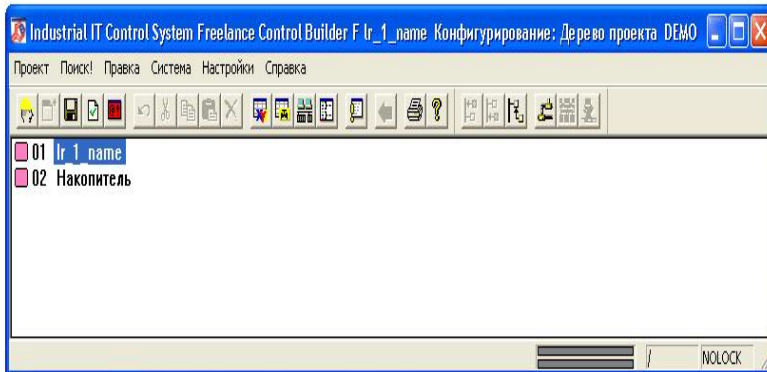


Рис. 1.12 – Вікно конфігурування дерева проекту.

Дерево проекту – це основний інструмент для управління усім проектом користувача та проведення пускових та налагоджувальних робіт. У структурі дерева відображаються всі дані про налаштування проекту.

Особливості властивостей дерева проекту:

- дані про налаштування проекту мають структурований зміст;
- визначені рівні завдань та тривалість циклів;
- програми зіставлені певному рівню завдань;
- дозволена зміна, копіювання і переміщення програм, вікон і журналів;
- програми перевіряються на достовірність, відображається стан їхнього оброблення;
- дозволений експорт та імпорт ресурсів та налаштувань проекту;
- програми користувача можуть завантажуватися як на операторські станції, так і на станції технологічних процесів.

В середовищі керування усіма сигналами, змінними та мітками, що містяться в налаштуваннях, здійснюється в загальній *базі даних проекту*. Це програмна частина проекту системи управління, яка потім зіставляється у відповідність апаратної складової проекту.

Об'єктами бази даних є:

- перелік змінних (вхідні, вихідні та внутрішні змінні);
- перелік тегів (функціональні блоки);
- тренди та аларми і повідомлення;
- кадри ЛМІ та лицеві панелі;
- програми.

Оскільки база даних є єдиною для усього проекту, введення даних здійснюється лише один раз. При цьому знижується імовірність помилок. Наявність єдиного файлу бази даних проекту полегшує завдання архівування та резервного копіювання.

Список змінних і тегів створюється автоматично в процесі генерування програми користувача.






Інші функціональні можливості переліку змінних та тегів:





- зміна імен, типу модуля, даних та коментарів у всьому проекті;
- пошук та перегляд на основі заданих критеріїв пошуку;
- функція перехресних посилань для швидкого пошуку в усіх програмах та вікнах, де міститься зазначена змінна або тег (пошук по всьому проекту).



Робоча область вікна конфігурування для новоствореного проекту складається з двох вузлів: основного, призначеного для користувача з ім'ям *lr\_1\_name* та вузла накопичувача *Накопитель* (див. рис. 1.12), який є тимчасовим архівом користувача. В режимі конфігурування верхній системний рядок інформує розробника про ім'я проекту та режимі роботи програми. Рядок меню змінюється в порівнянні з рядком в попередньому вікні (див. рис. 1.11). Додатково в системному рядку з'являються команда *Поиск!* та команди меню *Правка* та *Система*. Панель інструментів теж розширюється. При цьому вона розділена на функціональні зони з кнопками, які відповідні деяким командам рядка меню.

Інструменти першої зони призначені для керування деревом проекту: *Наладка*, *Заголовок*, *Сохранить*, *Проверить*, *Показать список оши-*



*бок*. Це відповідно кнопки:     . Далі розташовані інструменти, які реалізують стандартні функції редагування проекту –




    . Цим кнопкам відповідають команди: *Отменить*, *Вырезать*, *Копировать*, *Вставить*, *Удалить*. Вони повністю збігаються з відповідними командами ОС *Microsoft Windows* і мають той же набір гарячих клавіш для їх виклику. Наступна зона з інструментами відпові-

дає меню *Система*. Кнопки     дублюють команди виклику компонентів проекту: *Список переменных*, *Список тэгов*, *Структура аппаратных средств*, *Типы структурированных данных*. Наступна

кнопка  – *Список контрольных точек*; кнопка  з покажчиком

зеленого кольору для повернення в основне вікно дублює команду *Назад!*;

кнопки   для друку кадру та виклику довідки. Для редагування структури дерева проекту є кнопки для вставки елементів дерева

  . Це команди: *Вставити вище*, *Вставити нижче* і *Вставити наступний рівень*. Так може бути створена складна ієрархічна структура проекту. Завершує панель інструментів зона з кнопками

   для виклику вікон *Сеть*, *Конфигурирование связей* та *Автономно*.

Для новоствореного проекту деякі функції меню та кнопки неактивні, однак в процесі конфігурації вони стають доступними для використання. В режимі конфігурації також працює опція контекстно-залежного меню. Однак для активування контекстного меню необхідно обов'язково підвести курсор миші до потрібного елемента дерева. На рис. 1.13 показаний фрагмент вікна з відкритим контекстним меню в режимі конфігурування дерева проекту. Тут деякі команди недоступні, так як в дереві є всього два вузли, і немає елементів нижнього рівня.

Далі можна діяти в такий спосіб: або спочатку створити програмну частину (*software*), а потім – апаратну частину (*hardware*) проекту, або навпаки.

Система управління на базі ПТК *Freelance* може бути повністю налаштована, або *online* (ПК з *CBF* підключе-

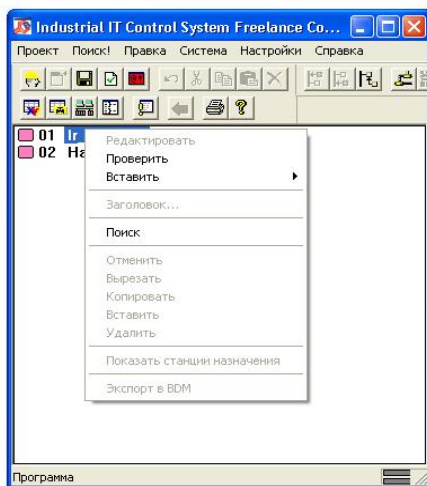


Рис. 1.13 – Вид контекстного меню.

ний до ПЛК), або *offline* (ПЛК відсутній, використовується емулятор ПЛК). При автономному налаштуванні підключення до контролера не обов'язково. Створений в процесі автономного налаштування додаток можна пізніше завантажити в контролер. Якщо передбачається робота з реальним ПЛК, наприклад, зі складу кейсу, то виберіть об'єкт – *Процессовая станция (AC700F)*. У разі відсутності ПЛК виберіть об'єкт – *Эмулятор контроллера*. Емулятор дозволить провести налагодження програми користувача без ПЛК.

Проте для запуску емулятора ПЛК необхідно його перш за все запустити та налаштувати. Для цього виконайте ряд кроків.

1) За допомогою стандартних процедур в ОС *Microsoft Windows* запустити на ПК браузер *Internet Explorer*.

2) В полі введення *TCP*-адреси введіть наступний рядок: <http://127.0.0.1:8888/>. В результаті відкриється сайт адміністратора для керування емулятором ПЛК *Freelance*, який показаний на рис. 1.14. Для швидкого запуску емулятора додайте відкритий сайт до папки *Избранное* браузера.

3) Введіть ідентифікаційний номер станції (*ID=\** ) в полі *Station ID*, який відповідає номеру ресурсу в розподіленій мережі під час конфігурування проекту. Далі ЛКМ натисніть на кнопку *Start Controller*.

Отже емулятор ПЛК готовий до завантаження програми користувача із середовища *CBF*.

Для створення апаратної структури ПЛК в меню *Система* виберіть команду *Аппаратная структура* або відповідну кнопку на панелі інструментів. Відкриється вікно, екранна форма якого представлена на рис. 1.15.

Інженерна станція у складі структури апаратної частини додається автоматично у вигляді схематично зображеного ноутбука. Зображення ноутбука з підключенням до шини *Ethernet* на рис. 1.15 відповідає інженерній станції розробника ППЗ зі встановленим середовищем *Control Builder F*. Ліворуч у вікні розташовано дерево апаратної частини проекту з вузлом *HWSYS (HWSYS)*.

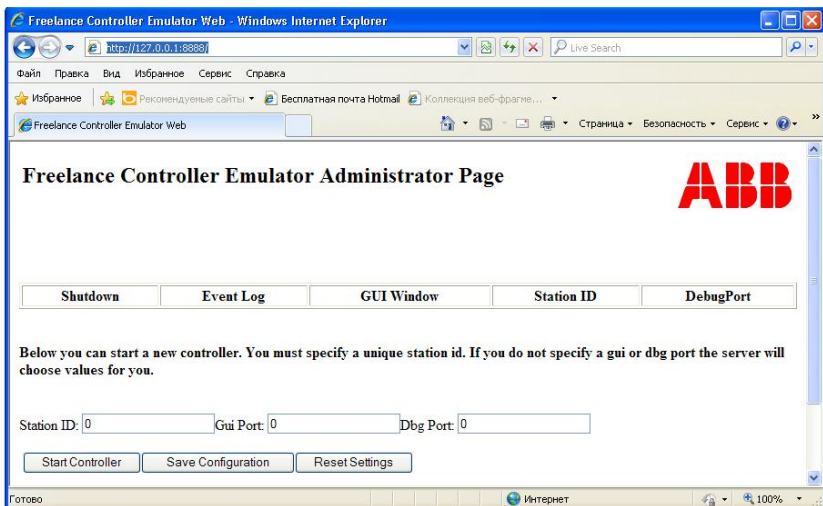


Рис. 1.14 – Вікно керування емулятором ПЛК.

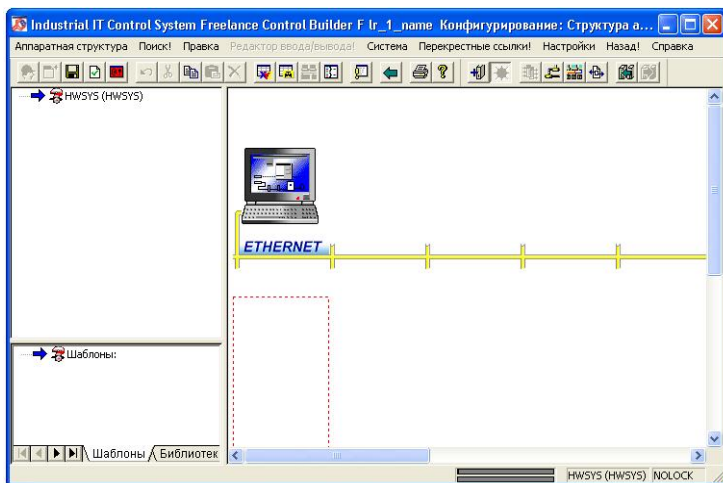


Рис. 1.15 – Вікно конфігурування апаратних засобів проекту.

Клікніть ЛКМ в робочому вікні ліворуч та нижче шини *Ethernet*. З'явиться рамка з пунктиру червоного кольору (див. рис. 1.15). Підведіть курсор в зону, обмежену рамкою, і натисніть ПКМ для виклику

контекстного меню. Виберіть в меню команду *Включить*, а в списку нових об'єктів виберіть потрібний, наприклад *Эмулятор*. Поруч з емулятором аналогічним чином створіть в одному проекті структуру реального ПЛК, наприклад, типу *AC700F*. Для візуалізації технологічного процесу та спостереження за параметрами в структурі проекту можна створити операторську станцію – *D-OS*. Тому у вікні конфігурації апаратних засобів клікніть ЛКМ зверху шини *Ethernet* та праворуч інженерної станції для появи нової червоної рамки. Усередині рамки ПКМ викличте контекстне меню та виберіть у ньому команду *Вставить* і виберіть об'єкт – *Операторская станция D-OS*. Остаточна структура проекту повинна відповідати зображеній на рис. 1.16.

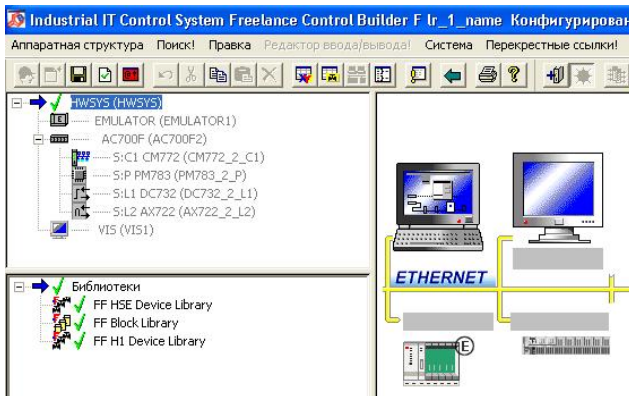


Рис. 1.16 – Структура апаратних засобів проекту.

Перевірте на правдоподібність структуру та поверніться в дерево проекту, застосувавши команду *Назад!*. Далі перевірте проект в цілому на правдоподібність. В результаті система повідомить про дві помилки в проекті, як це показано на рис. 1.17.

Ця ситуація відповідає дереву проекту з вузлами проекту та накопичувача, яке зображено на рис. 1.12. Програмну частину проекту можна конфігурувати пізніше. Тому знову відкрийте вікно конфігурування

апаратних засобів проекту для налаштування модулів, що входять до складу ПЛК в кейсі.

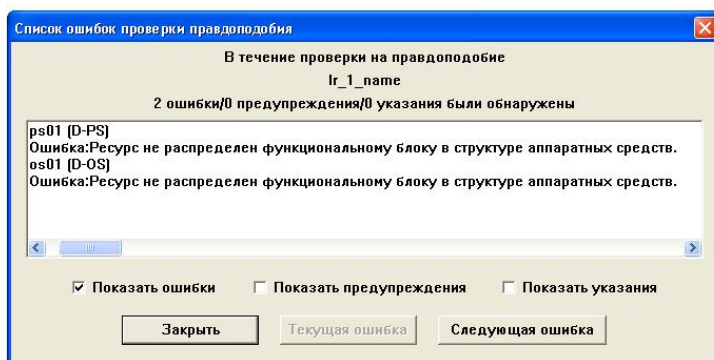


Рис. 1.17 – Результат перевірки на правдоподібність проекту.

Для конфігурування модулів в контролері двічі кликніть ЛКМ по міні-зображенню шаблону ПЛК. Відкриється шаблон ПЛК, який відповідає заданому раніше типу (*AC700F*). Шаблон це зображення термінальної бази для модулів: комунікаційного, процесорного та сигнальних (усього до 8 штук). Заповніть шаблон модулями, які входять в кейс, відповідно до рис. 1.3. Для цього спочатку ЛКМ кликніть по зоні шаблону для центрального процесора, далі ПКМ відкрийте контекстне меню і виберіть команду *Вставить*. Можна також перейти до вікна вибору модуля, якщо двічі кликнути ЛКМ по зоні шаблону. Тип модуля для вставки залежить від обраної зони шаблону ПЛК. Тому для вставки модуля центрального процесора в шаблон ПЛК у вікні вибору пропонується лише один варіант – процесорний модуль типу *CPU PM783F*. В інших зонах шаблону ПЛК для сигнальних модулів послідовно виберіть модулі *DC732* та *AX772*, які входять до складу кейса (див. рис. 1.3). Якщо на даному етапі мережні можливості системи не використовуються, то можна не вставляти ліворуч від процесорного модуля комунікаційний модуль *CM772* для шини *PROFIBUS DP Master*.

Остаточний шаблон ПЛК буде відповідати екранній формі, яка представлена на рис. 1.18. Відзначимо, що можна подвійним кліком ЛКМ по модулю відкрити його збільшене зображення з інформацією про його тип, місце розміщення та адресу. Наступний подвійний клік ЛКМ відкриє вікно налаштування властивостей входів/виходів відповідного модуля.

