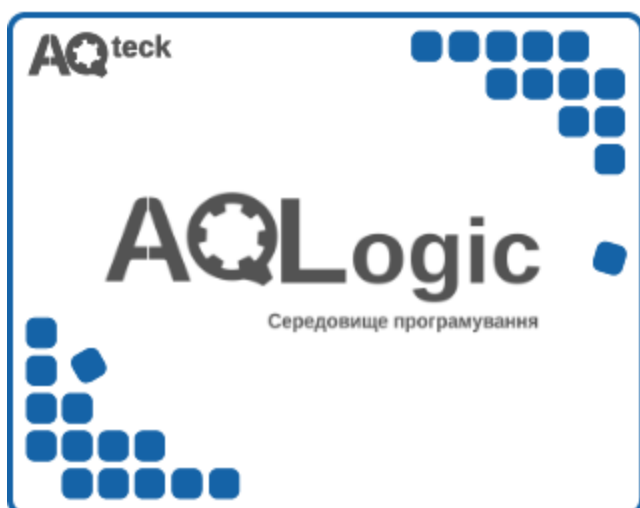


AQLogic (версія 2.5)



Настанова користувача

01.2024

Зміст

1 Про програму	5
1.1 Вимоги до ПК	5
1.2 Використовувані аббревіатури і терміни	5
2 Опис інтерфейсу	7
2.1 Головне меню	7
2.2 Панелі інструментів	9
2.3 Панель Бібліотека компонентів	10
2.4 Панель Властивості.....	12
2.5 Панель Змінні.....	13
2.6 Робоча область.....	15
2.7 Рядок стану	16
2.8 Менеджер і редактор екранів.....	17
3 Базові відомості	20
3.1 Принцип виконання програми	20
3.2 Розробка проекту та порядок роботи.....	21
3.3 Розробка програми	22
3.3.1 Блок коментарів.....	24
3.3.2 Блок змінноїРисунок 3.10	25
3.3.3 Блок константиРисунок 3.15	27
3.3.4 Лінія затримки.....	28
3.3.5 Блок мережевої змінної.....	29
3.3.6 Блок читання/запису у ФБ.....	31
3.3.7 Блок перетворення	31
3.3.8 Перенумерувати компоненти	32
3.3.9 Порядок виконання	33
3.4 Програмування дисплея пристрою	34
3.5 Режим симулятора.....	40
3.6 Налаштування порту та підключення пристрою	43
3.7 Запис програми у пристрій	46
3.8 Онлайн налагодження.....	46
3.9 Відомості про проект.....	48
3.10 Менеджер компонентів.....	49
3.11 Робота з макросами.....	52
3.12 Створення функції мовою ST	61
4 Налаштування пристрою	67
4.1 Екран.....	67
4.2 Годинник	68
4.3 Обмін по мережі.....	69
4.3.1 Інтерфейси	70
4.3.1.1 RS-485	71
4.3.2 Modbus.....	71
4.3.2.1 Робота за протоколом Modbus	72
4.3.2.2 Режим Master.....	75
4.3.2.3 Режим Slave.....	80
4.4 Модулі розширення	82
4.5 Входи та виходи	84
5 Змінні	86
5.1 Типи змінних.....	89
5.2 Сервісні змінні.....	90
5.3 Змінні інтерфейсу зв'язку	91
5.4 Копіювання змінних	92
6 Бібліотека	93
6.1 Функції.....	93
6.1.1 Логічні функції.....	93

6.1.1.1 I (AND).....	94
6.1.1.2 АБО (OR)	94
6.1.1.3 НЕ (NOT).....	95
6.1.1.4 Виключне АБО (XOR).....	96
6.1.2 Арифметичні функції	96
6.1.2.1 Додавання (ADD, fADD)	97
6.1.2.2 Віднімання (SUB, fSUB)	97
6.1.2.3 Множення (MUL, fMUL)	98
6.1.2.4 Ділення (DIV, fDIV).....	99
6.1.2.5 Ділення із залишком (MOD)	99
6.1.2.6 Зведення числа до ступеня (fPOW)	100
6.1.2.7 Взяття модуля від числа (fABS)	100
6.1.3 Функції порівняння	101
6.1.3.1 Дорівнює (EQ)	101
6.1.3.2 Більше (GT, fGT).....	101
6.1.3.3 Вибір (SEL, fSEL).....	102
6.1.4 Зсувні функції	103
6.1.4.1 Побітовий логічний зсув вліво (SHL)	103
6.1.4.2 Побітовий логічний зсув вправо (SHR)	104
6.1.5 Бітові функції	104
6.1.5.1 Читання біта (EXTRACT)	104
6.1.5.2 Запис біта (PUTBIT)	105
6.1.5.3 Дешифратор (DC32)	105
6.1.5.4 Шифратор (CD32)	106
6.2 Функціональні блоки	107
6.2.1 Тригери	107
6.2.1.1 RS-тригер з пріоритетом відключення	107
6.2.1.2 SR-тригер з пріоритетом включення	108
6.2.1.3 Детектор переднього фронту імпульсу (RTRIG)	109
6.2.1.4 Детектор заднього фронту імпульсу (FTRIG)	109
6.2.1.5 D-тригер (DTRIG)	109
6.2.2 Таймери	110
6.2.2.1 Імпульс включення заданої тривалості (TP)	110
6.2.2.2 Таймер із затримкою включення (TON).....	111
6.2.2.3 Таймер із затримкою відключення (TOF)	112
6.2.2.4 Інтервальний таймер (CLOCK).....	113
6.2.2.5 Інтервальний таймер із тижневим циклом (CLOCKWEEK).....	114
6.2.3 Генератори	115
6.2.3.1 Генератор прямокутних імпульсів (BLINK).....	115
6.2.4 Лічильники	116
6.2.4.1 Інкрементний лічильник з автоскиданням (CT)	116
6.2.4.2 Універсальний лічильник (CTN)	117
6.2.4.3 Інкрементний лічильник (CTU)	119
6.2.5 Регулятори	119
6.2.5.1 ПІД-регулятор (PID)	120
6.3 Макроси проекту	123
6.4 Функція на ST	124
6.5 Елементи керування.....	127
6.5.1 Мітка.....	127
6.5.2 Ввід/вивід (int/float)	129
6.5.3 Ввід/вивід (boolean)Рисунок 6.103	131
6.5.4 Динамічний текст.....	133
6.5.5 Комбінований список (ComboBox).....	135
7 Робота з пристроєм	138
7.1 Інформація про пристрій	138
7.2 Час циклу	138
7.3 Оновлення вбудованого ПЗ	139
7.4 Юстування	141
7.4.1 Юстування входів	142
7.4.2 Юстування виходів	144

8	Зміна цільової платформи	145
9	Посідання клавiш	146
10	Приклади проектiв	148
10.1	Включення свiтла з автоматичним вiдключенням	148
10.2	Автоматичне керування електродвигуном мiшалки	150
10.3	Виведення значення змiнної на екран	153
10.4	Задання уставок таймерiв з екрану	155
10.5	Пiдключення ПР200 до OwenCloud через шлюз ПМ210 за протоколом Modbus RTU	156
10.6	Пакування/розпакування бiт/цiлих чисел	160
10.7	Обробка бiтових змiнних за допомогою маски	161
11	Мова програмування ST	162
11.1	Загальнi вiдомостi	162
11.2	Типи даних	162
11.3	Конструкцiї мови	162
11.3.1	Арифметичнi операцiї	162
11.3.2	Логiчнi операцiї	163
11.3.3	Операцiї порiвняння	163
11.3.4	Присвоєння	163
11.3.5	Конструкцiя IF – ELSIF – ELSE	163
11.3.6	Конструкцiя CASE	164
11.3.7	Цикл FOR	165
11.3.8	Цикл WHILE	166
11.3.9	Цикл REPEAT UNTIL	167

1 Про програму

AQLogic – середовище програмування, призначене для створення алгоритмів роботи комутаційних пристроїв, що належать до класу програмованих реле, зокрема, пристроїв серій ПР1хх, ПР200 і панелі ИПП120 виробництва компанії АКУТЕК.

Основний функціонал AQLogic доступний для всіх пристроїв, функціонал та інтерфейси для пристроїв на новій платформі недоступні для інших пристроїв.

AQLogic дозволяє користувачеві розробити програму автоматизації системи за власним алгоритмом та записати її в незалежну пам'ять пристрою. Для створення програми використовується графічна мова FBD стандарту МЕК 61131-3.

Перелік пристроїв, для програмування яких можна використовувати AQLogic, представлений на сайті компанії АКУТЕК.

1.1 Вимоги до ПК

Для роботи AQLogic потрібна операційна система Windows 7/8/10 та програмна платформа «. NET Framework» версії 4.0. або вище. Якщо програмна платформа «. NET Framework» не встановлена, запит на її встановлення з'явиться автоматично.

Мінімальна конфігурація:

- процесор Intel Atom 1.5 ГГц;
- оперативна пам'ять 1 Гб;
- вільне місце на диску 100 Мб;
- вільний USB-порт для підключення пристрою;
- клавіатура та миша;
- дисплей з роздільною здатністю 1024 × 768.

Рекомендована конфігурація:

- процесор Intel Core i3 2 ГГц;
- оперативна пам'ять 4 Гб;
- вільне місце на диску 200 Мб;
- вільний USB-порт для підключення пристрою;
- клавіатура та миша;
- дисплей з роздільною здатністю 1280 × 800.

Підключення до Інтернету потрібне для таких дій:

- оновлення AQLogic;
- завантаження шаблонів мережевих пристроїв;
- завантаження макросів у Менеджері компонентів.

1.2 Використовувані аббревіатури і терміни

FBD (Function Block Diagram) — графічна мова діаграм функціональних блоків, яка призначена для візуального програмування.

OwenCloud — хмарний сервіс компанії «ОВЕН», що використовується для віддаленого моніторингу, керування та зберігання архівів даних пристроїв, які використовуються у системах автоматизації. Доступ до сервісу здійснюється за допомогою web-браузера або мобільної програми (докладніше див. aqteck.com.ua/ua/opys-servisu).