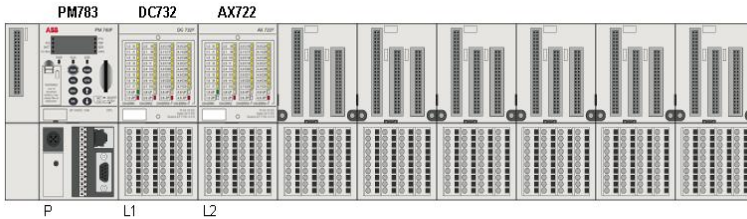


Рис. 1.18 – Шаблон ПЛК з модулями.

На рис. 1.19 показано вікно параметрування модуля *DC732*. В цьому модулі частина входів може бути налаштована на роботу в режимі виходів. Тому у вкладці *Конфигурирование выводов* необхідно ЛКМ поставити крапки у поля навпроти виходів *C16\_O...C19\_O* і *C24\_O...C28\_O* в стовпчиках, що позначені «1» в зоні *Значения замещения выводов* вкладення відповідно до схеми, яка зображена на рис. 1.3.

На рис. 1.20 показано вікно налаштування модуля *AX772* з відкритим вкладенням конфігурування аналогових входів відповідно до схеми, зображеної на рис. 1.3.

Далі після проведення налаштування модулів ПЛК перевірте на правдоподібність конфігурацію апаратних засобів. У разі позитивного результату перевірки поверніться у вікно конфігурації дерева проекту (див. рис. 1.12) за допомогою кліка ЛКМ по кнопці з зеленою стрілкою. В іншому випадку спочатку усуньте помилки, використовуючи вікно зі звітом про проведену перевірку та помилки. При переході до вікна дерева проекту в діалоговому вікні *Указание* збережіть отриману конфігурацію кліком ЛКМ по кнопці *Да*.

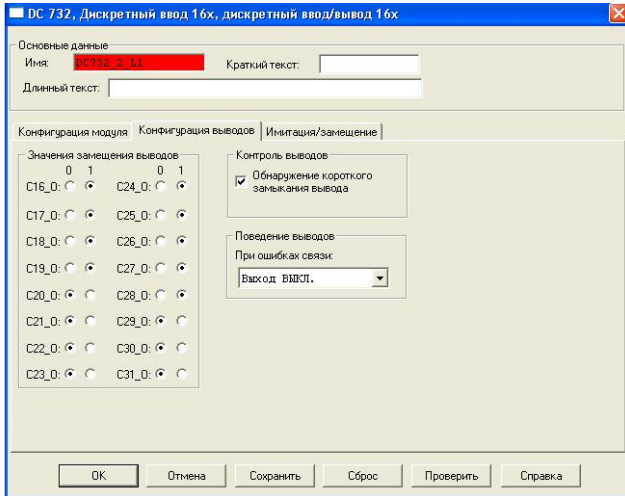


Рис. 1.19 – Вікно конфігурування входів модуля DC732.

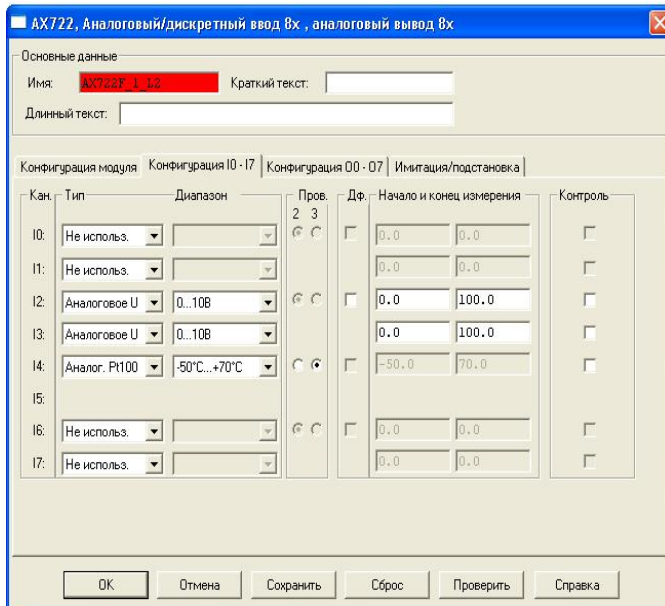


Рис. 1.20 – Вікно конфігурування входів модуля AX722.

### 1.4.3 Розроблення програмної частини проекту

Після повернення до вікні конфігурації дерева проекту, яке зображене на рис. 1.12, почніть складати його програмну складову частину. Для цього можна використовувати інструменти вставки елементів дерева проекту з відповідної зони або команди з меню.

Отже, за допомогою кнопки або контекстного меню створіть елемент наступного рівня для основного вузла проекту під назвою *lr\_1\_name (CONF)*, тобто програмну конфігурацію. При цьому послідовно будуть відкриті два діалогових вікна: вікно вибору елемента та вікно введення даних в заголовок елемента конфігурації проекту.

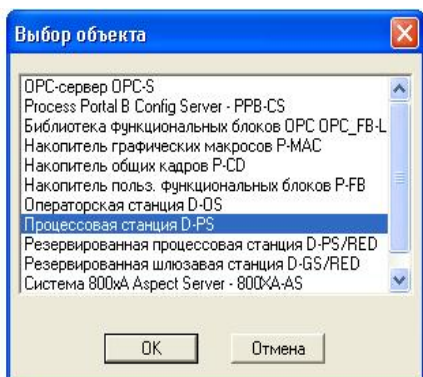


Рис. 1.21 – Вибір об'єкта наступного рівня в конфігурації.

Наступним кроком необхідно створити всередині конфігурації новий об'єкт наступного рівня з переліку, що показаний на рис. 1.21. Цим об'єктом буде процесова станція *D-PS (ps01)*, тобто програмна складова ПЛК. Після натискання на кнопку *OK* відкриється вікно основних налаштувань ресурсів ПЛК. В результаті, у складі конфігурації процесової станції буде автома-

тично створено два списки задач: призначений для користувача – *ps01.USRTask (TASKLIST)* та системний – *ps01.SYSTask (TASKLIST)*. Перший список визначає прикладні завдання, які визначені користувачем, а другий – призначений для управління системними ресурсами та режимами роботи самого ПЛК. Для згортання та розгортання списку елементів дерева використовуються кнопки «+» і «-» у відповідних рядках дерева. Згорніть список з системними завданнями. У рядку задач користу-

вача описаним вище способом створить новий елемент наступного рівня – завдання типу *По умовчанию TASK* з ім'ям *ps01USRTask*. В параметрах даної задачі залиште усе параметри за умовчанням. Таким чином, всередині списку для завдань користувача з'явиться новий елемент – завдання з ім'ям *ps01USRTask (Task)* (*Циклический, T#500ms*). Тепер всередині цього завдання можна створити або *SFC-програму*, або *Список программ PL*. Створить список програм *ps01USRTask (PL)* (*Вкл*), а в ньому програму на мові *FBD* – *ps01USRTask (FBD)*. Додайте в дерево проекту операторську станцію *os01 (VIS)*. Таким чином, ви отримаєте конфігурацію, аналогічну представленій на рис. 1.22.

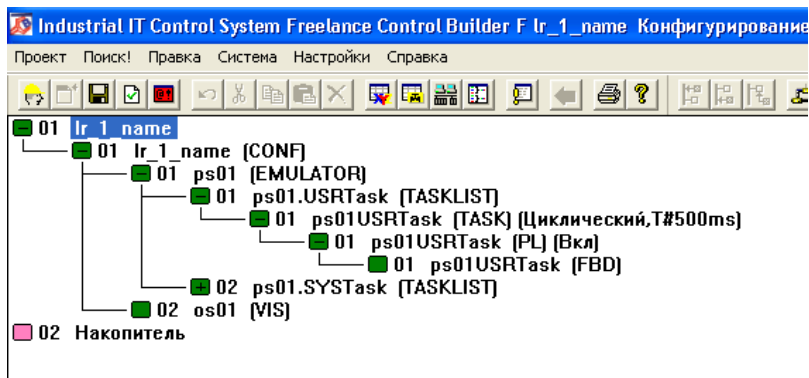


Рис. 1.22 – Дерево проекту.

Проте, в програмному середовищі *CBF* вже є готові шаблони, які прискорюють процес розроблення проекту. Наприклад, замість усіх попередніх дій по створенню програмної структури проекту можна скористатися готовим шаблоном користувача. Для цього в меню *Правка* виберіть команду *Импорт блока...*, а в вікні, яке з'явиться *Импорт части проекта* – потрібний шаблон, наприклад, *Template\_01.prt*. Цей шаблон має готову конфігурацію, яка складається з операторської та процесової станцій зі списком задач користувача та системних задач. Шаблон буде імпортовано в *Накопитель*. Тепер вміст шаблону необхідно перенести

до проекту. Для цього виділіть ЛКМ і не відпускаючи кнопку проведіть курсором по всіх елементах шаблону (функція *drag-and-drop*). Далі, перенесіть виділену область в елемент конфігурації проекту з ім'ям *lr\_1\_name (CONF)*. Далі дійте вищеописаним способом до одержання потрібної конфігурації, яка показана на рис. 1.22.

Зараз можна усунути помилки проекту, які показані у вікні на рис. 1.17. Перейдіть у вікно конфігурації апаратних засобів проекту для призначення кожному апаратному пристрою проекту відповідного програмного ресурсу. Для цього підведіть курсор миші в зону із зображенням *Емулятора* та за допомогою контекстного меню виберіть команду *Распределение ресурсов...*, а у вікні ресурсів виберіть *ps01*. В результаті буде сформована структура апаратної частини з призначеним ресурсом – емулятором ПЛК *ps01*, яке показане на рис. 1.23. Призначте подібним чином ресурс *os01* для операторської станції. Надалі, вибираючи потрібний ресурс, тобто номер *ID*, можна проводити наладку проекту на обох об'єктах (ПЛК або емулятор), по черзі призначаючи їх в якості ресурсу проекту, який буде задіяний.

Перевірте структуру апаратних засобів проекту на правдоподібність. Якщо помилок немає, то за допомогою команди *Назад!* або відповідної кнопки на панелі інструментів перейдіть у вікно з деревом проекту (див. рис. 1.22). На діалогове вікно з повідомленням про зроблені зміни конфігурації і питанням про збереження змін у разі згоди натисніть на кнопку *Да*. Також перевірте на правдоподібність проект в цілому.

#### 1.4.4 Конфігурування мережної структури проекту

Раніше, ніж завантажити програмні ресурси проекту до емулятору ПЛК та в систему візуалізації *DigiVis*, необхідно налаштувати мережну конфігурацію пристроїв проекту. Для цього клікніть ЛКМ по кнопці



для відкриття діалогового вікна. Далі налаштуйте *IP*-адреси для емулятора та операторської станції *DigiVis*, як це показано на рис. 1.24. Якщо інженерна станція, операторська та емулятор сформовані на од-

ному ПК, то у них у всіх повинен бути однаковий IP-адрес (наприклад, кільцевий – 127.0.0.1). Відмінність буде лише в номері ID-ресурсу.

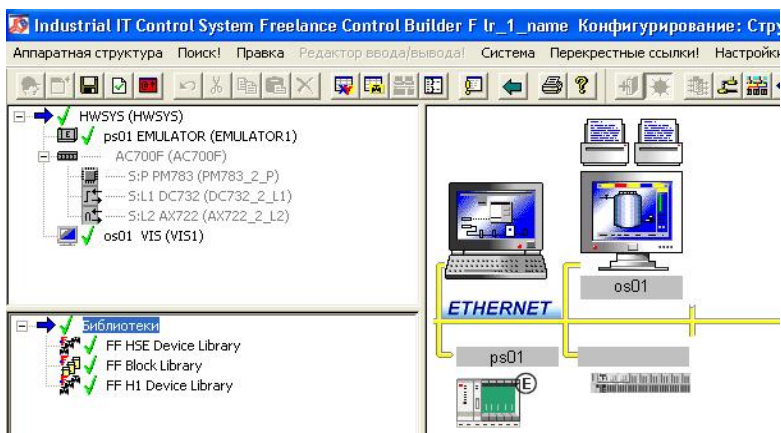


Рис. 1.23 – Структура апаратної частини проекту.

Конфигурация сети

Тип	Имя	Тип ресурса	Имя ресурса	ID ресурса	IP-адрес 1	IP-адрес 2
Инжен. ПК	CBF FF	D-ES		21	127.0.0.1	
VIS	VIS1	D-OS	os01	22	127.0.0.1	
AC700F	AC700F			2	172.16.1.2	
EMULATOR	EMULATOR1	D-PS	ps01	1	127.0.0.1	

Редактировать    ОК    Отмена

Рис. 1.24 – Вікно конфігурування мережі проекту.

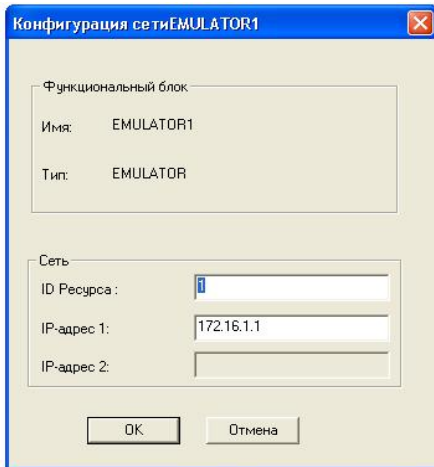


Рис. 1.25 – Вікно налаштування мережного ресурсу проекту.

Проведіть налаштування мережних параметрів інших ресурсів проекту відповідно до рис. 1.24.

Для зміни параметрів інженерної станції до складу ПТК *Freelance* входить окремий програмний модуль з ім'ям *Configure* та виконавчим файлом *FCNF32.EXE*, які знаходяться в папці «C:\Program Files\ABB Industrial IT\Freelance\exe\\*.».

Перевірте отриману мережну конфігурацію на правдоподібність та знову перейдіть до вікна з деревом проекту.

#### 1.4.5. Розроблення програми користувача

Після проведених дій можна розробляти програму користувача. Для цього ЛКМ двічі клікніть по рядку *ps01USRTask (FBD)* у вікні дерева проектів (див. рис. 1.22) для відкриття робочого простору для програмування. Відкриється вікно, екранна форма якого показана на рис. 1.26. В режимі розроблення програми користувача в рядку меню з'являться нові команди: *Программа FBD*, *Блоки*, *Элементы FBD*, *Перекрестные ссылки*. Контекстне меню складається з команд для створення зв'язку між компонентами, для вибору блоків та змінних.

Зверніть увагу, що редагування параметрів мережі можливо для всіх мережних ресурсів, крім інженерної станції. Можливо редагувати мережний ресурс, попередньо обрав ЛКМ рядок з його ім'ям і натиснувши на кнопку *Редактировать*. У вікні мережних налаштувань ресурсу можна поміняти *ID* або *IP*-адресу ресурсу. На рис. 1.25 показано вікно налаштувань мережних параметрів ресурсу *Emulator*.

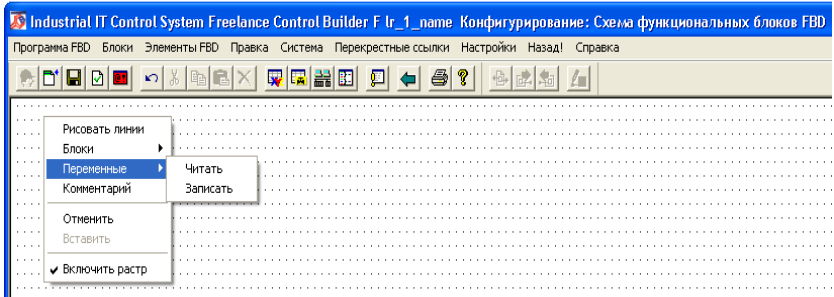


Рис. 1.26 – Робочий простір для створення програми користувача.