ST (Structured Text) — мова програмування стандарту IEC61131-3. Призначена для програмування промислових контролерів та операторських станцій.

ЕСПЗП (Електронно-Стираємий Програмований Постійний Запам'ятовувальний Пристрій) — частина інтегральної схеми мікропроцесора, яка використовується для зберігання даних.

Комутаційна програма, програма – розроблений користувачем на схемі проекту алгоритм роботи для подальшого запису в пристрій.

Контекстне меню — елемент графічного інтерфейсу, що представляє собою список команд, які викликаються користувачем для вибору необхідної дії над обраним об'єктом. В AQLogic викликається натисканням ПКМ по об'єкту.

ЛКМ — ліва кнопка миші.

Логічний сигнал («сигнал») — дискретна фізична величина (напруга чи струм), що приймає лише два значення: включено – відповідає логічній «1» (лог. «1»), і відключено – логічному «0» (лог. «0»).

Макрос — функціональний блок, розроблений користувачем.

ОЗП (Оперативний Запам'ятовувальний Пристрій) — енергозалежна частина пам'яті пристрою, в якій під час роботи зберігається код програми, що виконується, а також вхідні, вихідні та проміжні дані, які обробляються процесором.

Перетворювач — прилад, через який пристрій підключається до ПК.

ПЗ — програмне забезпечення.

ПЗП (Постійний Запам'ятовувальний Пристрій) — енергонезалежна пам'ять, яка використовується для зберігання масиву незмінних даних.

ПК — персональний комп'ютер.

ПКМ — права кнопка миші.

Полотно, схема – поле для розміщення графічних компонентів проекту та редагування з'єднувальних ланцюгів між ними.

Пристрій – програмований пристрій, наприклад ПР200.

Проект — файл, створений в AQLogic, що включає програму і налаштування для подальшого зберігання на ПК.

РКІ – рідкокристалічний індикатор.

Слот — роз'єм у пристрої для підключення інтерфейсу зв'язку або модуля розширення.

Уставка — значення параметра функціонального блоку (встановлений режим).

Функціональний блок (ФБ) – структурна одиниця програми, яка після виконання видає одне або більше значень. Може бути створено множину поіменованих екземплярів (копій) функціонального блоку.

Функція — структурна одиниця програми, яка після виконання видає лише одне значення. Функція не зберігає інформацію про свій внутрішній стан, тобто виклик функції з тими самими фактичними параметрами видає те саме значення.

Цикл — час виконання пристроєм заданої програми (залежить від кількості виконуваних операцій у програмних ланцюгах).

2 Опис інтерфейсу

Після запуску AQLogic на моніторі ПК відкривається Головне вікно:

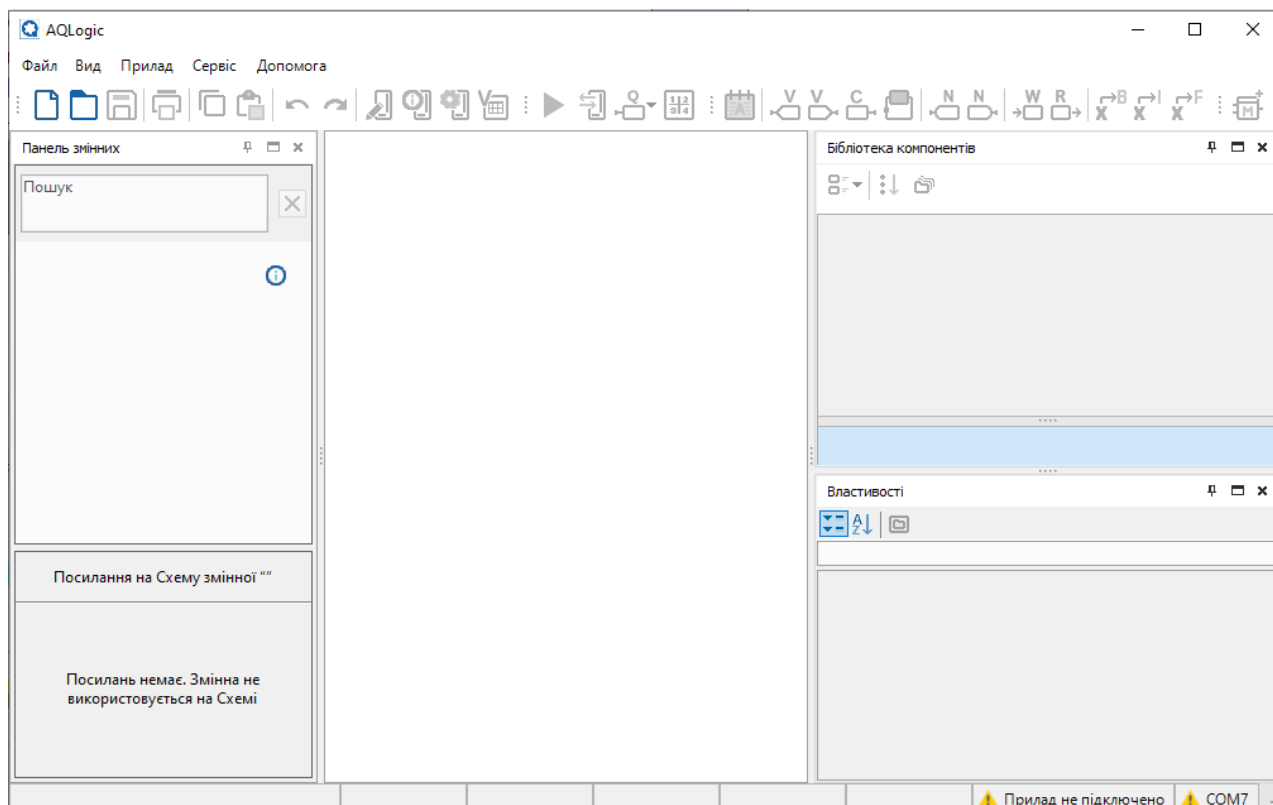


Рисунок 2.1

Головне вікно містить:

- [голове меню 2.1](#): **Файл, Вид, Прилад, Сервіс, Допомога**;
- [панелі інструментів 2.2](#) ;
- панелі [Бібліотека компонентів 2.3](#), [Властивості 2.4](#) і [Змінні 2.5](#) (до відкриття чи створення проекту в них немає інформації);
- [робочу область 2.6](#) проекту – поле редагування програми (до відкриття чи створення проекту порожнє);
- [рядок стану 2.7](#) в нижній частині головного вікна, що показує інформація про доступні ресурси пристрою та підключення до AQLogic;
- менеджер екранів.

2.1 Головне меню

Файл

Новий проект	Створення нового проекту. Поточний проект закривається, перед закриттям буде запропоновано зберегти проект
Зміна цільової платформи	Виклик вікна зміни модифікації пристрою для проекту
Відкрити проект	Відкриття раніше створеного та збереженого проекту
Зберегти поточний документ	Збереження поточного проекту
Зберегти проект	Перезбереження проекту під поточним ім'ям
Зберегти проект як	Збереження проекту під новим ім'ям або в іншому місці
Створити файл із ключем	Створення файлу з ключем
Відомості про проект	Виклик вікна відомостей про проект 3.9
Створити макрос	Виклик редактора макросів 3.11

Зберегти проект як новий макрос	Присвоєння імені макросу та збереження (кнопка неактивна, доки не відкритий редактор макросів)
Імпорт макросу з файлу	Імпорт макросу з файлу до бібліотеки компонентів
Експорт макросу до файлу	Експорт макросу для збереження на ПК
Менеджер компонентів	Виклик вікна Менеджера компонентів 3.10
Друк	Друк схеми відкритого проекту
Останні проекти	Список нещодавно відкритих проектів
Вихід	Вихід з AQLogic

Вид

Скасувати	Скасування останньої дії у проекті
Відновити	Відновлення скасованої дії
Індикатори станів	Включення/відключення індикаторів у рядку стану 2.7
Панель бібліотек	Відкрити/відобразити панель Бібліотека компонентів 2.3
Панель властивостей	Приховати/відобразити панель Властивості 2.4
Змінні	Приховати/відобразити панель Змінні 2.5
Менеджер екранів	Приховати/показати Менеджер екранів 3.4 (меню доступне для пристроїв з дисплеєм)
Скинути налаштування розташування панелей	Скидання налаштувань розташування панелей користувачем і повернення до налаштувань за умовчанням

Прилад

Записати програму в прилад	Запуск запису створеної програми до пристрою (кнопка неактивна, доки не підключено пристрій)
Оновити вбудоване ПЗ	Запуск оновлення прошивки підключеного пристрою
Інформація	Виклик вікна з інформацією 7.1 про підключений пристрій
Таблиця змінних	Виклик вікна для роботи зі змінними 5 у проекті
Юстування входів/виходів	Виклик вікна юстування 7.4 (кнопка неактивна, доки не підключено пристрій)
Налаштування приладу	Виклик вікна налаштування пристрою 4
Налаштування порту	Виклик вікна налаштування порту 3.6 зв'язку з пристроєм

Сервіс

Перенумерувати компоненти	Зміна нумерації ФБ у робочій зоні проекту
Режим симулятора	Запуск режиму симуляції 3.5
Режим OFFLINE	Включення/відключення заборони опитування пристрою по USB












Допомога

Довідка	Виклик вікна довідки
Історія версій	Виклик у браузері сторінки зі списком версій ПЗ та описом
Про програму	Виклик вікна інформації про поточну версію AQLogic

2.2 Панелі інструментів





Панель інструментів



	Новий проект	Створення нового проекту. Поточний проект закривається, перед закриттям буде запропоновано зберегти проект
	Відкрити проект	Відкриття раніше створеного та збереженого проекту
	Зберегти проект	Збереження поточного проекту. При першому збереженні викликає вікно для надання імені файлу
	Друк	Друк схеми відкритого проекту
	Копіювати	Копіювання виділеного в робочій області елемента
	Вставити	Вставка скопійованого елемента
	Відмінити	Скасування останньої дії у проекті
	Відновити	Відновлення скасованої дії
	Записати програму у пристрій	Запуск запису створеної програми до пристрою
	Інформація	Виклик вікна з інформацією про підключений пристрій
	Налаштування пристрою	Виклик вікна з налаштуваннями пристрою
	Таблиця змінних	Виклик вікна для роботи зі змінними у проекті

Панель налагодження



	Режим симулятора	Запуск режиму симуляції
	Онлайн налагодження	Запуск режиму онлайн налагодження
	Порядок виконання	Зміна порядку обчислення значень на виходах програми або макросу та порядку передання значень щодо зворотних зв'язків
	Перенумерувати компоненти	Зміна нумерації ФБ у робочій зоні проекту

Панель вставки

	Блок коментарів	Додавання коментаря на схему проекту
	Створення блоку вихідний змінної	Створення блоку для запису значення у змінну
	Створення блоку вхідної змінної	Створення блоку для зчитування значення зі змінної
	Створення блоку константи	Створення блоку з фіксованим числовим значенням
	Лінія затримки	Створення затримки на 1 цикл
	Створення мережевої вихідної змінної	Створення блоку змінної, значення якої задається через мережу
	Створення мережевої вхідної змінної	Створення блоку змінної, значення якої зчитується по мережі
	Створення блоку запису у ФБ	Створення блоку для запису значення змінної ФБ, на схемі може не відображатися з'єднувальна лінія ланцюга передачі сигналу
	Створення блоку читання з ФБ	Створення блоку для зчитування значення змінної ФБ, на схемі може не відображатися з'єднувальна лінія ланцюга передачі сигналу
	Перетворювач на булевське значення	Конвертація значення сигналу в булевське значення
	Перетворювач на цілочисельне значення	Конвертація значення сигналу у цілочисельне значення
	Перетворювач на число з рухомою точкою	Конвертація значення сигналу в число з рухомою точкою
	Створити макрос	Створення нового макросу у проекті
	Створити функцію на ST	Створення функції мовою ST для використання у проекті

2.3 Панель Бібліотека компонентів

Включити/відключити відображення панелі **Бібліотека компонентів** в робочій області можна в головному меню **Вид**.

Якщо в AQLogic відкрито проект, то на панелі **Бібліотека компонентів** відображаються її розділи:

- [Функції 6.1](#);
- [Функціональні блоки 6.2](#);
- [Макроси проекту 6.3](#);
- [Функції на ST 3.12](#).

Розділ вибирається у нижній частині панелі.

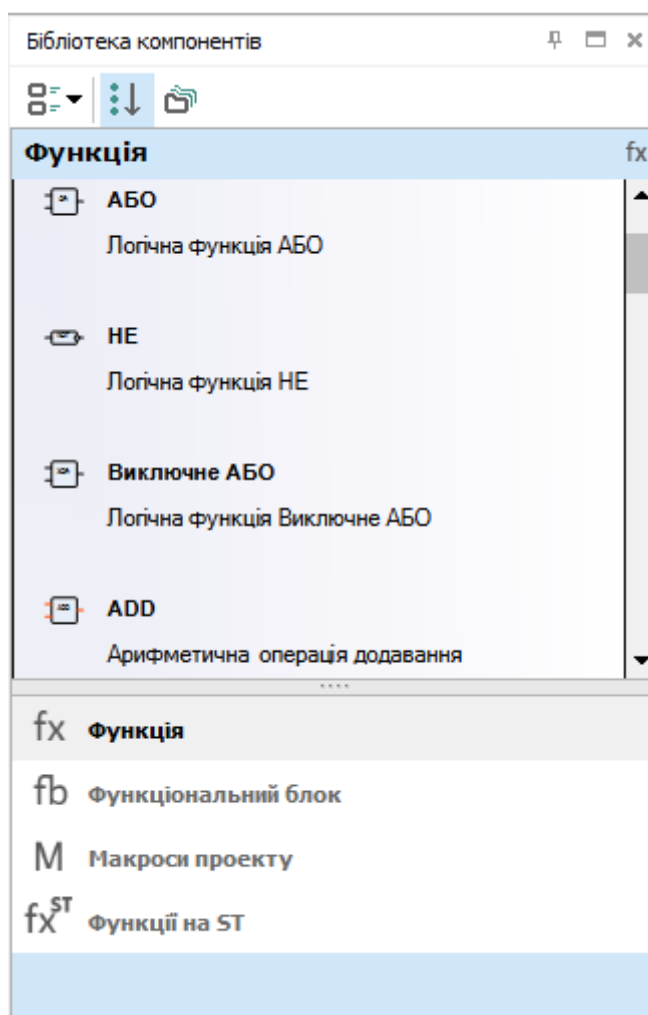


Рисунок 2.2

Вид

У випадковому меню можна вибрати вид розташування компонентів:

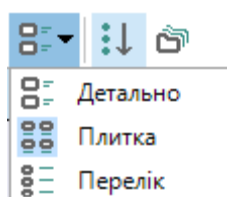


Рисунок 2.3

Спосіб відображення компонентів можна вибрати за допомогою кнопок у верхній частині панелі:

-  **Показати всі компоненти** – відображаються всі компоненти вибраного розділу;

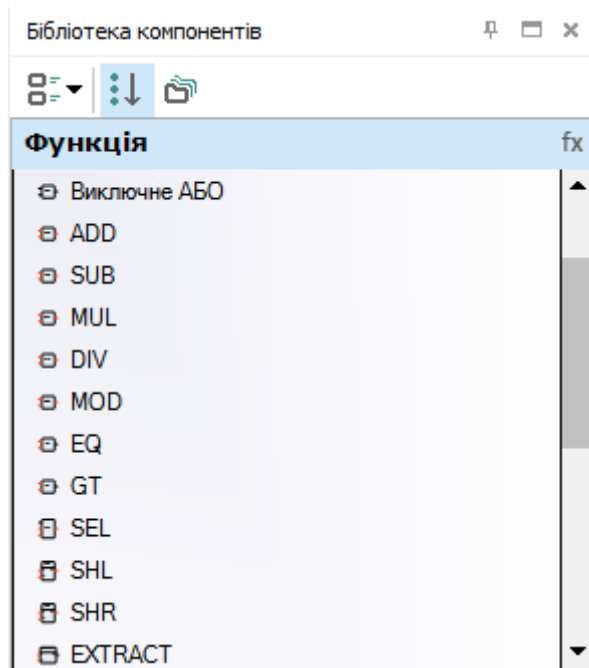


Рисунок 2.4

-  **Групування по папках** – відображаються папки, що містять компоненти різних типів.

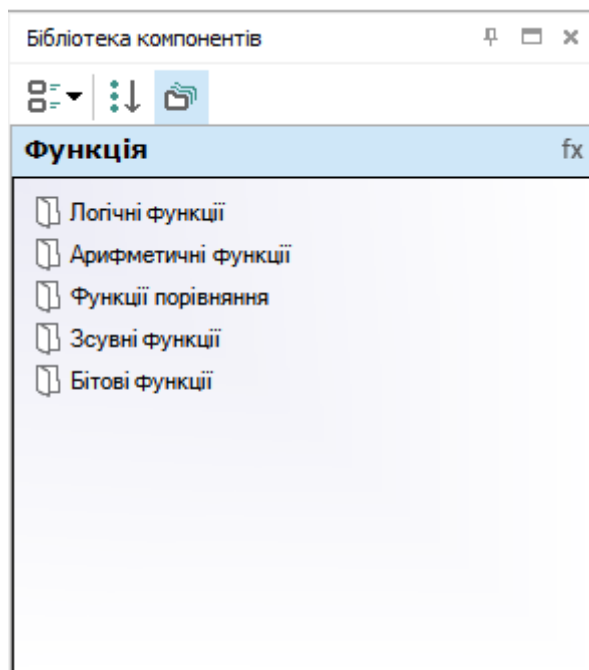


Рисунок 2.5

Опис розділів та компонентів панелі **Бібліотека компонентів** наведено у розділі [Бібліотека 6](#).

2.4 Панель Властивості

Включити/відключити відображення панелі **Властивості** в робочій області можна в головному меню **Вид**.

На панелі відображаються та редагуються:

- параметри елементів програми;
- розміри полотна;
- властивості входів і виходів, що часто використовуються (повний перелік властивостей знаходиться в розділі [Входи 4.5](#) налаштувань пристрою);
- параметри коментаря, вільних змінних, блоків перетворювачів.

Якщо елемент не вибрано, на панелі відображаються властивості полотна.

Для відображення властивостей елемента на панелі слід натиснути на нього ЛКМ.

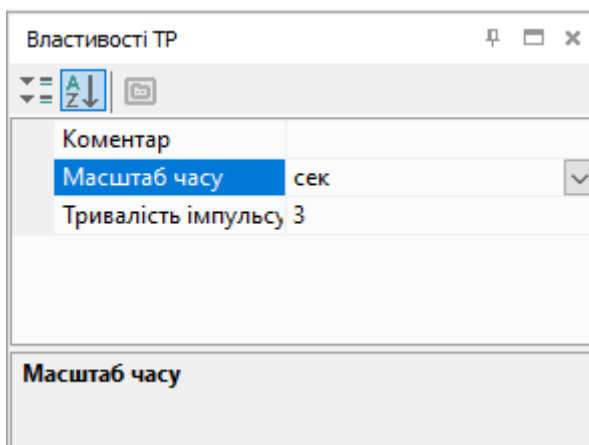




Рисунок 2.6

Способи відображення параметрів на панелі:

-  — в алфавітному порядку;
-  — за категоріями.

2.5 Панель Змінні

Включити/відключити відображення **Панелі змінних** в робочій області можна в головному меню **Вид**.

На **Панелі змінних** відображаються змінні з [Таблиці змінних 5](#).

Створення блоку змінної

Для створення **блоку вхідної змінної** слід перетягнути змінну (методом drag&drop) на полотно.