В середовищі *CBF* є можливість включення та виключення растра для більш точного позиціонування елементів програми користувача. Для більш докладного опису елементів меню *Блоки* використовуйте *Справку*. Змінні типу *Читать* є вхідними, а типу *Записать* – вихідними. Крім змінних, які пов'язані з фізичними входами та виходами в програмі можуть бути об'явлені локальні змінні, які відносяться лише до програми проекту. Для повернення в дерево конфігурації проекту використовуйте кнопку з зеленою стрілкою або команду *Назад!*.

Почніть розроблення програми користувача зі створення вхідних змінних. Перейдіть в режим вставки елементів в програму. Для цього, за допомогою контекстного меню виберіть команду *Читать*. Далі позиціонуйте курсор в потрібне місце робочого простору та ЛКМ клікніть потрібну кількість разів для вставлення необхідної кількості вхідних змінних. Вхідна змінна представляє собою рамку з виходом праворуч. Активність режиму «вставки» відображається курсором у вигляді кута в лівій верхній частині рамки елемента. Повернення в режим «вибору елемента» за допомогою ПКМ. Аналогічним чином створіть елементи логічної операції *AND* та вихідну змінну, але при цьому використайте команду *Записать*. Заповніть порожні рамки змінних символічними іменами. Для цього подвійним кліком ЛКМ відкрийте вікно конфігурування змінної, яке показане на рис. 1.27.

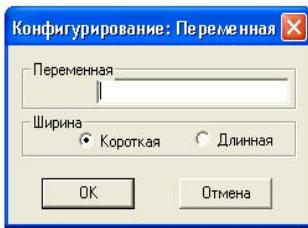


Рис. 1.27 – Вікно конфігурування змінної.

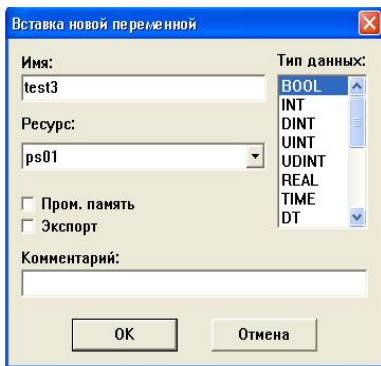


Рис. 1.28 – Вікно вставки нової змінної.

можливості автоз'єднувача, якщо додатково натиснути клавішу *Shift*, при цьому курсор перетвориться в лупу. Об'єднання сигналів позначається чорною точкою. Для інвертування сигналу сумісно з клавішею *Ctrl* клікніть ЛКМ по входу або виходу. Інвертований сигнал позначається чорною точкою.



Рис. 1.29 – Фрагмент програми користувача (1).

Наступне вікно, яке показано на рис. 1.28, додасть змінну до бази даних. В цьому вікні вказується ресурс, до якого відноситься змінна та тип даних. Крім того, можна поставити маркери в поля *Пром. пам'ять* та *Експорт*, а також коментар. Завершіть вставку змінної натисканням ЛКМ на кнопку *OK*. З'єднайте всі вставлені елементи в схему функціональних блоків, яка показана на рис. 1.29.

Лінії з'єднання змінних та блоків створюються за допомогою кліка ЛКМ з одночасним натисканням клавіші *Ctrl* шляхом проведення курсору в потрібному напрямку. Для зміни напрямку лінії відпустіть ЛКМ та клікніть для зміни напрямку. Використовуйте функціональні

Додайте в програму дві схеми для порівняння двох змінних типу *REAL*. Результатом операції порівняння буде змінна типу *BOOL*. На рис. 1.30 показаний фрагмент програми користувача. Як видно з рис. 1.29 та 1.30 лінії з'єднання залежать від типу сигналу: для змінних типу *BOOL* лінія тонше, а для змінних типу *REAL* – товще. Крім того, кольором помічається лінія, яка вказує на тип змінної, значення якої передається.

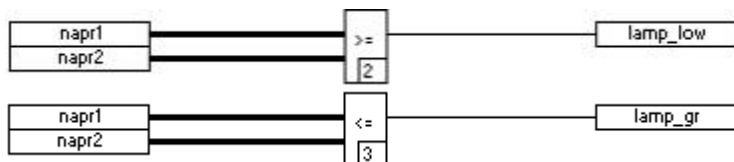


Рис. 1.30 – Фрагмент програми користувача (2).

Переміщення блоків без розірвання зв'язків можливе, якщо виділити ЛКМ блок, потім одночасно натиснути на кнопки Ctrl та Shift і ЛКМ на виділеному блоці і позиціонувати в нове місце робочого простору.

Проведіть перевірку на правдоподібність розроблену програму користувача та поверніться до дерева проекту.

#### 1.4.6. Завантаження та налагодження програми користувача

Проведіть повну перевірку проекту. Для цього клікніть ЛКМ на корінь в дереві проекту. Якщо немає помилок, то усі елементи дерева проекту будуть позначені в заголовку прямокутниками зеленого кольору.

Активуйте емулятор ПЛК за допомогою браузера *Internet Explorer*, а в ньому сайт *Freelance Controller Emulator Web*, який запитується за адресою <http://127.0.0.1:8888>. На сайті емулятора ПЛК (див. рис.1.14) в поле *Station ID* вставте номер ресурсу, наприклад «1» та натисніть ЛКМ на кнопку *Start Controller*.

Далі натисніть ЛКМ на кнопку *Наладка*. У вікні, яке зображено на рис. 1.31, вкажіть курсором на елемент дерева *ps01 (Emulator)* і через контекстне меню за допомогою команди *Завантажить*→*Вся станція* завантажте проект до емулятору ПЛК.

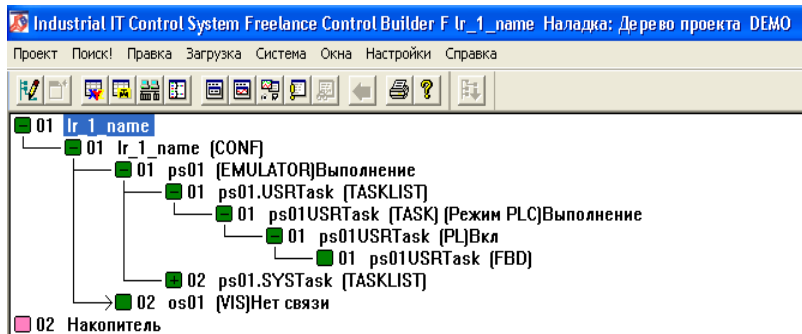


Рис. 1.31 – Вікно налагодження проекту.

Подвійним кліком ЛКМ по рядку *ps01USRTask (FBD)* активуйте вікно з програмою. У вікні, яке зображено на рис. 1.32 представлений фрагмент запущеної програми. Тут змінна *napr1* має більше значення, ніж змінна *napr2*. Тому змінна *lamp\_low* знаходиться в стані *TRUE*. Змінна *lamp\_alarm* також знаходиться в стані *TRUE*, тому що змінні *test1* і *test2* знаходяться одночасно в стані *TRUE*. При цьому одиничне значення змінної позначається за допомогою суцільної лінії, а нульове – пунктирною лінією.

Для зміни значення змінних досить підвести до них курсор та ПКМ викликати контекстне меню. Потім вибрати команду *Записати значення...*, а в полі введення, ввести нове значення та натиснути ЛКМ на кнопку *OK*, як це показано на рис. 1.33.

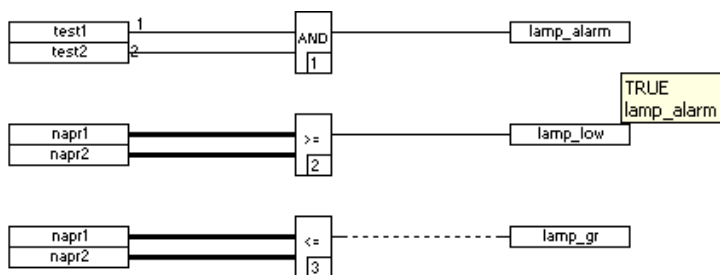


Рис. 1.32 – Фрагмент програми користувача, яка виконується.

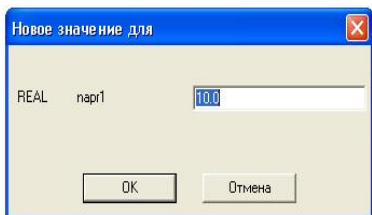



Рис. 1.33 – Вікно введення нового значення.

Для запуску ЛМІ проекту необхідно спочатку додати його до дерева проекту операторської станції *os01 (VIS)*. Далі запустити файл *BUBMAIN.EXE*, який запускає програму *DigiVis* та знаходиться в папці зі шляхом «*C:\Program Files\ABB Industrial IT\Freelance\exe\\*.\**». Запустити візуалізацію можливо також за

допомогою кліка ЛКМ по ярлику програми  на робочому столі *Windows* або за допомогою кнопки *Пуск* системного меню послідовно натискаючи ЛКМ на рядки меню: *Все программы*→*ABB Industrial IT*→*Freelance V9.2*→*DigiVis*.

## РОЗДІЛ 2

### ПРАКТИКУМ З РОЗРОБЛЕННЯ ППЗ ТА ЛМІ РСУ НА ОСНОВІ ПТК *FREELANCE 700F*

#### 2.1 Розроблення ППЗ для системи управління дискретним об'єктом

Метою цього практичного завдання є розроблення ППЗ для управління технологічним устаткуванням, наприклад, приводом виконавчого механізму. Більш детально умови завдання будуть пояснені далі, в процесі розроблення ППЗ.

Отже запустить середовище *CBF* та створить новий проект, як це описано в п.1.4.1 та п.1.4.3. В результаті отримаєте проект з двома вузлами: проектом та накопичувачем. За допомогою контекстного меню, яке викликається ПКМ створить конфігурацію проекту (*CONF*). Далі, для прискорення процесу ініціювання нового проекту додайте до накопичувача шаблон *ProjectTree.prt* та перенесіть компоненти шаблону до конфігурації вузла проекту. В результаті отримаєте дерево проекту, яке зображене на рис. 2.1. В цьому дереві імпортовано два вузли: процесова та операторська станції. В процесній станції присутні два списки задач: одна задача користувача з циклом виконання *500мс* та сім системних задач. На даному етапі системні задачі не потрібні. Тому згорніть їх, натиснувши ЛКМ на символ «-» напроти відповідного рядку.

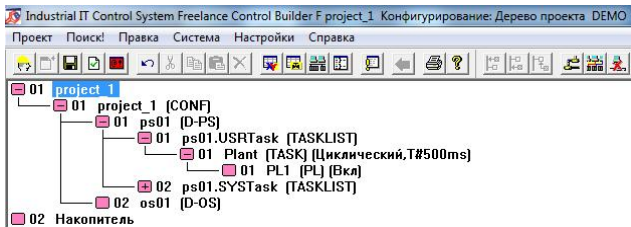


Рис. 2.1 – Структура проекту для виконання завдання.

На рис. 2.1 задача користувача вміщує список програм, який є її структурним елементом. В цей список в подальшому будуть додаватися різноманітні програмні компоненти, наприклад програми та функціональні блоки. Додайте до списку компонент нижнього рівня *PL1 (PL)* – програму з ім'ям *NP10* та мовою реалізації *FBD*. Подвійний клік ЛКМ відкриває компонент *NP10*.

За допомогою команди меню *Блоки* активуйте ЛКМ список *Разомкнутое управление*. В цьому списку виберіть ФБ з ім'ям «*БУ приводом механизма (IDF1)*». При цьому блок з курсором в лівому верхньому куті з'явиться в робочій області та може бути переміщеним за допомогою миши в будь-яке місце, поки не клікнути ЛКМ в потрібному місці. Далі активується другий екземпляр блоку з активним курсором для вставлення блоку. Якщо другий блок не потрібен необхідно натиснути ПКМ. Для зміни розміщення блоку потрібно його виділити ЛКМ і утримуючі ЛКМ перемістити в нове місце робочої області.

Відкрийте подвійним кліком ЛКМ блок для параметрування. Відкривається бланк, який зображений на рис. 2.2.

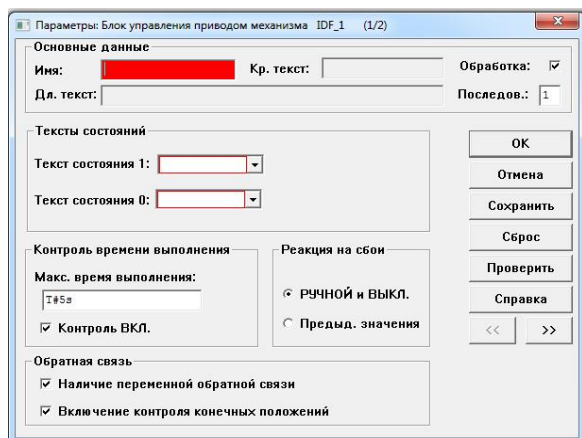


Рис. 2.2 – Вікно параметрування блоку «*БУ приводом механизма (IDF\_1)*» (перше вкладення).

Бланк параметрування блоку складається з двох вкладень. Переключення між ними здійснюється натисненням ЛКМ на кнопки «>>» та «<<». На рис. 2.3 зображено друге вкладення бланку параметрування блоку управління приводом механізму *IDF\_1*.

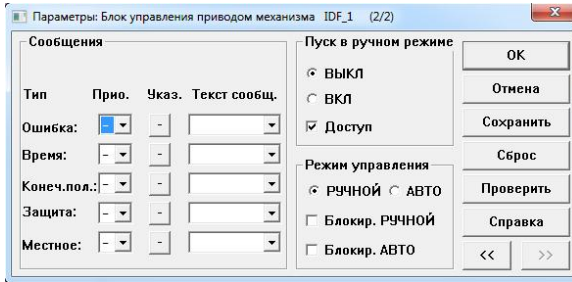




Рис. 2.3 – Вікно параметрування блоку «БУ приводом механізму (*IDF1*)» (друге вкладення).

Відмітимо, що заповнення полів, які позначені червоним кольором обов'язкові. Введіть в першому вкладенні бланка ім'я блоку – *NP10*, а в зоні «Тексты состояний» у відповідні поля – «ON» та «OFF». Збережіть введену інформацію натиснувши на кнопку «OK». Далі необхідно провести лінії, які з'єднують вихід блоку «OUT» з його входами «FBO» та «FBI», причому «FBO» у інверсному стані. Для цього потрібно натиснути одночасно ЛКМ та клавішу «Ctrl». Для зміни напрямку лінії необхідно відпустити ЛКМ та знову натиснути і провести лінію в новому напрямку. Для креслення зв'язків у режимі авто побудови необхідно додатково натиснути на клавішу «Shift». Для інвертування сигналу натисніть на ввід або вивід блоку утримуючи клавішу «Ctrl». Інвертований сигнал позначається чорною крапкою на вході або виході блоку. Взагалі немає сенсу створювати такий зворотній зв'язок в реальних процесах. Тобто, використовують реальні сигнали від приладів, які встановлені на технологічних установках. В даному випадку таке з'єднання дозволить імітувати роботу функціонального блоку.