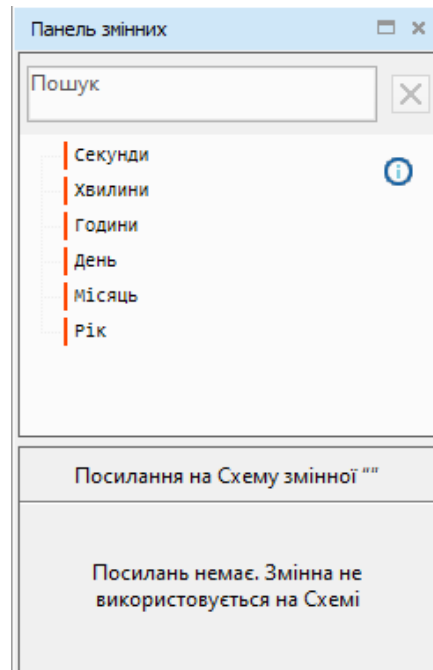


Рисунок 2.7

Для створення **блоку вихідної змінної** слід перетягнути змінну на полотно, утримуючи клавішу Shift. Якщо змінну перетягнути до конектора елемента схеми, буде створено блок змінної, приєднаний до цього конектора.

Посилання

Для відображення блоків, до яких прив'язана змінна, слід натиснути на ім'я змінної на панелі. У ділянці посилань панелі з'являться посилання на блоки. Якщо натиснути на посилання, то на полотні буде виділено блок, до якого прив'язана змінна.

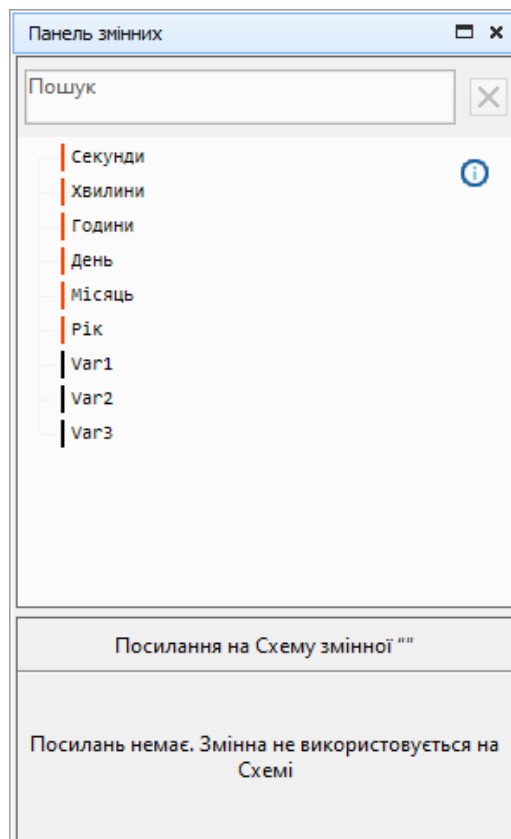


Рисунок 2.8

Прив'язки змінної до інших блоків можна переглянути в пункті **Показати посилання** контекстного меню блоку змінної. Якщо натиснути на посилання в контекстному меню, то буде здійснено перехід до обраного блоку.

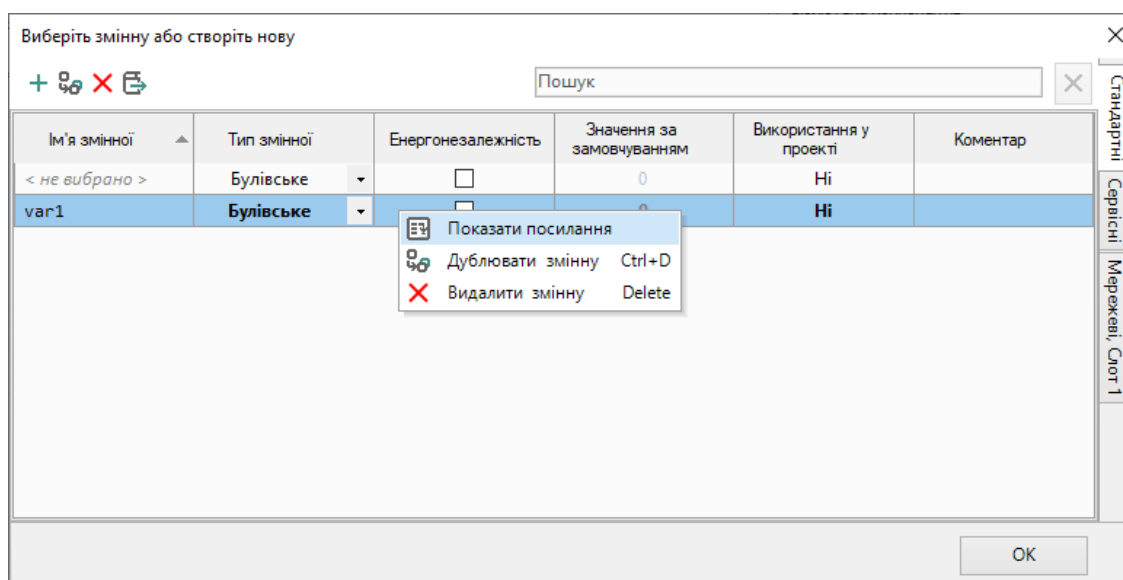


Рисунок 2.9

2.6 Робоча область

Після створення проєкту або відкриття збереженого проєкту в робочій області з'являється полотно для розробки програми на мові FBD.

Якщо пристрій підтримує роботу з функціями мовою ST, з'явиться вкладка **Редактор функції**.

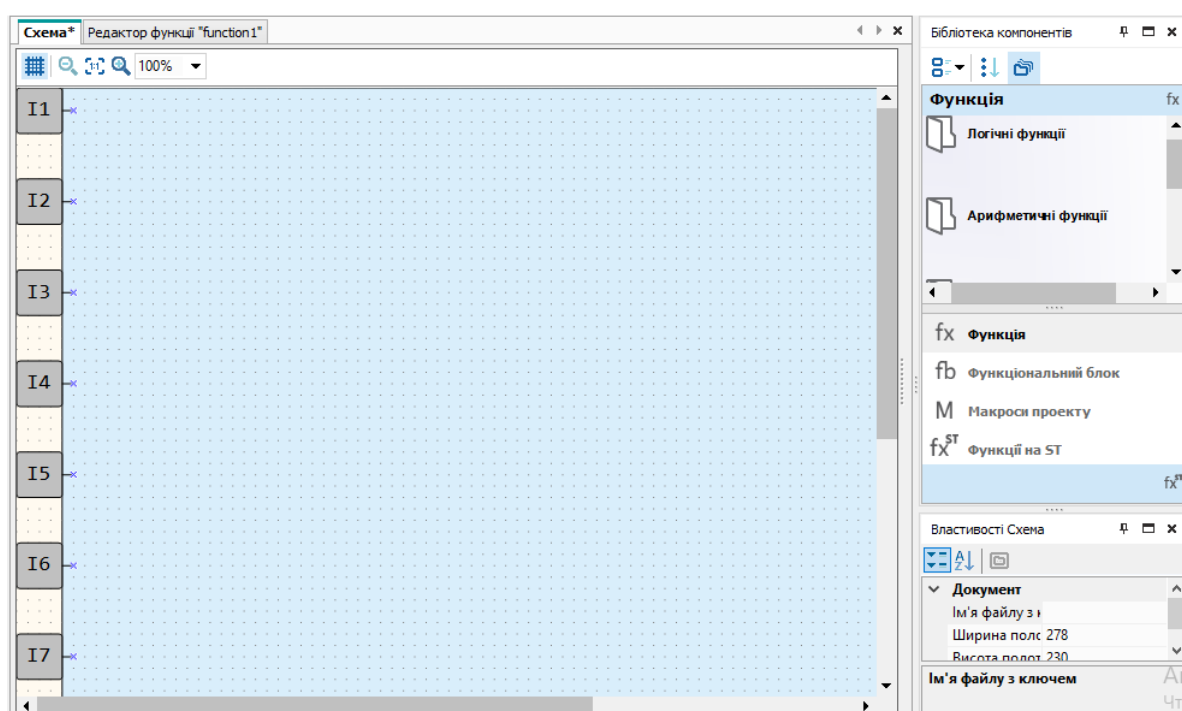


Рисунок 2.10

На полотні розміщуються елементи та блоки з [панелі Бібліотека компонентів 2.3](#) та [панелі вставки](#). Змінити розміри полотна можна на [панелі Властивості 2.4](#).

Входи та виходи

Входи та виходи пристрою позначені у вигляді квадратів з маркуванням номерів:

- вздовж лівого краю полотна розташовані **входи** (**Ix** – дискретні, **Fix** – швидкі дискретні, **Aix** – аналогові);
- вздовж правого краю полотна розташовані **виходи** (**Qx** – дискретні, **AOx** – аналогові, **K** – ключі, **Fx** – світлодіоди).

Цифри в позначеннях відповідають номерам фізичних входів та виходів пристрою. Входи та виходи можна перетягувати вгору/вниз уздовж полотна методом drag&drop, видаляти та додавати для зручності компонування схеми.

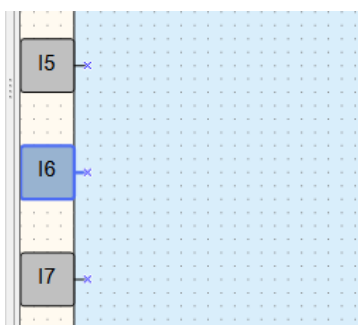


Рисунок 2.11

Керування

Над полотном розташовуються кнопки:

	Включити/Відключити сітку	При включеній сітці на полотні відображаються вертикальна та горизонтальна лінійки і сітка. Елементи та лінії зв'язку при розміщенні прив'язуються до сітки
	Зменшити масштаб	Зменшення масштабу на 10 % від початкового
	Оригінальний розмір	Повернення до вихідного масштабу
	Збільшити масштаб	Збільшення масштабу на 10 % від початкового

Задати потрібний масштаб можна за допомогою меню, що випадає праворуч від описаних вище кнопок.

У нижній частині полотна знаходяться кнопки:

	Розділити	Використовується для одночасного відображення двох ділянок однієї схеми у різних вікнах
	Об'єднати	Використовується для відображення схеми в одному вікні

2.7 Рядок стану

На рядку стану відображається інформація про доступні ресурси пристрою та підключення.

Вид

Залежно від пристрою, вибраного під час створення проекту, перелік індикаторів у рядку стану може різнитися.

ФБ: 0%	Змін.: 0%	EEPROM: 13%	ПЗП: 1%	ОЗП: 6%	⚠ Прилад не підключено	✓ COM6
--------	-----------	-------------	---------	---------	------------------------	--------

Рисунок 2.12

Індикатори

При підключеному пристрої у рядку стану відображається інформація:

- **ФБ** — доступна кількість екземплярів функціональних блоків;
- **Змін.** — доступна кількість змінних;
- **Стек** — рівень використання пам'яті, виділеної під стек. Стек використовується для проміжних обчислень у програмі;
- **Сист. ЕСППЗП** — доступна кількість системної енергонезалежної пам'яті. Індикатор заповнюється у разі використання мережних змінних на схемі, прив'язки змінних до візуалізації та параметрів пристрою і додавання до проекту модулів розширення;
- **ЕСППЗП** — доступна кількість енергонезалежної пам'яті. Індикатор заповнюється у разі використання на схемі стандартних незалежних змінних;
- **ПЗП** — доступний ПЗП пристрою у відсотках загального обсягу: «ПЗП: 0...100 %»;

- **Сист. ОЗП** — доступний системний ОЗП пристрою. Індикатор з'являється у разі заповнення ОЗП більш ніж на 80 %.
- **ОЗП** — доступний ОЗП пристрою у відсотках від загального обсягу: «ОЗП: 0...100 %». Чим більше використовується у програмі блоків та функцій, тим більше пам'яті потрібно для роботи у пристрої. AQLogic автоматично розраховує доступну пам'ять пристрою та у разі критичного значення виводить відповідне попередження;
- **ПРxxx-xxx** — модель підключеного пристрою. Якщо пристрій не підключено, на індикаторі відображається напис **Прилад не підключено**. Натискання на індикатор включає/відключає режим OFFLINE;
- **СОМx** — номер вибраного користувачем порту для підключення пристрою. Натискання на індикатор відкриває вікно [налаштування порту 3.6](#).



ПОПЕРЕДЖЕННЯ

Якщо будь-який з індикаторів переповниться, запис програми до пристрою буде заборонено.

Включити/відключити відображення індикаторів у рядку стану можна у головному меню **Вид**.

Режим OFFLINE

У режимі OFFLINE встановлюється заборона зв'язку з пристроєм.

Для переходу в режим OFFLINE слід натиснути на індикатор моделі пристрою в рядку стану або вибрати головне меню **Сервіс** → **Режим OFFLINE**. Вид індикатора зміниться на **OFFLINE**. Повторне натискання відключає режим OFFLINE.

Докладніше див. у розділі [Завантаження в пристрій 3.7](#).

2.8 Менеджер і редактор екранів

Менеджер екранів

Якщо пристрій оснащений дисплеєм, на нього можна запрограмувати виведення інформації з програми. Інструкцію щодо роботи з екранами наведено в розділі [Програмування дисплея пристрою 3.4](#).

Включити/відключити відображення вкладки **Менеджер екранів** у робочій області можна у головному меню **Вид**.

Менеджер екранів відкривається натисканням на однойменну вкладку у лівому краї робочої області.

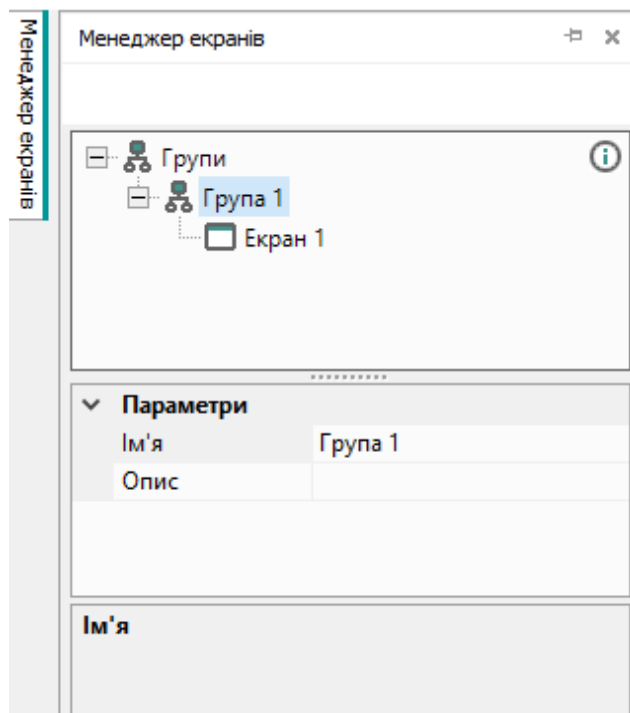


Рисунок 2.13

Менеджер екранів складається з дерева елементів у верхній частині та полів властивостей у нижній частині.

У дереві екранів відображаються та редагуються групи екранів та екрани.

Основні функції:

	Додати екран (знаходиться у верхній частині менеджера екранів та у контекстному меню групи екранів)
	Видалити екран (у контекстному меню екрана)
	Редагувати екран (у контекстному меню екрана)
	Додати групу (знаходиться у верхній частині менеджера екранів)
	Редагувати групу (у контекстному меню групи екранів, доступний тільки для пристроїв з графічним кольоровим РКІ)

Під час вибору екрана або групи екранів у нижній частині **Менеджера екранів** відображаються їхні властивості.

Редактор екранів та групи екранів для пристроїв з монохромним текстовим РКІ

Редактор екранів відкривається вибором команди **Редагувати екран** у контекстному меню екрана в **Менеджері екранів**.

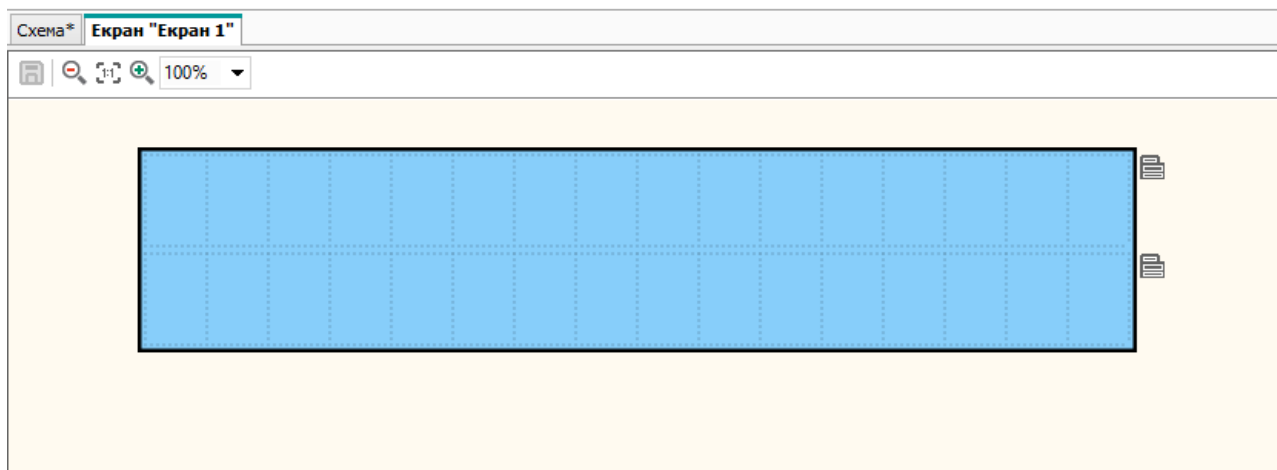


Рисунок 2.14

Для пристроїв з монохромним текстовим РКІ також доступний редактор групи екранів, який відкривається вибором команди **Редагувати групу** в контекстному меню групи екранів в **Менеджері екранів**.

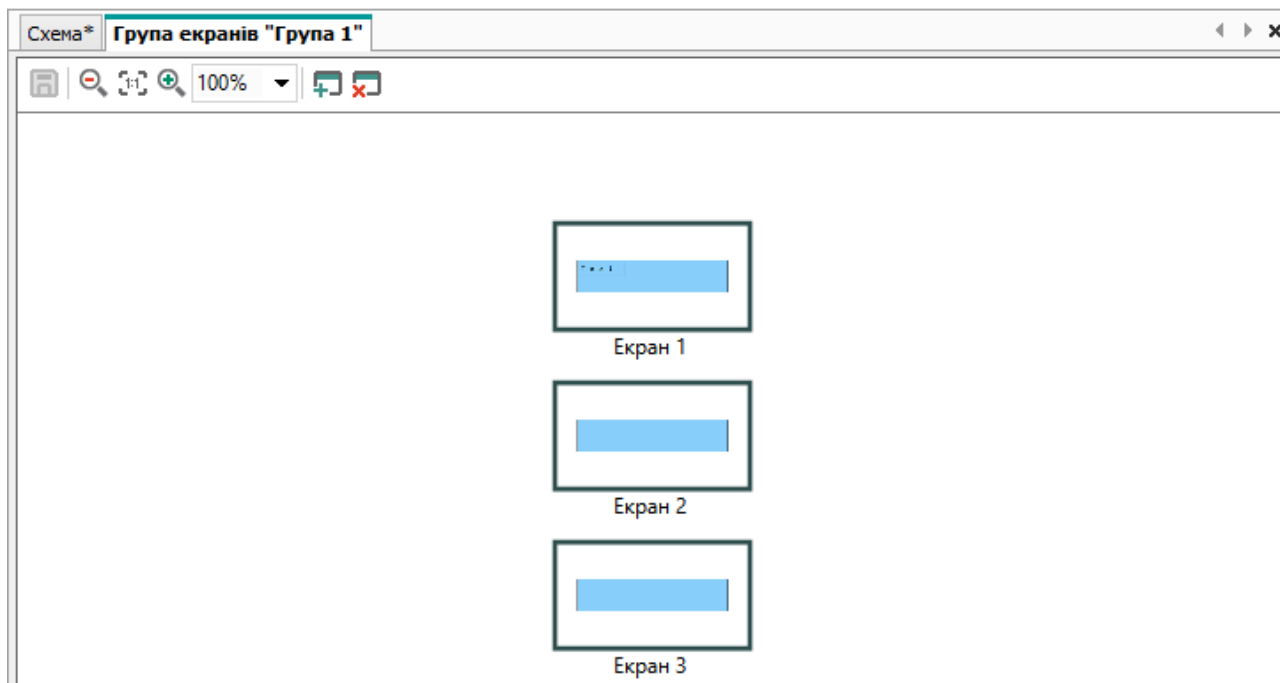






Рисунок 2.15

У верхній частині редактора екранів та групи екранів розташовані кнопки:

	Зберегти документ
	Зменшити масштаб
	Оригінальний розмір
	Збільшити масштаб

Масштаб може бути змінений за допомогою меню, що випадає праворуч від кнопок.