Перевірте отриману схему на правдоподібність. Якщо отримаєте інформаційне вікно про відсутність помилок, закрийте це вікно та завершить режим редагування блоку, тобто поверніться до дерева проекту. Для цього натисніть на кнопку з зеленою стрілкою «Назад!». На запит про збереження змін натисніть ЛКМ на кнопку «Да». Далі ЛКМ виберіть вузол *project\_1* та повторіть перевірку на правдоподібність змісту усього вузла. Як і раніше (див. п. 1.4.2) з'явиться інформаційне вікно з повідомленням про помилки (див. рис. 1.17). Дані помилки є наслідком того, що в проекті не зроблено розподіл ресурсів для операторської та процесової станцій. На даний час вони лише частина дерева проекту. Тому відкрийте апаратну структуру проекту. Для цього натисніть на кнопку структури апаратних засобів . В новому вікні буде зображена лише інженерна станція. Далі дійте до інструкцій, що викладені у п.1.4.2 для отримання апаратної структури, подібної до зображеної на рис. 1.16. (за виключенням фізичного контролеру). Призначити апаратні ресурси процесової та інженерної станції. Поверніться до дерева проекту зі збереженням налаштувань.

Далі потрібно вказати IP-адреси для емулятора ПЛК та операторської станції *DigVis*. Для цього в головному меню ЛКМ клікніть по списку команд *Проект* і далі по команді *Сеть...*. Відкриється вікно мережних налаштувань. Вкажіть спеціальну IP-адресу «127.0.0.1» та ID для усіх ресурсів в серієвці *CBF*. Вікно налаштувань буде відповідати зображеному на рис. 2.4. Поверніться до дерева проекту та перевірте проект на правдоподібність. Система повідомить про відсутність помилок.

Отже проект готовий до завантаження до емулятора та операторської станції. Натисніть ЛКМ на кнопку  для налагодження проекту в емуляторі та перевірки роботи операторської станції. Дерево проектів перемкнеться в цей режим. Напроти ресурсів буде вказано про відсутність відповідних апаратних ресурсів, тобто про те, що вони не на зв'язку.

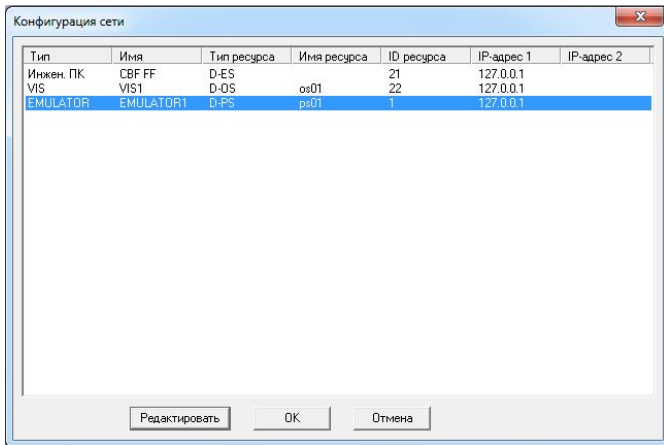


Рис. 2.4 – Вікно мережних налаштувань ресурсів проекту.

Таким чином потрібно запустити емулятор ПЛК та *DigVis*, як це показано в першому розділі. Отже запустить емулятор ПЛК. Для цього в будь-якому браузері відкрийте сторінку емулятора. Вкажіть на сторінці емулятора у полі *Station ID* номер *ID* апаратного ресурсу. Це буде цифра «1». Натисніть ЛКМ на кнопку *Start Controller*. Поверніться до середовища програмування *CBF*. Якщо середовище буде в режимі налагодження, то в дереві проекту буде показано, що процесова станція працює, але з невірною версією проекту. Тому потрібно завантажити до неї актуальний проект. Але, раніше за допомогою стандартних дій в меню *ПУСК* операційної системи *Windows* запустить операторську станцію *DigVis*. Відкриється системний кадр *DigVis*.

Далі завантажимо процесову станцію до емулятора. Для цього в основному меню виберіть список *Завгрузка*, а в ньому – команду *Вся станция*. На запит про підтвердження натисніть ЛКМ на кнопку *Да*. Після цього буде завантажено актуальний проект до емулятора. Подібним чином завантажить операторську станцію *DigVis*.

Після завершення завантаження відкрийте вікно *DigVis*. В меню *Кадри* виберіть команду *Лицевая панель* для відкриття переліку лицевих панелей.

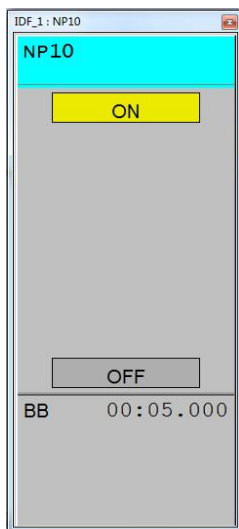


Рис. 2.5 – Лицева панель *NP10*.

В цьому переліку буде усього один компонент з ім'ям *NP10*. Подвійним кліком ЛКМ або натиском на кнопку *OK* відкрийте цю панель. Її зображення показано на рис. 2.5. В нижній частині системного кадру *DigVis* будуть доступні інтуїтивно зрозумілі кнопки *Управл.* та *Руч/Авт* для керування приводом механізму. За допомогою цих кнопок можливе вмикання або вимикання приводу та управління режимом роботи приводу: ручним або автоматичним. Усі дії оператора потребують підтвердження шляхом натиску ЛКМ на кнопку *Подтв.* Завершення дій щодо керування приводом визначається натиском ЛКМ на кнопку *Закреть*.

Відмітимо, що на панелі, крім стану приводу, є цифри, які визначають час перемикання приводу. У даному випадку цей час за умовчанням дорівнює *5s*. Якщо привід запущений, то лінія, яка з'єднує вихід зі входом блоку буде безперервною. В протилежному випадку – штриховою.

Перейдіть до середовища програмування *CBF* у вікно блока керування приводом. В системі *CBF* блок має узагальнений термін – тег. Подвійним кліком ЛКМ відкрийте цей тег та перейдіть у друге вкладення. В цьому вкладенні також можливе керування приводом шляхом заповнення відповідних полів у вкладенні. Одночасно можна спостерігати за лицевою панеллю тега. Якщо активовано команду вимикання приводу, то з'єднувальна лінія стане штриховою, а лицева панель оновиться. Такий випадок показаний на рис. 2.6. Сам процес вимикання приводу позначений стрілкою жовтого кольору, яка направлена до індикатора *OFF*.

На цьому можна завершити розгляд першого прикладу практикуму. Підсумуємо результати виконання цього прикладу:

- в конфігурації нового проекту було створено процесову та операторську станцію, а також програму користувача на мові функціональних блоків;
- було проведено апаратне конфігурування проекту з підключенням ресурсів проекту;
- було налаштовано мережні параметри ресурсів проекту та їхні ідентифікатори;
- було запущено емулятор ПЛК та *DigVis* та завантажено до них відповідні станції;
- було проведено налагодження програми користувача в *DigVis* шляхом використання лицевої панелі та в *CBF* шляхом використання бланка налаштування тега.

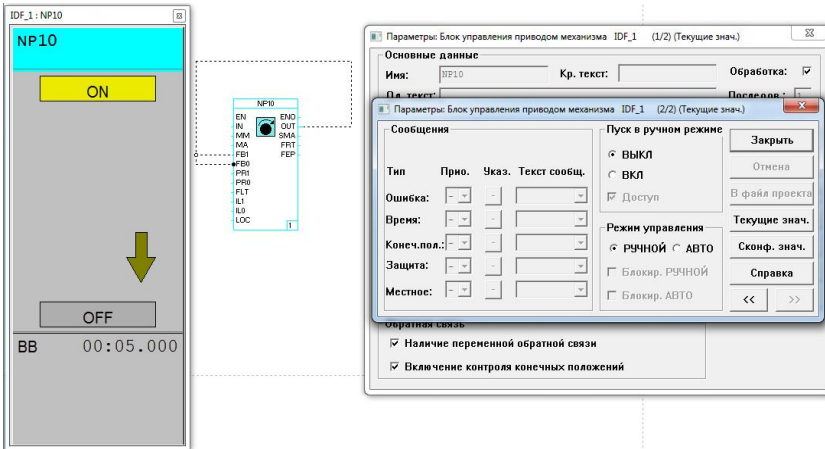


Рис. 2.6 – Вікно з тегом, лицевою панеллю та бланком параметрування. (процес перемикання стану приводу механізму)

### Завдання для самостійного виконання

1. Розробіть програму користувача, яка реалізує дискретний автомат для управління вихідними пристроями. Умовою включення або виключення вихідних пристроїв будуть логічні вирази, які наведені в табл. 2.1. В цій таблиці логічним змінним  $X1...X8$  відповідають перемикачі  $S1...S8$  (див. рис. 1.3) на входах модуля  $DC732$ , а результат логічної операції  $Y1...Y8$  присвоюється відповідним виходам  $L1...L8$  (див. рис. 1.3). Складіть таблицю істинності для логічної умови.

Таблиця 2.1 – Варіанти умов завдань для самостійного виконання

№ з/п	Логічні умови для входів та виходи	
	$X1...X8$ – входи	$Y1...Y8$ – виходи
1	$(X1 \text{ AND } X3) \text{ OR } (X4 \text{ AND } X5 \text{ AND } \text{NOT } X8)$	$Y1$
2	$(X2 \text{ OR } X3) \text{ AND } (\text{NOT } X1 \text{ AND } X5 \text{ AND } X7)$	$Y2$
3	$(X1 \text{ AND } X2) \text{ OR } X4 \text{ AND } \text{NOT } X6 \text{ AND } X8)$	$Y3$
4	$(X2 \text{ AND } X3) \text{ XOR } (X5 \text{ AND } X6 \text{ AND } X7)$	$Y4$
5	$(\text{NOT } X1 \text{ XOR } X3) \text{ AND } (X5 \text{ OR } X7 \text{ OR } X8)$	$Y5$
6	$(X2 \text{ AND } \text{NOT } X3) \text{ OR } (X4 \text{ AND } X5 \text{ AND } X8)$	$Y6$
7	$X2 \text{ AND } X3 \text{ AND } \text{NOT } X4 \text{ AND } (X1 \text{ XOR } X7)$	$Y7$
8	$(X1 \text{ OR } X3 \text{ OR } X4) \text{ AND } (\text{NOT } X7 \text{ OR } X8)$	$Y8$

2. Розробіть програму користувача для керування електричним бойлером та витяжним пристроєм. Вмикання та вимикання вихідних пристроїв залежить від стану дискретних датчиків та кнопок (див. рис. 1.3). Умови будуть такими:

- вмикання сигналізації ( $L9$ ) при виниканні будь-якої аварії на об'єкті ( $S3$  – пожежа,  $S4$  – аварія бойлера);
- вимикання бойлера ( $L1$ ) при виниканні будь-якої аварії на об'єкті;
- вмикання витяжного пристрою ( $L2$ ) у випадку пожежі;
- вимикання витяжного пристрою ( $L2$ ), якщо вимкнеться датчик пожежі;

- вмикання бойлера за допомогою кнопки (S1) за умови відсутності аварії;

- вимикання бойлера за допомогою кнопки (S2).

### Контрольні запитання

- 1) Надайте коротку характеристику ПТК *Freelance* 800.
- 2) Яка архітектура побудови ПТК *Freelance* 800?
- 3) Який порядок створення, конфігурування та налагодження проекту в середовищі *CBF*?
- 4) Порядок налаштування апаратних ресурсів.
- 5) Мережні налаштування ресурсів проекту в середовищі *CBF*.
- 6) Робота середовища *CBF* в режимі моніторингу та відображення процесу.
- 7) Як завантажити проект в ПЛК (емулятор)?

## 2.2. Розроблення ППЗ для системи управління установкою водопостачання

Розглянемо технологічний об'єкт – установку водопостачання. Її мнемосхема зображена на рис. 2.7. До складу установки входять насос та клапан. На трубопроводі встановлені вимірювачі тиску та витрати.

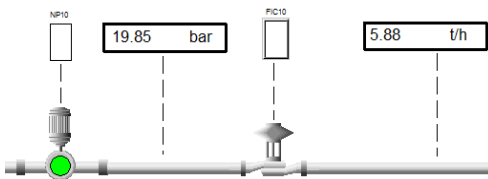


Рис. 2.7 – Мнемосхема системи водопостачання

Насос після вмикання створює у трубопроводі тиск з максимальним значенням до 20 бар. Клапан забезпечує регулювання витрати відповідно до завдання.

Отже запустить середовище *CBF* та створить новий проект, як це описано в п.1.4.1...1.4.3 та в попередньому підрозділі. Імпортуйте в накопичувач наступні компоненти:

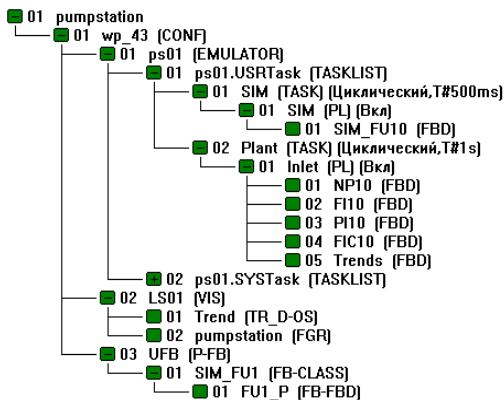


Рис. 2.8 – Дерево проекту системи водопостачання

тувача. В процесній станції створено два завдання: одне (*SIM*) для імітування роботи установки з циклом *500ms*, інше (*Plant*) – система керування установкою з циклом *1s*. Операторська станція буде відображати мнемосхему установки та тренди.

Далі заповніть програми *FU10*, *NP10*, *FIC10*, *PI10*, *FI10* та *TRENDS*, які входять до списків завдань. Програми симулятора, програми управління насосом, програми розрахунку витрати та тиску, регулятор та програма відображення трендів показані на рис. 2.9...2.14.

Надамо пояснення щодо заповнення програм, які зображені на рис. 2.9...2.14. Почнемо з рис. 2.9, на якому зображена програма симулятора. Якщо подвійним кліком ЛКМ відкрити програму *FU10* для редагування, то вона вочевидь буде порожньою. Додамо до програми функціональний блок стимулятора установки. Це буде блок користувача, який раніше був імпортований до накопичувача. Додавання блока в програму здійснюється шляхом виклику зі списку меню *Блоки* команди *Польз. функциональные блоки...*. З'явиться вікно зі списком доступних ФБ користувача. У даному випадку це буде *SIM\_FU1*. Виберіть блок стимуля-

*PUMPSTATION.PRT*,  
*SIM\_FU1.PRT* та  
*SIM\_FU10.PRT*. Це, поперше, мнемосхема установки, по-друге – блок симулятора роботи установки та блок керування насосом. В результаті отримайте дерево проекту, яке зображене на рис. 2.8. В цьому дереві присутні два вузли та функціональний блок користувача.

тора натисненням ПКМ та підтвердить вибір натисненням ПКМ на кнопку *OK*. В робочому просторі з'явиться блок з активованим курсором, як це було раніше, при вставленні звичайних блоків. Переміщенням курсору з блоком встановить блок в потрібне місто, клікніть ЛКМ для вставлення для фіксації блоку та ПКМ для виходу з режиму вставлення. Далі потрібно з'єднати потрібні входи та виходи ФБ зі змінними, як це зображено на рис. 2.9. Тут вхідні змінні (тобто змінні з функцією читання) отримують значення від інших блоків в програмах.

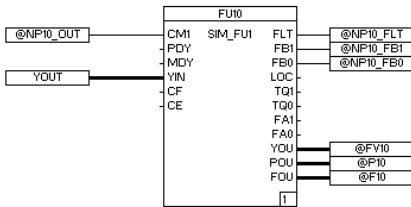


Рис. 2.9 – Програма стимулятора системи водопостачання.

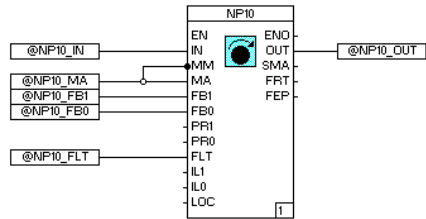


Рис. 2.10 – Програма керування насосом.