3 Базові відомості

У цьому розділі описуються основні поняття роботи пристрою та принцип створення програми для завантаження у пристрій:

- [Принцип виконання програми 3.1;](#)
- [Розробка проекту та порядок роботи 3.2;](#)
- [Розробка програми 3.3;](#)
- [Програмування дисплея пристрою 3.4;](#)
- [Режим симулятора 3.5;](#)
- [Налаштування порту та підключення пристрою 3.6;](#)
- [Запис програми у пристрій 3.7.](#)
- [Онлайн налагодження 3.8;](#)
- [Відомості про проект 3.9;](#)
- [Менеджер компонентів 3.10;](#)
- [Робота з макросами 3.11;](#)
- [Створення функції мовою ST 3.12.](#)

3.1 Принцип виконання програми

Програма для пристрою складається з урахуванням кількості входів, виходів і наявності годинника реального часу.

Функціональна схема роботи пристрою:

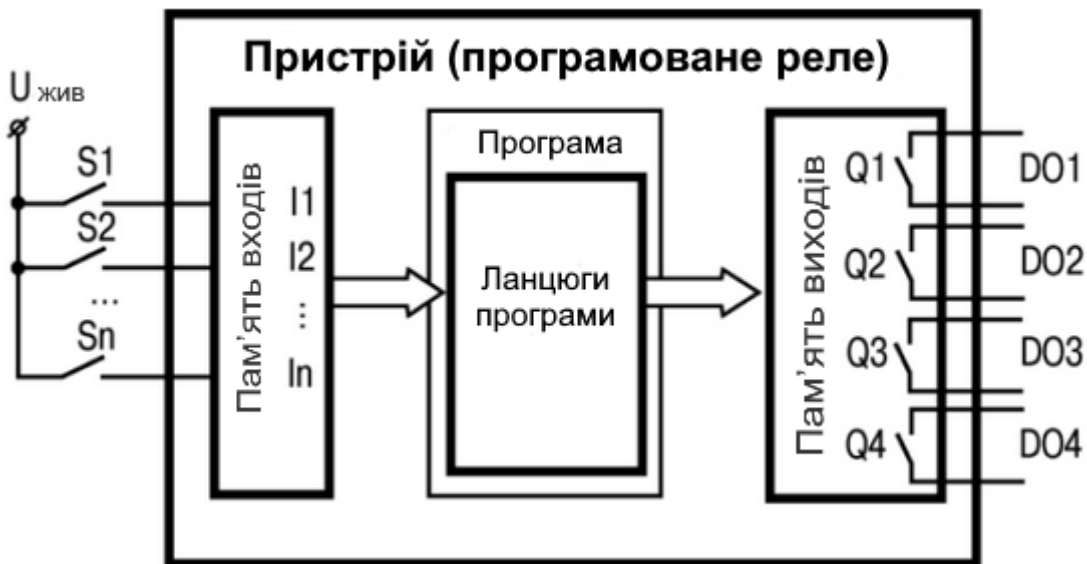


Рисунок 3.1

Роботу пристрою можна подати у вигляді послідовно виконуваних кроків (робочий цикл):


1. Логічний стан входів автоматично записується в комірки пам'яті входів (кількість комірок дорівнює кількості входів – $I1 \dots In$).
2. Програма зчитує значення з комірок пам'яті входів та виконує над ними логічні операції відповідно до алгоритму роботи.
3. Після обробки всієї програми результати записуються на фізичні виходи пристрою (для включення вихідних елементів $Q1 \dots Q4$).
4. Перехід до Кроку 1 (після виконання всіх попередніх кроків обробки програми цикл роботи пристрою повторюється з першого кроку).

Час виконання всіх кроків залежить від складності алгоритму програми (докладніше див. розділ [Час циклу 7.2](#)).

3.2 Розробка проекту та порядок роботи

Створення та редагування проекту

Для створення нового проекту слід:

1. Натиснути кнопку  на панелі інструментів або вибрати у головному меню **Файл** → **Новий проект...**
2. У вікні, що відкрилося, вибрати модифікацію пристрою і натиснути **ОК**. Меню вибору пристрою підтримує фільтрацію за моделлю.

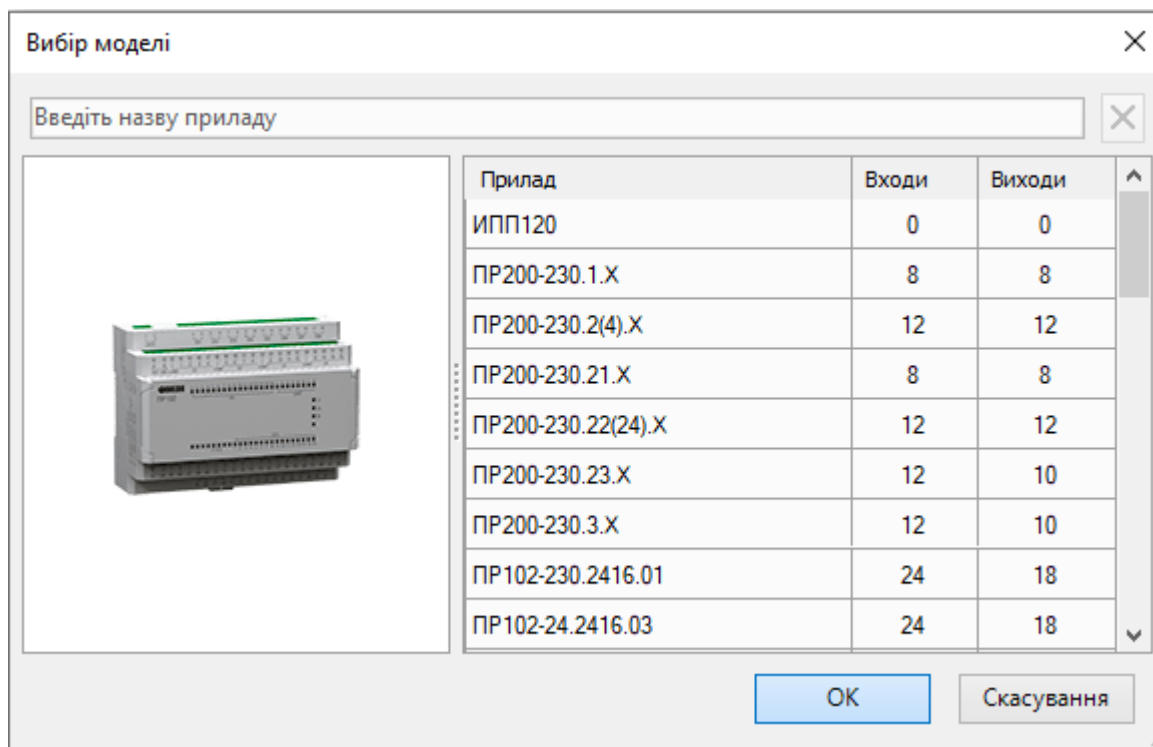


Рисунок 3.2



ПРИМІТКА

Якщо до ПК підключено пристрій, AQLogic запропонує модель підключеного пристрою у вікні вибору.

У головному вікні AQLogic відобразиться полотно проекту, стануть активними [рядок стану 2.7](#), панель [Бібліотека компонентів 2.3](#), [Властивості 2.4](#) та [Змінні 2.5](#).

Якщо у вибраного пристрою є дисплей, у головному вікні з'явиться вкладка [Менеджер екранів 2.8](#), де можна [налаштувати 3.4](#) виведення інформації на дисплей.

Зберегти поточний або відкрити збережений проект можна за допомогою кнопок на панелі інструментів або в головному меню **Файл**.

Порядок програмування пристрою у проекті:

1. Розробка програми.
2. Налаштування програми.
3. Підключення пристрою до ПК.
4. Завантаження розробленої програми у пристрій.

Розробка програми

На полотні проекту створюється [програма 3.3](#) мовою FBD за допомогою блоків та елементів вставки з [Бібліотеки компонентів 6](#).

Налаштування програми

Після створення можна змоделювати роботу програми в [режимі симулятора 3.5](#). Під час перевірки правильності роботи програми користувач може змінювати стан входів, контролюючи стан виходів на

відповідність потрібним умовам. Після виявлення помилок можна повернутися до редагування програми.

Підключення пристрою до ПК

Для підключення пристрою до ПК слід [налаштувати порт 3.6](#) в AQLogic і за допомогою кабелю підключити пристрій. Далі на пристрій подається живлення (якщо потрібно), і відбувається [налаштування приладу 4](#) в AQLogic .

Завантаження програми у пристрій

Після налаштування з'єднання можна [завантажити програму 3.7](#) у пристрій з AQLogic .

Онлайн налагодження

Завантажену програму можна перевірити в режимі [онлайн налагодження 3.8](#) з реальними значеннями, що надходять з пристрою.

3.3 Розробка програми

Розробку програми рекомендується розпочати із планування. План повинен описувати всі можливі стани пристрою під час функціонування (у вигляді діаграми режимів, таблиці станів, електричної чи функціональної схеми та/або ін.).

Після того як продумані всі завдання, які мають виконуватися пристроєм, слід скласти програму на полотні проекту за допомогою елементів [Бібліотеки компонентів 6](#).

Розміщення компонентів та створення зв'язків

Компоненти розміщуються на схемі шляхом вибору потрібного компонента в панелі [Бібліотека компонентів 6](#) та переміщення на полотно методом drag & drop.

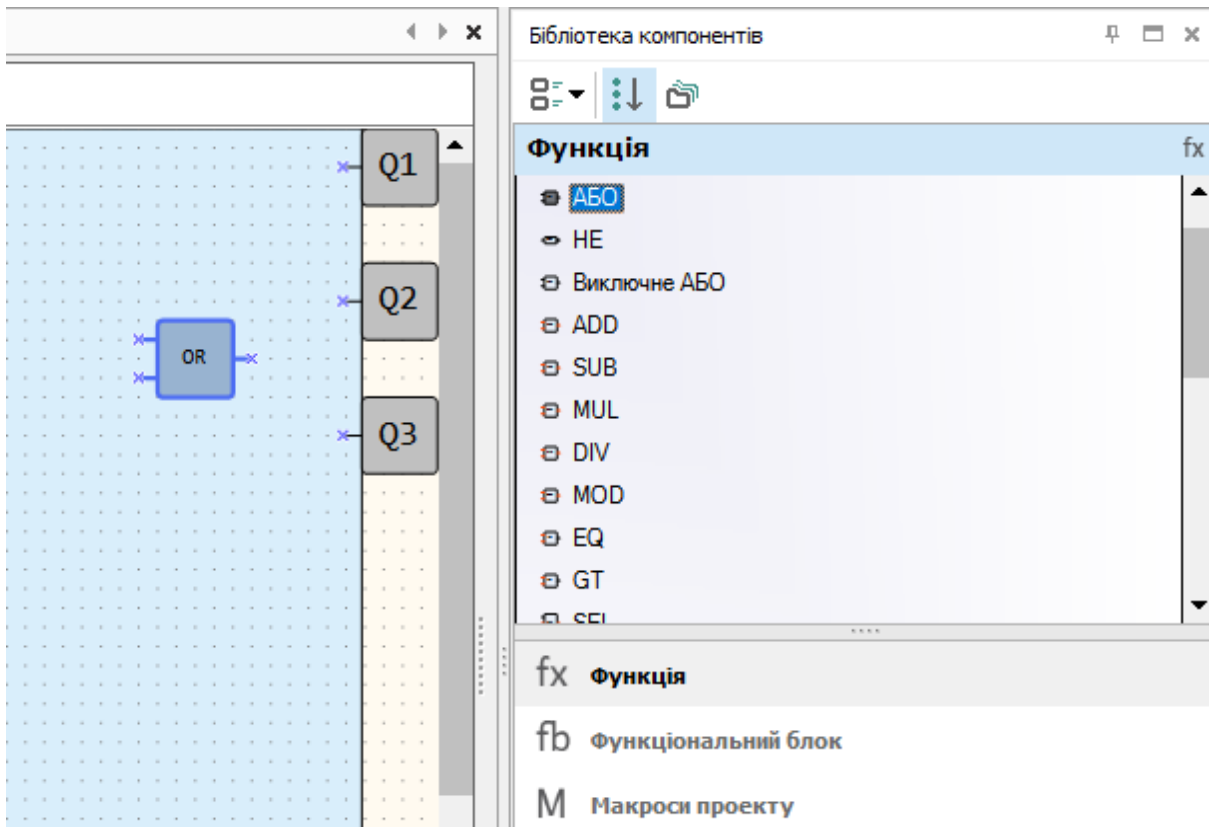


Рисунок 3.3

Входи та виходи пристрою та компонентів слід з'єднати, провівши між ними лінії зв'язку. Для створення сполучної лінії слід:

- натиснути ЛКМ вихід першого компонента. Лінія приєднається до нього і потягнеться за курсором миші;
- для створення згинання лінії слід натиснути на полотні для позначення точки повороту лінії;
- протягнути лінію до входу другого компонента та натиснути по ньому. Лінію зв'язку створено.

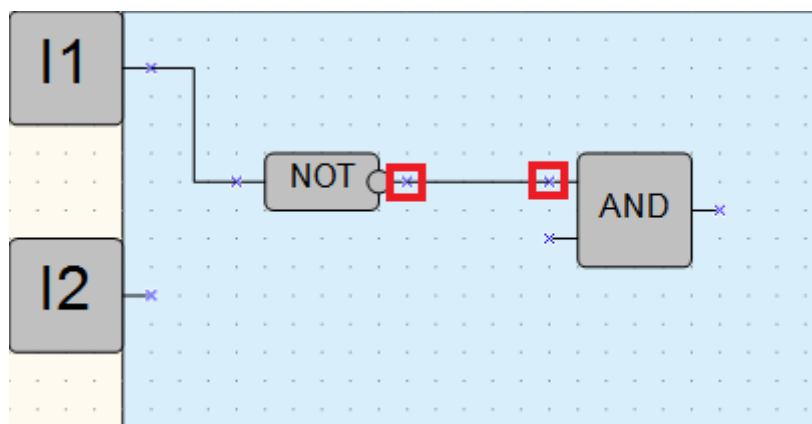


Рисунок 3.4

Якщо типи входу та виходу компонентів не збігаються, лінія не створиться. Докладніше див. розділ [Типи змінних 5.1](#).

Для виділення блоку або компонента на схемі слід натиснути на ньому ЛКМ. Щоб виділити кілька елементів схеми, можна скористатися «ласо» – укласти потрібні елементи усередину прямокутника, що розтягується слідом за курсором миші, утримуючи ЛКМ.

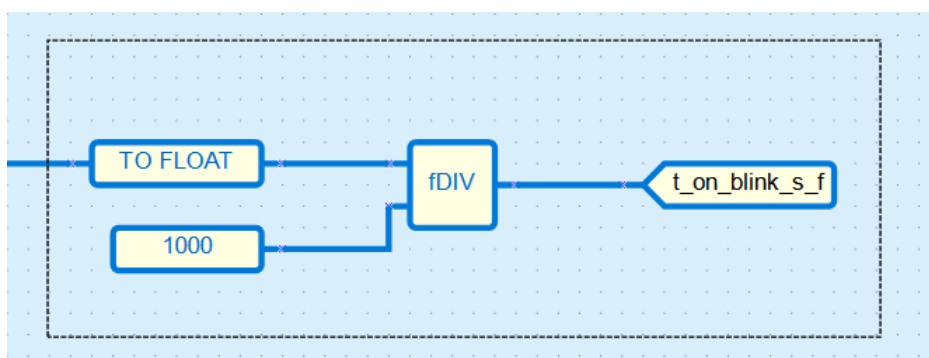


Рисунок 3.5

Налаштування параметрів компонентів

Щоб відредагувати параметри, слід виділити потрібний компонент або блок на схемі та налаштувати значення на панелі [Властивості 2.4](#).

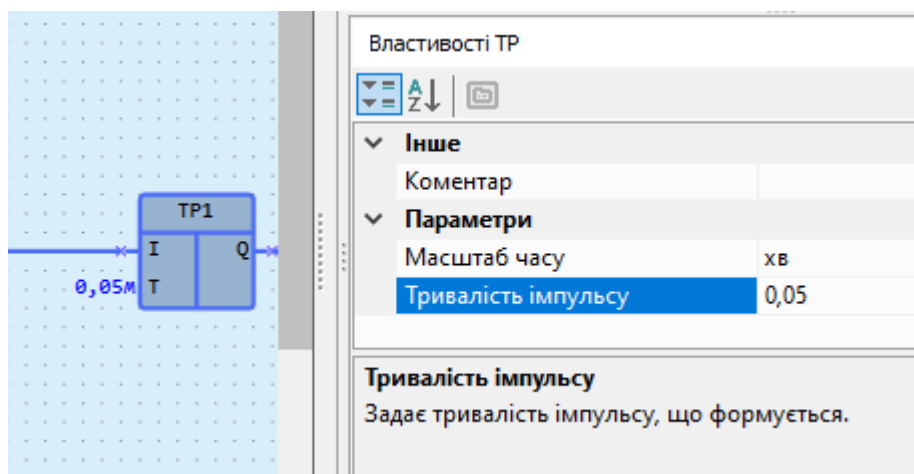


Рисунок 3.6

Властивості компонента або блоку можна відкрити за допомогою контекстного меню.

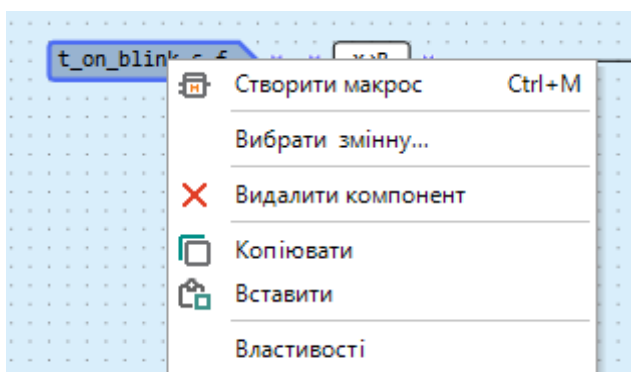



Рисунок 3.7

Для розробки програми використовуються такі блоки та функції, що викликаються на панелі вставки:

Блок коментарів 3.3.1	Розміщення текстового коментаря на схемі
Блок змінної 3.3.2	Розміщення блоку змінної для запису чи читання значень програми
Блок константи 3.3.3	Розміщення блоку з фіксованим числовим значенням
Лінія затримки 3.3.4	Створення затримки на цикл передання значення з виходу компонента на вхід
Блок мережевої змінної 3.3.5	Розміщення блоків обміну даними з іншими приладами, підключеними до пристрою через мережу
Блок читання/запису у ФБ 3.3.6	Запис/читання значень окремих параметрів із ФБ до змінної та навпаки
Блок перетворення 3.3.7	Конвертація значень різних типів для передання
Перенумерувати компоненти 3.3.8	Перепризначення порядкових номерів ФБ схеми
Порядок виконання 3.3.9	Зміна порядку обчислення значень виходів програми

3.3.1 Блок коментарів

Блок коментарів дозволяє залишати записи та коментарі до схеми проекту.