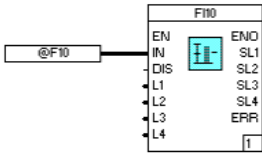


Рис. 2.11 – Програма розрахунку витрати.

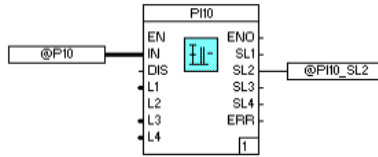


Рис. 2.12 – Програма розрахунку тиску.

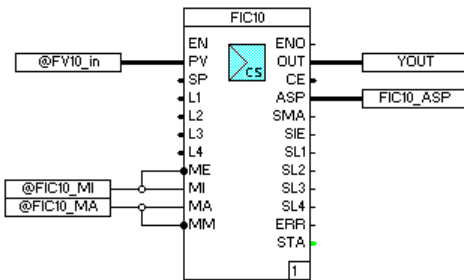


Рис. 2.13 – Програма регулювання витрати.

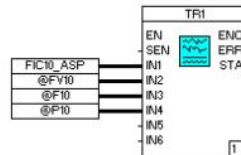


Рис. 2.14 – Програма відображення трендів.

Для визначення джерела в *CBF* є потужний інструмент. Це перевірка на *перехресні зв'язки*. Наприклад, для визначення джерела змінної *@NP10\_out* потрібно ЛКМ виділити її та за допомогою ПКМ викликати контекстне меню, потім вибрати команду *Перекрестные ссылки*. З'явиться вікно, яке зображено на рис. 2.15, де показано зв'язки змінної з іншими ФБ та програмами з позначенням режимів доступу: читання та запису.

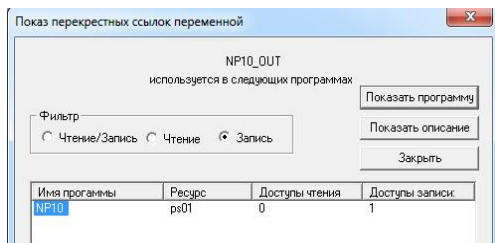


Рис. 2.15 – Вікно з перехресними зв'язками.

На рис. 2.15 вікно інформує, що змінна *@NP10\_out* в режимі запису отримує значення з виходу програми *NP10*. Якщо клікнути ЛКМ на кнопку *Показать программу*, то спочатку відкриється вікно із запитом на збереження активної програми (у випадку поточних змін в програмі). Далі натиснення ЛКМ на кнопку *Да* автоматично відкриє програму-джерело. Додатково зазначимо, що наявність в імені змінної символу «@» вказує на місце зберігання значення змінної. Це так звана *проміжна пам'ять*. Рекомендовано використовувати цю пам'ять для роботи з симулятором. Відсутність символу вказує на основну пам'ять, тобто глобальну.

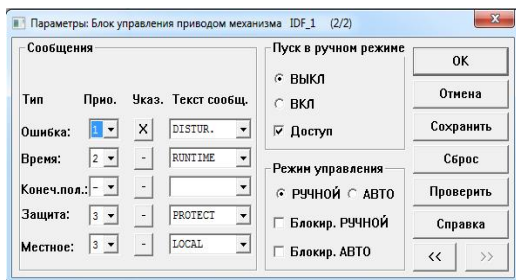


Рис. 2.16 – Вікно з налаштуванням блоку *IDF\_1* (друге вкладення).

Програма керування насосом *NP10* реалізована за допомогою вже відомого блоку керування приводом механізму *IDF\_1*. Налаштування блоку на першому вкладенні подібні тим, що наведені в попередньому підрозділі. На рис. 2.16

показані налаштування блоку на другому вкладенні.

Програма розрахунку витрати та тиску (*FII0*, *PII0*) реалізовані за допомогою блока *Контроль аналогового сигнала M\_ANA*. В налаштуваннях блоку потрібно встановити параметри шкали вимірювання, одиницю виміру та гістерезис. Для витрати це відповідно: шкала – від 0 до 10 м<sup>3</sup>/ч, гістерезис – 3 м<sup>3</sup>/ч. Тиск вимірюється в діапазоні від 0 до 20 бар з гістерезисом 3 бар та контролем за значенням тиску на рівні 15 бар з формуванням повідомлення про досягнення максимального значення. Налаштування блоку вимірювання тиску показані на рис. 2.17.

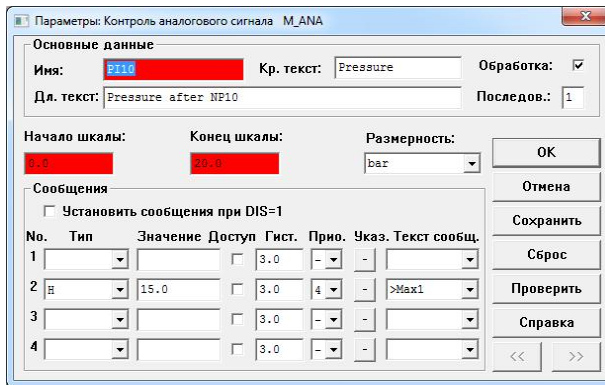


Рис. 2.17 – Вікно з налаштуванням блоку *M\_ANA*.

Далі розглянемо налаштування регулятора витрати, який реалізований в програмі *FIC10*, яка зображена на рис. 2.13. В програмі використаний ФБ *Непрерывный стандартный регулятор C\_CS*. Приклад налаштувань ФБ *C\_CS* зображені на рис. 2.18...2.20 з трьома вкладеннями.

Насамкінець розглянемо ФБ *Сбор трендов* з програми *TRENDS*. Він не потребує ніяких налаштувань. Відповідно до рис. 2.14 в програмі будуть зображені графіки зміни тиску, витрати, положення клапана та вихідне значення регулятора *FIC10*. Зауважимо, що налаштування трендів робиться в вузлі *TREND (TR\_D-OS)* операторської станції *LS\_01(VIS)*.

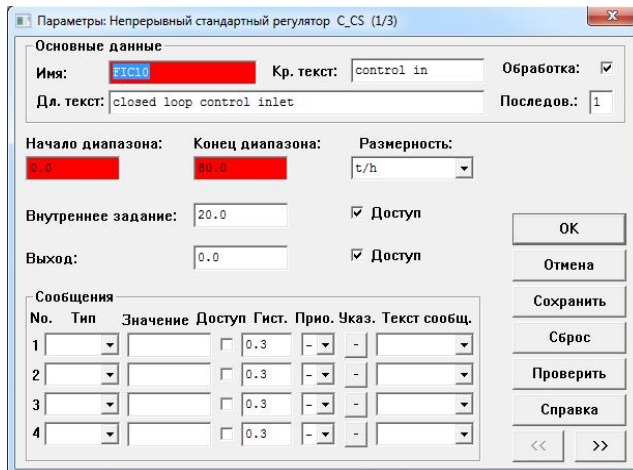


Рис. 2.18 – Вікно з налаштуванням блоку C\_CS (1/3).

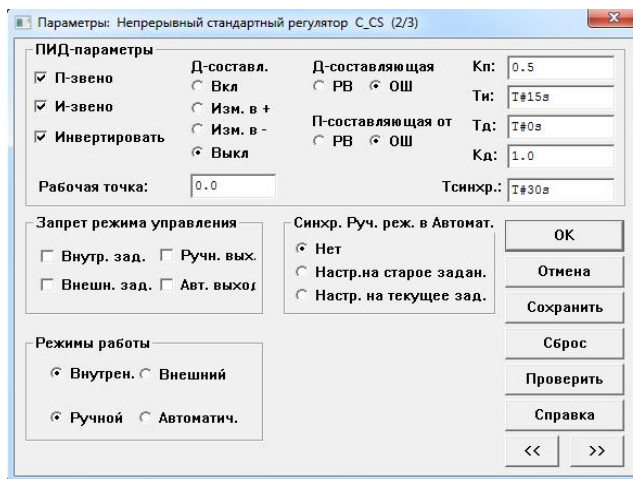


Рис. 2.19 – Вікно з налаштуванням блоку C\_CS (2/3).

Далі проведемо налаштування операторської станції, в якій створені два ресурси: *Trend* та *Pumpstation*. Вікно налаштувань кадру трендів на вкладенні *Кадр* показано на рис.2.21. Во вкладенні *Сбор трендов*

необхідно встановити флаг джерела даних. Це буде ФБ *Сбор трендов* з програми *TRENDS* з процесової станції (див.рис.2.14). Однак в *CBF* є інша можливість формування трендів. Це використання сервера трендів, який є окремим ресурсом у складі конфігурації проекту.

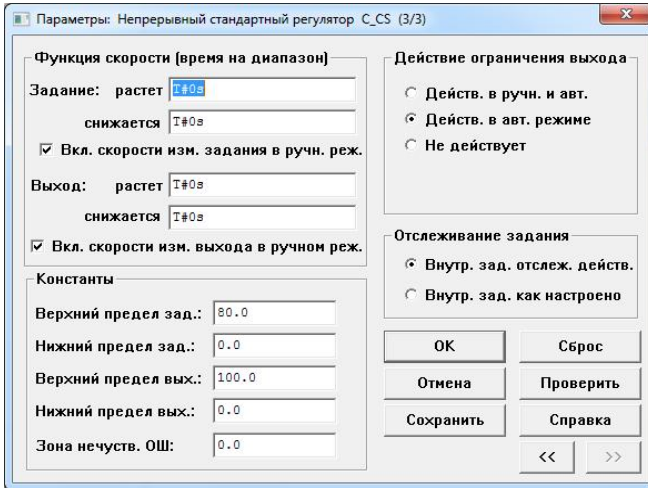


Рис. 2.20 – Вікно з налаштуванням блоку *C\_CS* (3/3).

Мнемосхема була імпортована до накопичувача та переміщена до кадру дисплея. Однак вона є статичною. Тому необхідно до неї додати динамічні елементи. Це будуть індикатори технологічних параметрів (тиску і витрати) та стану пристроїв: насоса і клапана. Так, наприклад стан насоса відображається кольором елемента «коло» всередині статичного елемента насос. Отже, в режимі конфігурування подвійним кліком ЛКМ відкрийте кадр з мнемосхемою *Pumpstation (FGR)* в операторській станції *LS\_01(VIS)* на редагування. Відкриється мнемосхема, яка показана на рис. 2.7, але в режимі редагування з активним маркером, який управляється мишею. ЛКМ виділить графічний елемент «коло» та подвійним кліком ЛКМ відкрийте вікно його налаштувань. Це вікно показано на рис. 2.22.

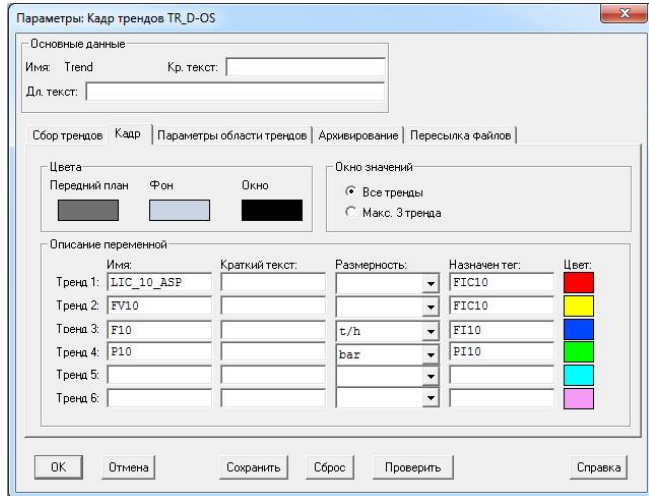


Рис. 2.21 – Вікно з налаштуванням вузла *TREND (TR\_D-OS)*.

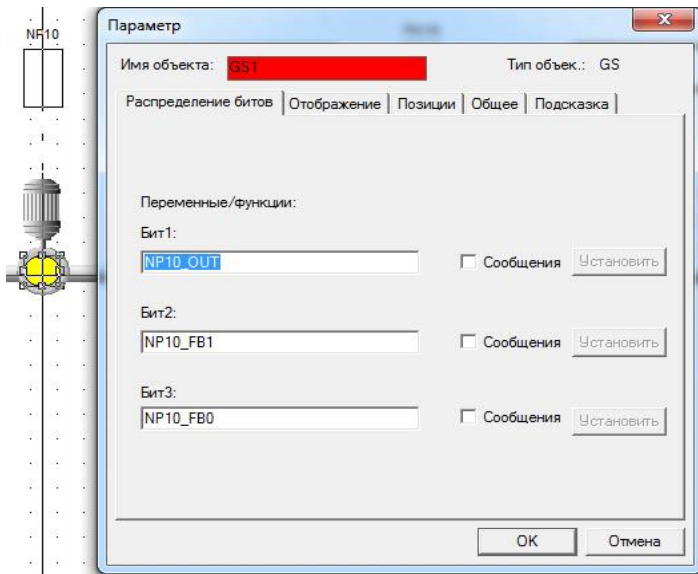


Рис. 2.22 – Вікно з налаштуванням елемента «коло».

Кольори елемента визначаються станом змінних, якими заповнені поля бітів. Це стан змінних *NP10\_OUT*, *NP10\_FBI* та *NP10\_FB1*, які належить тегу *NP10*. Відображення кольором налаштовується в вкладенні *Отображение*. Для виклику лицевих панелей використовують графічний елемент *Выбор*. В налаштуваннях елемента во вкладенні *Общее* ЛКМ натисніть на кнопку *Действия* і далі вкажіть потрібний тег. На рис.2.23 показані налаштування графічний елемент *Выбор*. На цьому рисунку викликається лицева панель тега *FIC10*.

Числове значення технологічних параметрів відображається за допомогою графічного елемента *Буквенно-цифровой дисплей*. Порядок налаштувань цього елемента подібний до налаштувань елемента *Выбор*.

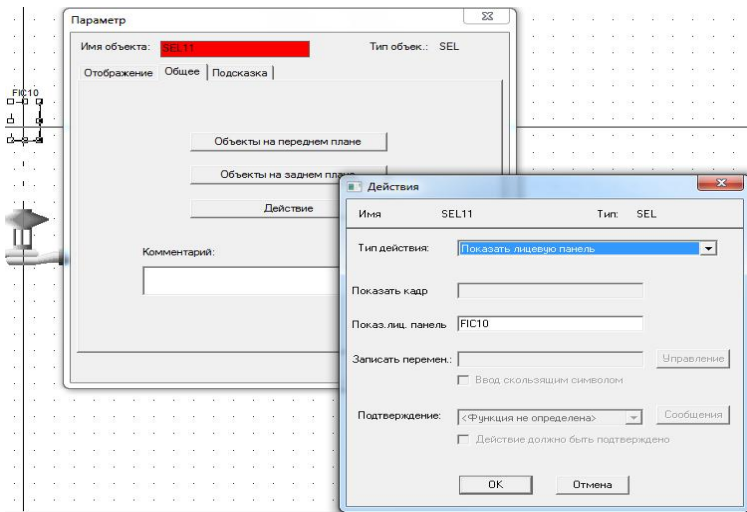


Рис. 2.23 – Вікно з налаштуванням елемента *Select*.

На завершення завантажте ресурси проекту до емулятору і операторської станції та перевірте роботу розробленої системи управління установкою водопостачання.

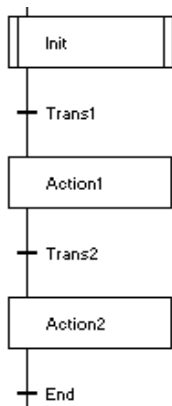
### Контрольні запитання

- 1) Яким чином в дереві проекту створюються нові об'єкти?
- 2) Як імпортувати до проекту шаблони?
- 3) Порядок копіювання та переміщення вузлів в дереві проекту?
- 4) Як додавати та налаштовувати вільну графіку до операторської станції?
- 5) Яким чином додавати та налаштовувати графічні елементи до мнемосхеми проекту?

### **2.3. Розроблення програми послідовного керування установкою водопостачання на мові SFC**

В *CBF* існує можливість створення програм для послідовного управління. Такі програми розробляються на мові *SFC* та мають назву програма логічного управління (*ПЛУ*). Додамо до проекту управління установкою водопостачання, яка розроблена в попередньому підрозділі, програму послідовного управління на мові *SFC*. Така програма потрібна для першого запуску установки та керування установкою в подальшому для автоматизації процедур, що повторюються. Тобто ця програма дозволяє керувати дискретно-безперервними технологічними процесами, наприклад, періодичним заповненням ємності, перемішуванням, видержуванням у часі та випорожненням. Для попереднього прикладу буде реалізоване ПЛУ установкою водопостачання, яка вмикається після перерви.

Спочатку завантажте проект, який був розроблений раніше. Це проект, дерево якого показано на рис. 2.8. За допомогою контекстного меню додайте в задачу користувача *Plant* нову програму з ім'ям *sfc* на мові *SFC*. Відкрийте цю програму на редагування. З'явиться вікно з робочим простором, поділеним на комірки вертикальними та горизонтальними лініями. У цьому просторі будується структура програми *sfc*, яка складається з кроків та переходів. При цьому, крокі мають виконувати дії, а переходи мають умови. Ці елементи програми з'єднуються горизонтальними та вертикальними лініями. Програма обов'язково повинна



мати стартовий крок. Виберіть ЛКМ комірку та викличте ПКМ контекстне меню, в якому виберіть ЛКМ команду *Начальний шаг*. Кожному кроку відповідає перехід з умовою. Далі за допомогою контекстного меню додайте вертикальні лінії до переходу. Доданий перехід створює можливість додати новий крок, якщо виконаний попередній крок. Далі подібно додамо кроки та переходи. Одночасно вкажемо імена крокам та переходам. В результаті

Рис. 2.24 – *SFC*-програма. отримаємо структуру програми *sfc*, яка зображена на рис. 2.24. Поверніться до дерева проекту натиснувши ЛКМ на зелену стрілку та підтвердіть збереження введеної структури програми. Система *CBF* автоматично згенерує в дереві проекту програму з ім'ям *sfc(SFC)*. Дерево проекту приме вид, який наведений на рис. 2.25.

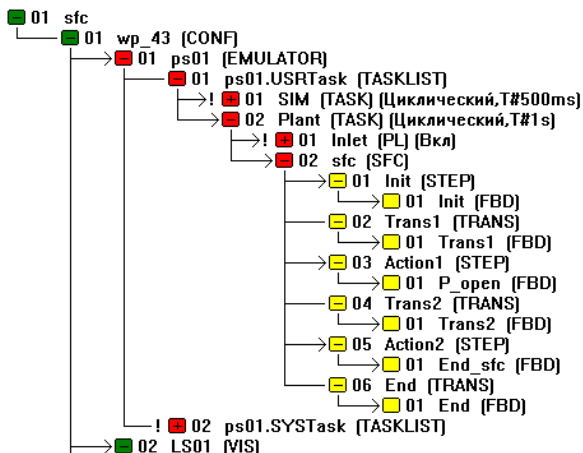


Рис. 2.25 – Дерево проекту з доданим новим елементом *sfc*.

Проте, в цій програмі не визначені дії кроків та умови переходів. Тому заповнимо структурні компоненти програми змістом, тобто кожному кроку та переходу призначимо власну програму. Мову реалізації дій та умов виберіть *FBD*.

Отже, для кроку *Init* створимо компонент – програму *Init*. Програма наведена на рис.2.26. Ця програма автоматично запускає насос (змінна *@NP10\_MA*) та регулятор витрати (змінна *@FIC10\_MA*), якщо витрата в стимуляторі (змінна *@F10*) дорівнює нулю.

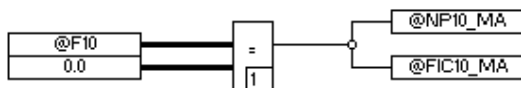


Рис. 2.26 – Програма кроку *Init*.

Програма з умовою переходу зображена на рис. 2.27. В умові переходу передається статус *OFF* змінної зворотного зв'язку *@NP10\_FB0* від програми-симулятора установки (див. рис. 2.9) в змінну-прапорець *.RESULT*.



Рис. 2.27 – Програма переходу *Trans1*.

На рис. 2.28 зображена програма кроку *Action1*, в якій в змінну *@NP10\_IN* записується значення *TRUE* з блоку константи *P\_open* типу *BOOL*.

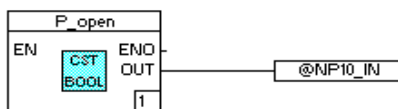


Рис. 2.28 – Програма кроку *Action1*.

Умовою переходу *Trans2* (див. рис. 2.29) є присвоєння статусу *ON* змінної зворотного зв'язку *@NP10\_FB1* від програми-симулятора установки (див. рис. 2.9) в змінну-прапорець *.RESULT*.



Рис. 2.29 – Програма переходу *Trans2*.

На рис. 2.30 зображена програма кроку *Action2*, в якій в змінну *@FIC10\_MA* записується значення *TRUE* з блоку константи *End\_sfc* типу *BOOL*.

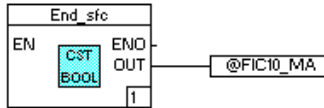


Рис. 2.30 – Програма кроку *Action2*.

Програма з умовою переходу *End* зображена на рис. 2.31. В умові переходу передається результат порівняння поточного значення витрати (змінна *@F10*) від програми-симулятора установки (див. рис. 2.9) з максимальним значенням 20.0 т/год. в змінну-прапорєць *.RESULT*.

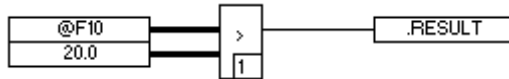


Рис. 2.31 – Програма переходу *End*.

Після заповнення структури ПЛУ перевірте весь проект на правдоподібність. Далі потрібно додати кадр для відображення ПЛУ в операторській станції *LS01(VIS)*. Подвійним кліком ЛКМ відкрийте кадр ПЛУ ресурс з ім'ям *SFC(SFCP)* в операторській станції *LS01(VIS)*. В діалоговому вікні заповніть поле, яке позначене червоним кольором. Для цього натисніть ЛКМ на поле і потім на функціональну клавішу *F2*. З'явиться список доступних ресурсів (програм *SFC*), в якому буде лише одна програма з ім'ям *sfc*. Виберіть цю програму ЛКМ та підтвердіть вибір натисненням на кнопку *OK*. Закрийте вікно налаштування ресурсу *SFC(SFCP)* натисненням ЛКМ на кнопку *OK*. Перевірте весь проект на правдоподібність. В позитивному випадку система *CBF* автоматично

згенерує в операторській станції додатковий ресурс – кадр з ім'ям *SFC(SFCP)*.

На завершення завантажить ресурси до процесової та операторської станції. Далі спостерігайте за виконанням проекту в *DigiVis*.

#### 2.4. Розроблення ППЗ для системи управління установкою змішування речовин.

В цьому підрозділі буде розглянуто порядок розроблення ППЗ для системи управління установкою змішування речовин. Враховуючи, що раніше були розглянуті основні етапи створення ППЗ с системі *CBF*, зосередимося на особливостях розроблення проекту. Насамперед надамо коротку характеристику об'єкту – станції змішування. Станція змішування складається з ємностей, трубопроводів, зовнішнього обладнання та вимірювальних і регулюючих пристроїв. Робота станції описується так: оператор вибирає рецепт та запускає процес. Відкриваються клапани подавання сировини та наповнюється ємність. Система контролює рівень в ємності. Після заповнення ємності до визначеного рівня закриваються клапани подавання. Далі вмикається мішалка, яка працює встановлений час. Потім вмикається насос для випорожнення ємності. Функціональна схема автоматизації установки показана на рис. 2.32.

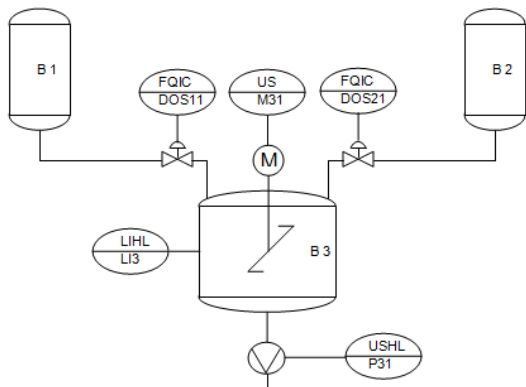


Рис. 2.32 – Функціональна схема автоматизації станції змішування.

Відповідно на рис. 2.33 зображена мнемосхема технологічної установки.

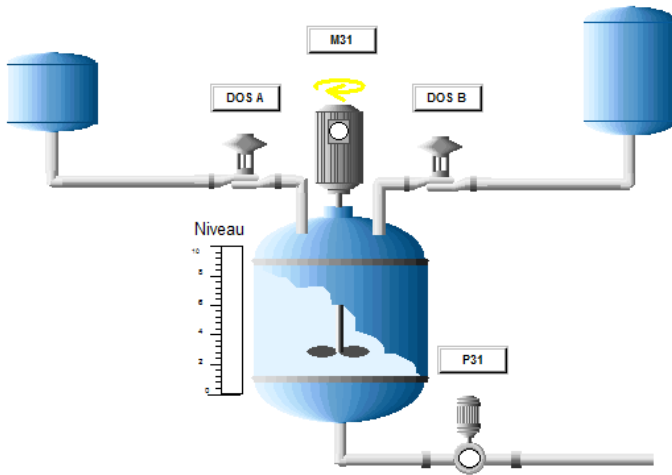


Рис. 2.33 – Мнемосхема станції змішування.

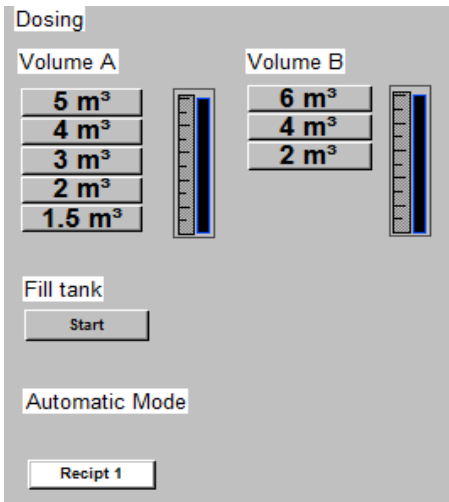


Рис. 2.34 – Людино-машинний інтерфейс станції змішування.

Технологічна установка складається з двох ємностей з початковими речовинами, вхідних трубопроводів з клапанами-дозаторами, ємності з мішалкою, вихідного трубопроводу з насосом. Також в мнемосхемі присутній людино-машинний інтерфейс, який дозволяє формувати рецепти продукту, що виробляється та керувати процесом змішування (див. рис. 2.34).

Отже, створить новий проект в системі *CBF* та додайте до накопичувача блок

користувача з симулятором станції змішування та вільну графіку з мнемосхемою. Далі створить в проекті конфігурацію з програмними ресурсами. Одразу створить апаратну структуру проекту на основі емулятора та зв'яжуть ресурси проекту з апаратними ресурсами. Налаштуйте мережні параметри. В результаті отримаєте дерево проекту, яке показано на рис. 2.35.

В дереві проекту в процесній станції присутні три задачі користувача: рецептурна задача, задача симулятора та власне задача керування установкою змішування. Операторська станція вміщує кадр з мнемосхемою та кадр ПЛУ.

Спочатку розглянемо перелік змінних, які використовуються в проекті. В таблиці 2.2 наведено усі змінні проекту.

У вузлі з рецептурною задачею знаходиться список з програмою запуску установки та програма логічного управління. Розглянемо докладніше цей вузел. Програма запуску установки *Start\_SFC (FBD)* виконує завдання щодо підготовки станції до роботи, тобто до запуску ПЛУ з імям *SFC(SFC)*.

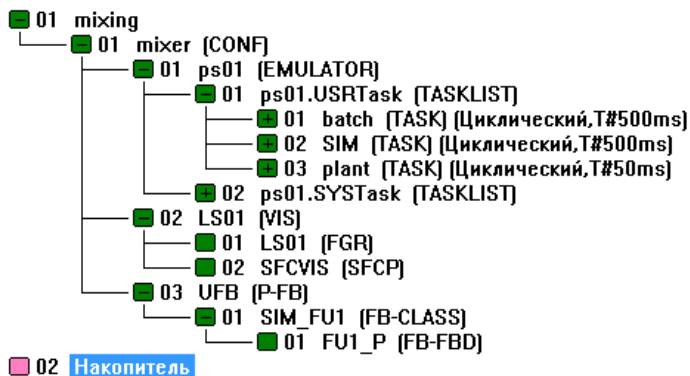


Рис. 2.35 – Дерево проекту станції змішування.

Таблиця 2.2 – Змінні проекту з поясненнями.

Імя	Тип	Пояснення	Meas./output variable
DOS_A_MA	Binary	Dos_A Automatic mode	On / off
DOS_A_MM	Binary	Dos_A Manual mode	On / off
DOS_B_MA	Binary	Dos_B Automatic mode	On / off
DOS_B_MM	Binary	Dos_B Manual mode	On / off
FS_A1	Analog	Flow A	m <sup>3</sup> /s
FS_A2	Analog	Level A	m <sup>3</sup>
FS_B1	Analog	Flow B	m <sup>3</sup> /s
FS_B2	Analog	Level B	m <sup>3</sup>
Inhalt_A	Analog	Input volume, substance A	m <sup>3</sup>
Inhalt_B	Analog	Input volume, substance A	m <sup>3</sup>
Init_AS1	Binary	SFC program starts	On / off
LI03	Analog	Level B3	Meters
LI031	Binary	Limit value B3	On / off
LI032	Binary	Limit value B3	On / off
LI033	Binary	Limit value B3	On / off
LI034	Binary	Limit value B3	On / off
M31_0	Binary	M31 end position 0	On / off
M31_1	Binary	M31 end position 1	On / off
M31_FLT	Binary	Fault input	On / off
M31_MA	Binary	M31 Automatic mode	On / off
M31_MM	Binary	M31 Manual mode	On / off
M31_P0	Binary	M31 Control command 0	On / off
M31_P1	Binary	M31 Control command 1	On / off
M31_out	Binary	Agitator output	On / off
P31_FB0	Binary	P31 End position 0	On / off
P31_FB1	Binary	P31 End position 1	On / off
P31_FLT	Binary	Fault input	On / off
P31_IN	Binary	P31 Input	On / off
P31_MA	Binary	M31 Automatic mode	On / off
P31_MM	Binary	M31 Manual mode	On / off
P31_P0	Binary	P31 control command 0	On / off
P31_P1	Binary	P31 control command 1	On / off
P31_out	Binary	P31 Output	On / off

Продовження табл. 2.2

PRG_AB	Analog	New batch	On / off
Restart_DA	Binary	Limit value metering A	On / off
Restart_DA1	Binary	New batch A	On / off
Restart_DB	Binary	Limit value metering B	On / off
Restart_DB1	Binary	New batch B	On / off
SET_DA	Binary	Reset counter A and B	On / off
TD_A	Binary	Reduction of inward flow A	On / off

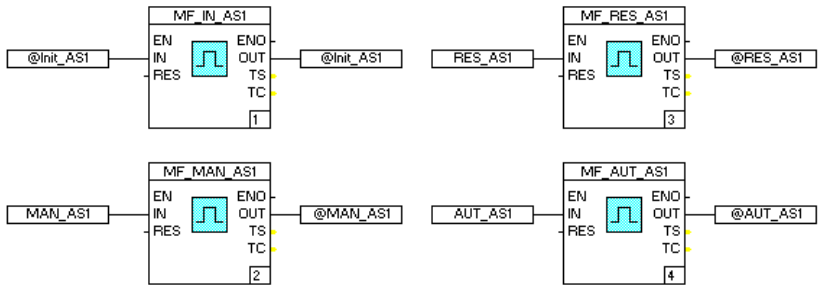


Рис. 2.36 – Програма запуску установки *Start\_SFC (FBD)*.

Дана програма використовує стандартні ФБ *Ждущий мульти-вibrator MONO\_F* і потрібна для ініціювання роботи установки (див. табл. 2.1).

Структура ПЛУ показана на рис. 2.37. Зміст кроків та переходів роз'яснено нижче.

Так в кроці *Start* реалізоване вибір автоматичного режиму роботи установки (див. рис. 2.38). Програма умови переходу зображена на рис. 2.39. Умовою переходу є інверсія змінної *@DOS\_A\_MM*, яка визначає ручний режим роботи дозатора **A**, та присвоєння її стану змінній-прапорцю *.RESULT*.

- 02 SFC [SFC]
  - 01 Start [STEP]
  - 02 Start\_OK [TRANS]
  - 03 SetDos [STEP]
  - 04 Set\_OK [TRANS]
  - 05 Dos [STEP]
  - 06 DosCheck [TRANS]
  - 07 Mix [STEP]
  - 08 MixEnd [TRANS]
  - 09 Pump [STEP]
  - 10 LevelD [TRANS]
  - 11 Set\_D [STEP]
  - 12 End SFC [TRANS]

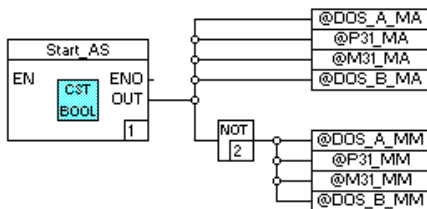


Рис. 2.38 – Програма кроку *Start*.

Рис. 2.37 – Структура ПЛУ станції змішування.

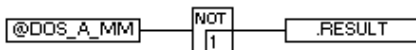


Рис. 2.39 – Програма умови переходу *Start\_OK*.

В кроці *SetDos* встановлюються режими роботи дозаторів (див. рис. 2.40).

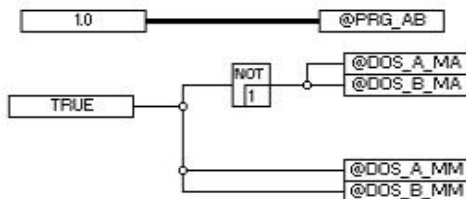


Рис. 2.40 – Програма кроку *SetDos*.

Умовою переходу є одночасний автоматичний режим роботи дозаторів (див. 2. 41).

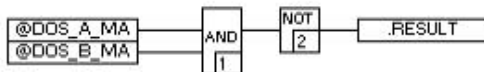


Рис. 2.41 – Програма умови переходу *Set\_OK*.

В кроці *Dos* встановлюється об'єми речовин для дозаторів **A** та **B**. Також в кроці запускається таймер з затримкою вмикання/вимикання *TONOF*. Вихід таймера дозволяє присвоєння значення 1.0 (типу *REAL*) змінній *@PRG\_AB*. Програма кроку *Dos* зображена на рис. 2.42.

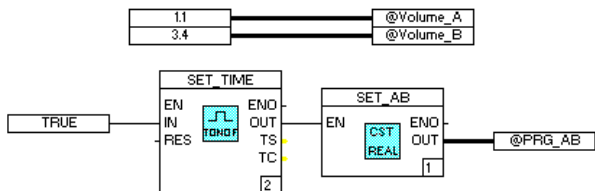


Рис. 2.42 – Програма кроку *Dos*.

Умова переходу до наступного кроку зображена на рис. 2.43. В програмі переходу *DosCheck* використаний *RS*-тригер (з пріоритетом скидання) для визначення стану змінної-меркера *.RESULT*.

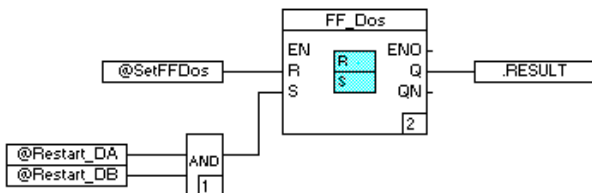


Рис. 2.43– Програма переходу *DosCheck*.

Наступний крок *Mix* здійснює вмикання мішалки на термін, який розраховує лічильник часу *CTT*. Програма цього кроку показана на рис. 2.44.

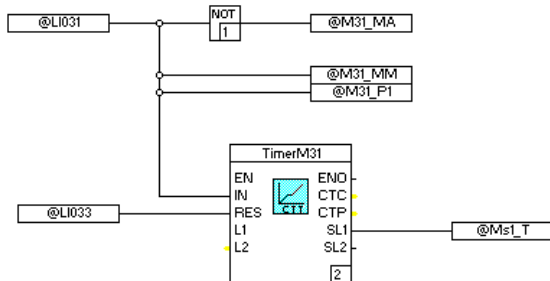


Рис. 2.44 – Програма кроку *Mix*.

Умова переходу *MixEnd* активується за допомогою змінної *@Ms1\_T*, яка отримує статус TRUE наприкінці кроку *Mix*. Це звичайне присвоєння її статусу змінної-меркеру *.RESULT*, як це показано на рис. 2.45.

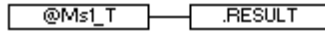


Рис. 2.45 – Програма переходу *MixEnd*.

Наступний крок *Pump* призначений для встановлення режиму роботи блоку *P31* в програмі керування випускним клапаном (програма *Outlet*). Програма кроку показана на рис. 2.46.

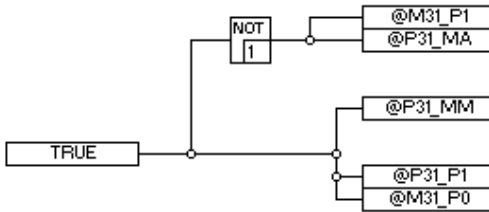


Рис. 2.46 – Програма кроку *Pump*.

Цей крок (рис. 2.46) активний, якщо рівень в ємності (в програмі симулятора) не досягне мінімального значення, тобто готовий розчин не буде повністю викачаний з ємності. В такому випадку змінна *@P31\_P0* встановлюється та її стан присвоюється змінної-меркеру *.RESULT*, як це показано в умові переходу *LevelD* на рис. 2.47.

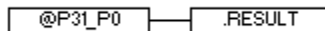


Рис. 2.47 – Програма переходу *LevelD*.

Наступний крок *Set\_0* (див. рис. 2.48) виконує функцію рестарту установки із затримкою за допомогою таймера з затримкою вмикання. Крок діє доки не буде виконана умова переходу *End*, яка показана на рис. 2.49. В умові її стан знов присвоюється змінної-меркеру *.RESULT*.

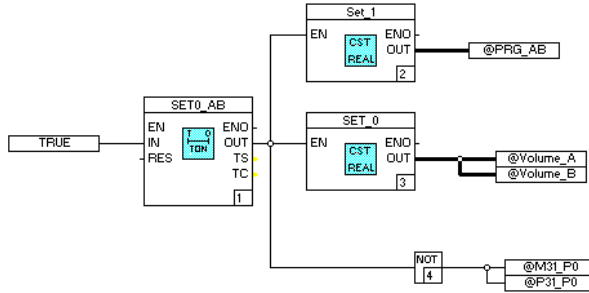


Рис. 2.48 – Програма кроку *Set\_0*.

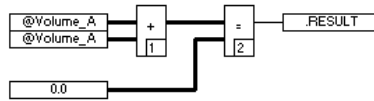


Рис. 2.49 – Програма кроку *End*.

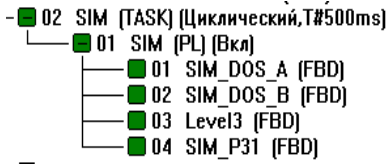


Рис. 2.50 – Дерево задачі *SIM*.

На цьому закінчується цикл роботи станції змішування. Далі розглянемо програми, які входять до списку *SIM(PL)*. Фрагмент дерева проекту з задачею користувача *SIM(TASK)* зображений на рис. 2.50.

До списку *SIM(PL)* дерева проекту входять програми стимуляторів. Це симулятори дозаторів (*SIM\_DOS\_A* та *SIM\_DOS\_A*), рівня (*Level3*) в ємності та насосу (*SIM\_P31*). На рис. 2.51 - 2.54 зображені відповідні програми.

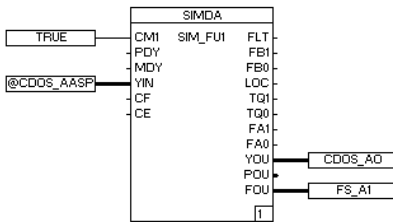


Рис. 2.51 – Програма *SIM\_DOS\_A*.

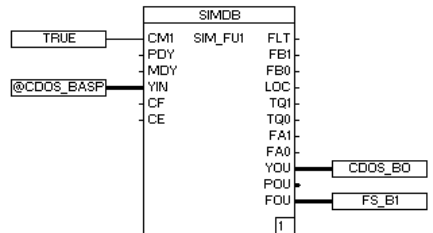


Рис. 2.52 – Програма *SIM\_DOS\_B*.

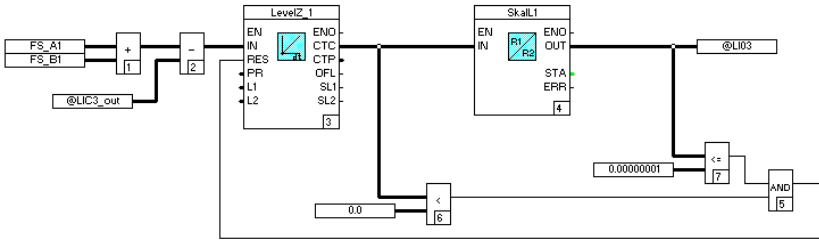


Рис. 2.53 – Програма *Level3*.

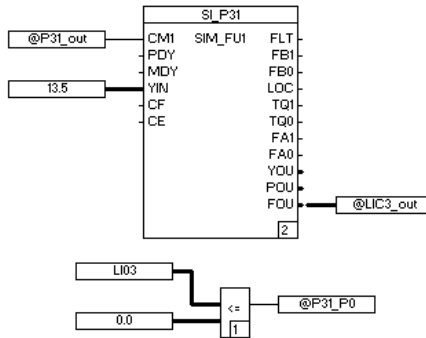


Рис. 2.54 – Програма *SIM\_P31*.

Залишилось розглянути останній вузол дерева проекту – задачу користувача *plant*, в якій знаходяться списки з програмами керування компонентами

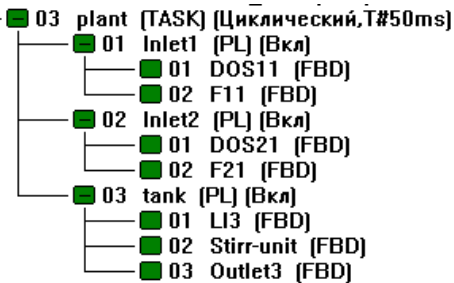


Рис. 2.55 – Дерево задачі *Plant*.

установки: вхідним клапаном, вихідним клапаном та ємністю (див. рис. 2.55). Це власне основна програма керування установкою. До списку програм *Inlet1(PL)* входять програми *DOS11* та *F11*, які зображені на рис. 2.56 та 2.57 відповідно.

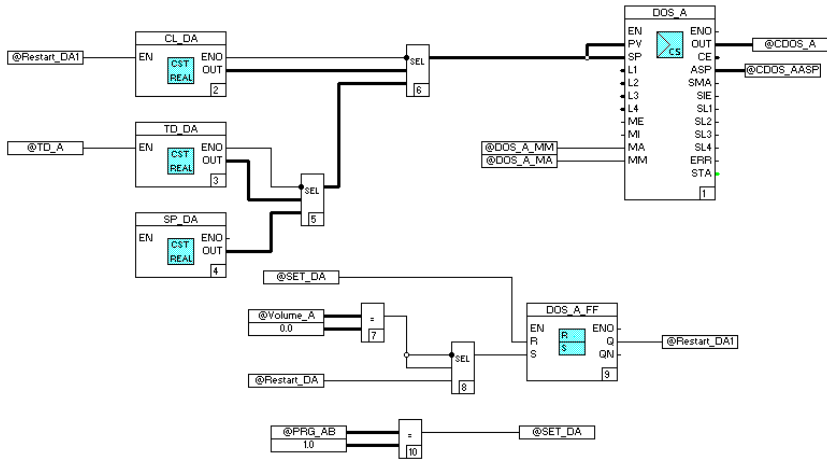


Рис. 2.56 – Програма керування дозатором **A**.

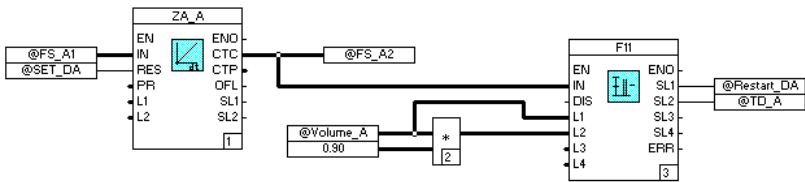


Рис. 2.57 – Програма контролю витрати в дозаторі **A**.

В програмі *DOS11* (див. рис. 2.56) основним блоком є ФБ *C\_CS*, який є безперервним ПІД-регулятором. Налаштуйте параметри регулятора в вікні налаштувань подібно до рис. 2.18 – 2.20. Програма вимірювання витрати (див. рис. 2.57) забезпечується ФБ *CT\_ANA*, який обчислює та накопичує значення витрати та блоком сигналізації *M\_ANA*, який контролює межі витрати. Налаштування блоку *M\_ANA* подібні раніше розглянутим та наведеним на рис. 2.17.

Список програм *Inlet2(PL)*, до якого входять програми *DOS21* та *F21*, які зображені на рис. 2.58 та 2.59 подібні до програм в списку *Inlet1(PL)*.

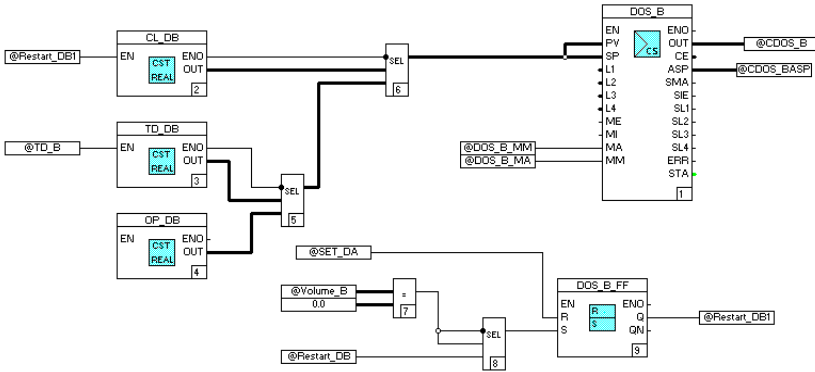


Рис. 2.58 – Програма керування дозатором **B**.

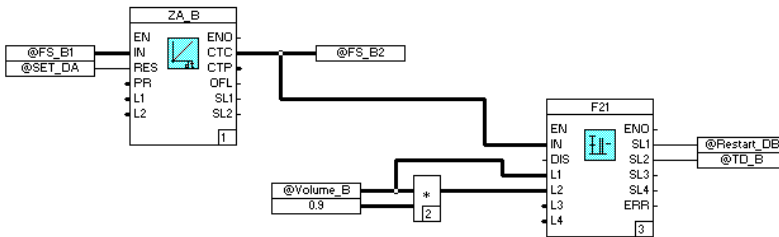


Рис. 2.59 – Програма контролю витрати в дозаторі **B**.

До списку програм *tank(PL)* входять програми *LI3*, *Stirr\_unit* та *Outlet3*, які зображені на рис. 2.60 - 2.62 відповідно. В налаштуваннях блоку *M\_ANA* встановлені такі межі: 0.01, 1.0, 1.5 та 9.5 м.

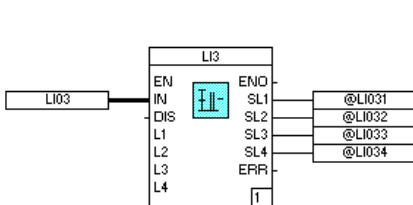


Рис. 2.60 – Програма контролю рівня в ємності.

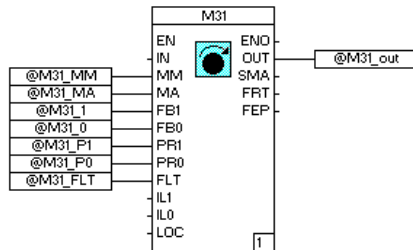


Рис. 2.61 – Програма керування мішалкою в ємності.

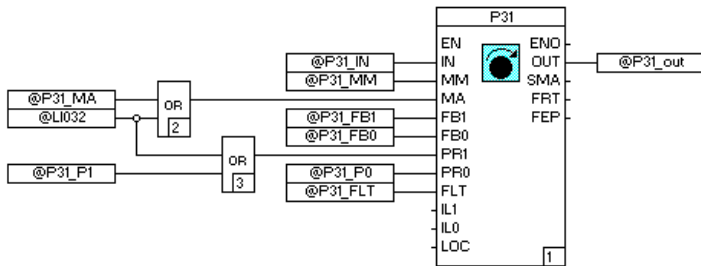


Рис. 2.62 – Програма керування приводом насосу.

В програмах керування мішалкою та приводом насосу використаний вже відомий ФБ *IDF\_1*, який розглянутий раніше (див. рис. 2.16).

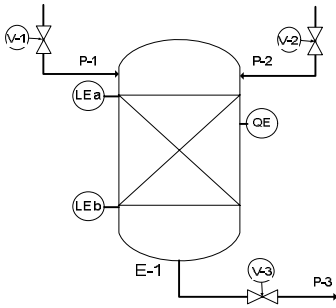
## 2.5. Розроблення самостійного проекту з ППЗ для системи управління установкою змішування речовин.

Для закріплення навичок з розроблення ППЗ в середовищі *CBF* пропонуються завдання для самостійного виконання. В цих завданнях необхідно розробити проект управління реактором або ємністю для змішування. В табл. 2.3 надані завдання для самостійного розроблення.

Таблиця 2.3 – Варіанти завдань для розроблення ППЗ.

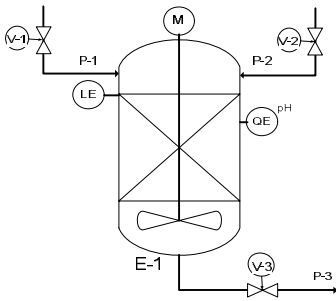
Варіант 1	
	<p>Після натискання кнопки «Пуск» відкривається клапан <i>V-1</i> і рідина заповнює апарат на 40 %. Після цього <i>V-1</i> закривається і відкривається клапан <i>V-2</i> на 90 % на трубопроводі подачі пари. Після досягнення заданої температури 70 °С клапан <i>V-2</i> закривається повністю і рідина витримується в апараті 900 с. Коли термін часу вичерпався відкривається клапан <i>V-3</i> і рідина зливається з апарата. Коли апарат порожній – цикл повторюється</p>

## ВАРІАНТ 2



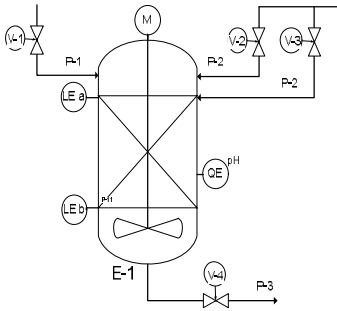
Після натискання кнопки «Пуск» відкривається клапан V-1 і апарат заповнюється продуктом до рівня «b». Далі клапан V-1 закривається. Після витримки часу 140 с відкривається клапан V-2 на 10 %. Якщо через 80 с концентрація (рН) в апараті не досягне заданого значення, то клапан V-2 відкрити на 25 %. Коли концентрація досягнута або досягнуто рівень «a» – повністю закрити клапан V-2 і відкрити клапан V-3 (для зливу суміші з апарата). Після досягнення рівня «b» – цикл повторюється

## ВАРІАНТ 3



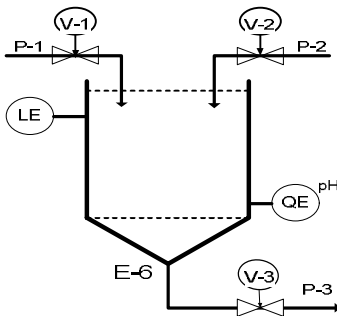
Якщо апарат порожній та натискається кнопка «Пуск», відкривається клапан V-1, апарат заповнюється водою до рівня 50 %. Далі клапан V-1 закривається і на 60 % відкривається клапан V-2. Продукт з 2-го трубопроводу заповнює апарат до рівня 90 %. Далі клапан V-2 закривається і на 540 с вмикається мішалка. Після вичерпання часу – відкривається клапан V-3 і суміш зливається з апарата. Коли апарат порожній – цикл повторюється

### ВАРІАНТ 4



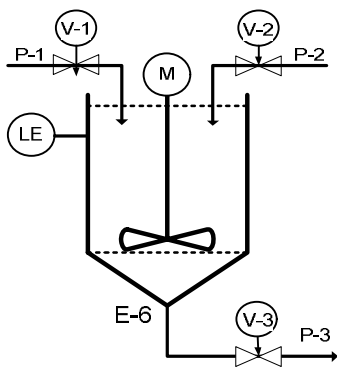
Якщо апарат порожній та натискається кнопка «Пуск», відкривається клапан V-1 і рідина заповнює апарат до рівня «b». Далі клапан V-1 закривається, вмикається мішалка і на 40 % відкривається клапан V-2. Якщо через 175 с концентрація ( $pH$ ) в апараті не досягне потрібного значення, то додатково відкривається клапан V-3 на 15 %. Якщо концентрація досягнута або досягнуто рівень «a» – закрити клапани V-2 та V-3, відкрити клапан V-4 (злив суміші за апарата). Після досягнення рівня «b» – цикл повторюється

### ВАРІАНТ 5



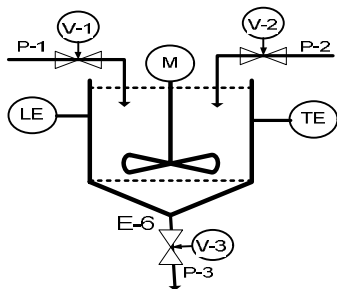
Якщо апарат порожній та натискається кнопка «Пуск», відкривається клапан V-1 і вода заповнює апарат на 60 %. Далі клапан V-1 закривається, на 30 % відкривається клапан V-2 і продукт надходить в апарат до досягнення потрібної концентрації. Якщо концентрація не досягнута, а рівень більше 80 % клапан V-2 закривається, клапан V-3 відкривається (рідина виливається з апарата). Після досягнення нульового рівня – цикл повторюється

### ВАРІАНТ 6



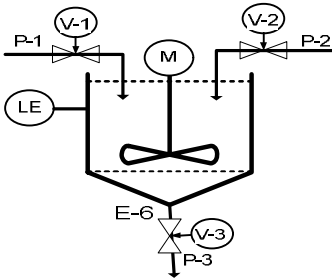
Якщо апарат порожній та натискається кнопка «Пуск», відкривається клапан V-1 на 90 % і вода заповнює апарат на 30 %. Далі клапан V-1 закривається, відкривається клапан V-2 і продукт надходить в апарат до рівня 50 %. Клапан V-2 закривається і на 200 с вмикається мішалка. Коли термін часу вичерпався повністю відкривається клапан V-1 і рідина заповнює апарат на 80 %. Закривається клапан V-1 і знову вмикається мішалка на 10 хв. Далі після закінчення часу відкривається клапан V-3 і рідина виливається з апарата. Після досягнення нульового рівня – цикл повторюється

### ВАРІАНТ 7



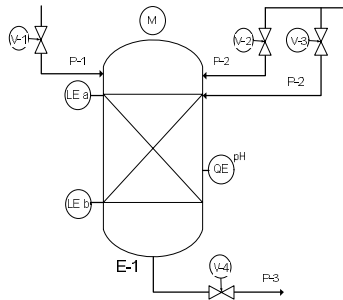
Якщо апарат порожній та натискається кнопка «Пуск», відкривається клапан V-1 і вода заповнює апарат на 50 %. Далі клапан V-1 закривається, відкривається клапан V-2 і вмикається мішалка, гарячий продукт надходить в апарат до досягнення температури заданої величини. Якщо температура досягнута – вмикається мішалка та клапан V-2, відкривається клапан V-3 і рідина виливається з апарата. Після досягнення нульового рівня – цикл повторюється

### ВАРІАНТ 8



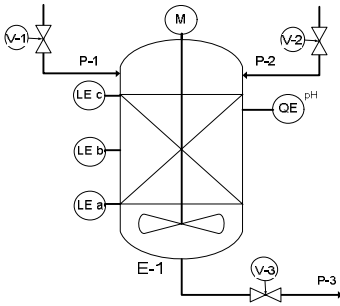
Якщо апарат порожній та натискається кнопка «Пуск», відкривається клапан V-1 і вода заповнює апарат на 40 %. Далі клапан V-1 закривається до 15 %, відкривається клапан V-2 і продукт надходить в апарат до досягнення рівня 80 %. Клапани V-1 та V-2 закриваються і на 220 с вмикається мішалка. Коли час вичерпався відкривається клапан V-3 і рідина виливається з апарата. Після досягнення нульового рівня клапан V-3 закривається. Далі цикл повторюється

### ВАРІАНТ 9



Якщо апарат порожній та натискається кнопка «Пуск» відкривається клапан V-1 і вода заповнює апарат до рівня «b». Далі клапан V-1 закривається і на 150 с на 15 % відкривається клапан V-2. Після цього клапан V-2 залишають відкритим на 10 % і відкривають клапан V-3 на 75 %. Коли концентрація (pH) в апараті досягне потрібного значення або рівень досягне значення «a», клапани V-2 і V-3 повністю закриваються, клапан V-4 відкривається і рідина виливається з апарата до рівня «b». Далі клапан V-4 закривається, а цикл повторюється

## ВАРІАНТ 10



Після натискання кнопки «Пуск» відкривається клапан V-1 на 60 % і вода заповнює апарат до рівня «а». Далі клапан V-1 закривається і на 130 с відкривається клапан V-2 та вмикається мішалка. Після цього клапан V-1 знову відкривається. Коли рівень в апараті досягне «с» або значення рН досягне критичного значення, клапан V-1 закривається. Клапан V-3 відкривається і суміш зливається з апарата до рівня «а». Далі клапан V-3 закривається, а цикл повторюється

Після виконання самостійного завдання необхідно за допомогою вбудованих засобів документування провести документування готового проекту з ППЗ. Крім того, за допомогою функції *Print\_Screen* необхідно зробити скріншоти екранів з візуалізаціями проекту в запущеному стані. Також в звіті потрібно навести потрібні схеми, таблиці та пояснення.

Звіт, крім електронних матеріалів (проект з ППЗ), повинен бути виконаним у друкованому вигляді та складатись з таких елементів:

- а) титульного аркуша;
- б) аркуша із завданням;
- в) обґрунтування технічних рішень для реалізації завдання;
- г) роздруківок лістингу проекту з усіма отриманими результатами у вигляді скріншотів;
- д) висновків та рекомендацій щодо застосування ППЗ.

## СПИСОК ЛИТЕРАТУРИ

1. *EngineerIT Control Builder F*. Инструкция по инжинирингу. Программирование в соответствии со стандартом МЭК 61131-3. Версия 7.1 – Номер заказа *3BDD012504R0222*. – *ABB Inc*. Отдел автоматизации процессов. – Москва.
2. *Freelance*. Каталог продукции. Версия 9.2. – Номер заказа *3BDD015188*. – *ABB Inc*. Отдел автоматизации процессов. – Москва.
3. Ел. джерело: <http://www.abb.com>.
4. Ел. джерело: <http://www.abb.ru/controlsystems>.

## ЗМІСТ

ВСТУП .....	3
РОЗДІЛ 1 ЗАГАЛЬНІ ВІДОМОСТІ ПРО АПАРАТНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ПРОГРАМНО-ТЕХНІЧНОГО КОМПЛЕКСУ <i>FREELANCE 800F</i> ВІД <i>ABB Group</i> .....	5
1.1 Загальні відомості про кейс <i>FREELANCE AC700F</i> .....	5
1.2 Загальні відомості про кейс <i>S700 Remote I/O</i> .....	8
1.3 Архітектура побудови та технічні характеристики ПТК <i>FREELANCE 800F</i> .....	11
1.4 Основні принципи роботи в середовищі <i>Control Builder F</i> .....	20
1.4.1 Початок роботи в середовищі <i>Control Builder F</i> .....	20
1.4.2 Конфігурування структури апаратної частини проекту .....	23
1.4.3 Розроблення програмної частини проекту .....	33
1.4.4 Конфігурування мережної структури проекту .....	35
1.4.5. Розроблення програми користувача .....	37
1.4.6. Завантаження та налагодження програми користувача .....	40
РОЗДІЛ 2 ПРАКТИКУМ З РОЗРОБЛЕННЯ ППЗ ТА ЛМІ РСУ НА ОСНОВІ ПТК <i>FREELANCE 800F</i> .....	43
2.1 Розроблення ППЗ для системи управління дискретним об'єктом ..	43
2.2. Розроблення ППЗ для системи управління установкою водопоста- чання .....	51
2.3. Розроблення програми послідовного керування установкою водопо- стачання на мові <i>SFC</i> .....	60
2.4. Розроблення ППЗ для системи управління установкою змішування речовин .....	63
2.5. Розроблення ППЗ для системи управління реактором .....	76
СПИСОК ЛІТЕРАТУРИ .....	83
ЗМІСТ .....	84

Навчально-інструктивне видання

МЕТОДИЧНІ ВАЗІВКИ

для проведення комп'ютерного практикуму з курсу

«Розподілені системи управління»

для студентів спеціальності

151 – «Автоматизація та комп'ютерно-інтегровані технології»  
денної та заочної форм навчання

Укладачі: ПОДУСТОВ Михайло Олексійович  
ЛИСАЧЕНКО Ігор Григорович  
БАБІЧЕНКО Анатолій Костянтинович  
ШУТИНСЬКИЙ Олексій Григорович

Відповідальний за випуск І. Г. Лисаченко

Робота к виданню рекомендована Н. М. Самойленко

Редактор

План 2018 р., поз.

Підп. до друку \_\_.\_\_.18. Формат 60x84 1/16. Папір офсетний.

Друк – різнографія. Гарнітура Times. Ум. друк. арк. 5,2.

Наклад 50 прим. Зам. №\_\_. Ціна договірна.

---

Видавничий центр НТУ „ХПІ”.

Свідоцтво про державну реєстрацію ДК № 3657 от 24.12.2009 г.  
61002, Харків, вул. Фрунзе, 21.

---

ТОВ "Видавництво "Підручник НТУ "ХПІ".

Свідоцтво про державну реєстрацію ДК № 3656 від 24.12.2009 р.  
61002, Харків, вул. Фрунзе, 21.