Щоб додати текстовий коментар до схеми, слід натиснути кнопку  на панелі вставки. Потім утримуючи ЛКМ накреслити прямокутну ділянку блоку коментарів на тому місці схеми, де має розташовуватися коментар.

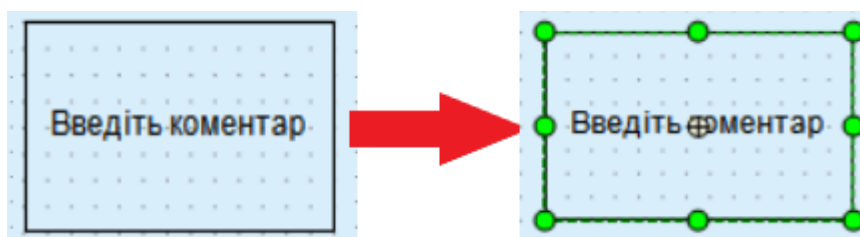


Рисунок 3.8

Для виділеного блоку коментарів можна змінити налаштування введення тексту на панелі властивостей.

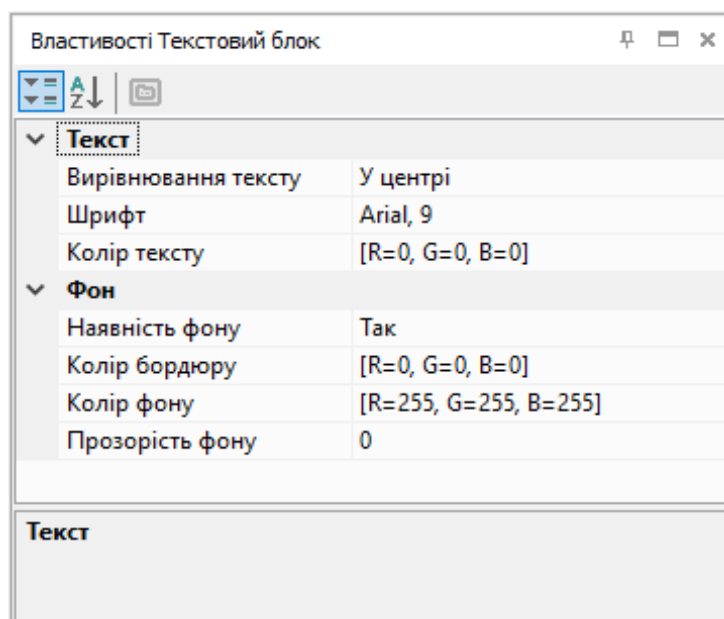


Рисунок 3.9

Щоб колір фону блоку коментарів було видно, слід встановити значення параметру **Прозорість фону** більше 20.

Для введення тексту слід двічі натиснути ЛКМ на блок коментарів та ввести новий текст.

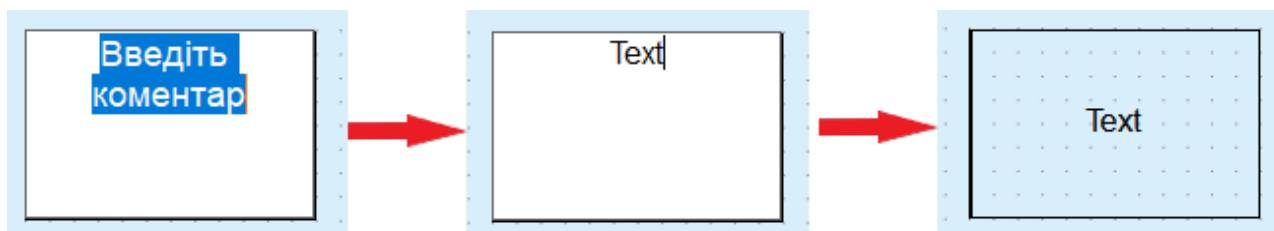




Рисунок 3.10

3.3.2 Блок змінної

Блок змінної слугує для використання змінної на схемі проекту.

Щоб додати блок змінної на схему, слід натиснути на його піктограму на панелі вставки:

-  **вхідна змінна** – для передання значення у програму;
-  **вихідна змінна** – для запису в неї значення з програми.

Потім натиснути на схемі у те місце, куди потрібно помістити блок.

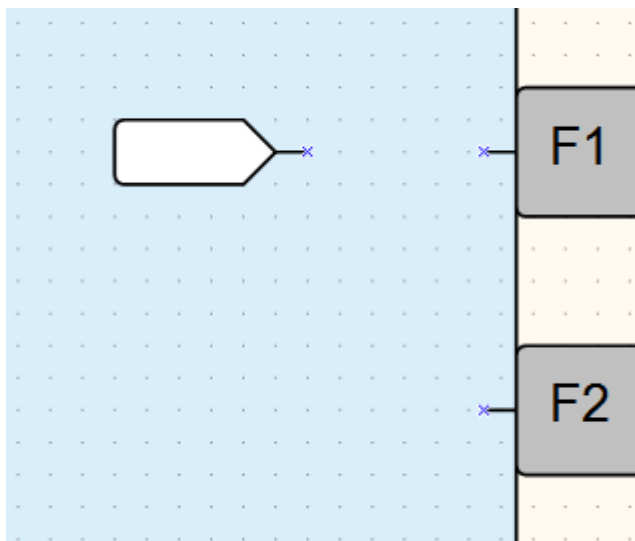


Рисунок 3.11

Блок змінної можна також додати на схему з панелі [Змінні 2.5](#). Для прив'язки змінної до блоку змінної слід:

1. Виділити блок змінної.
2. У властивостях блоку змінної натиснути кнопку «...» в полі **Змінна** або двічі натиснути на блок змінної на схемі.

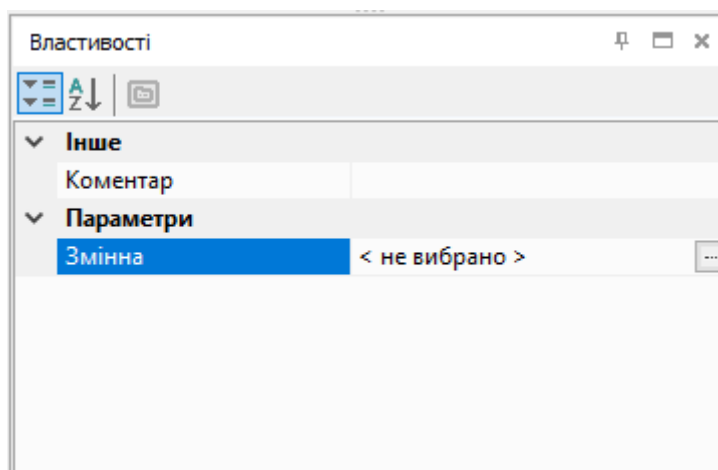


Рисунок 3.12

У вікні [Таблиці змінних 5](#), що відкрилося, вибрати або створити нову змінну і натиснути кнопку **ОК**. Вибрана змінна буде прив'язана до блоку.

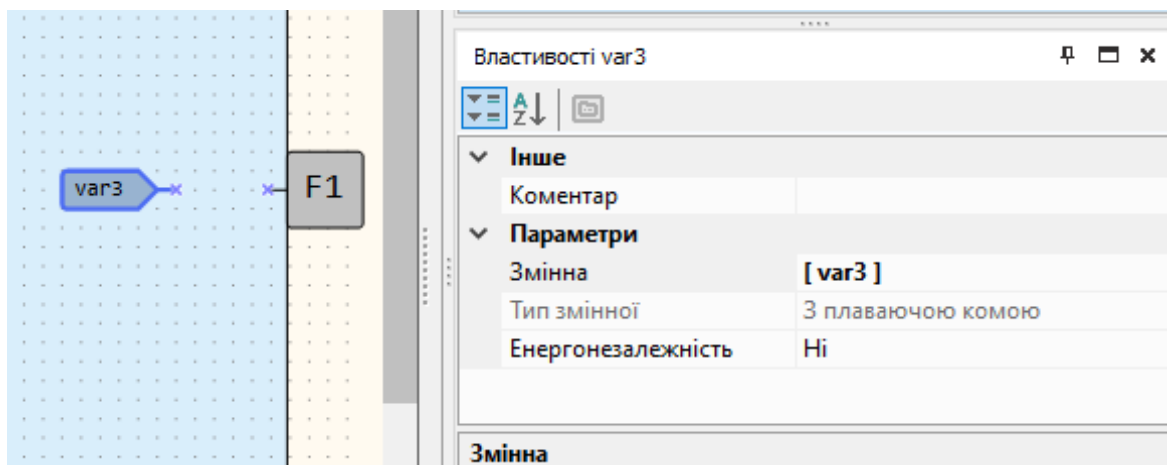


Рисунок 3.13

Якщо після прив'язки блок змінної виділяється червоним, значить, присутня помилка. Повідомлення про помилку відображається у рядку стану.

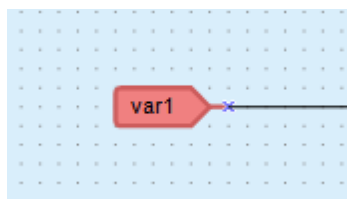


Рисунок 3.14

Для блоку вихідної змінної доступне налаштування **Запис наприкінці циклу**, яке слугує для запису значення у змінну після всіх операцій читання в [робочому циклі 3.1](#).

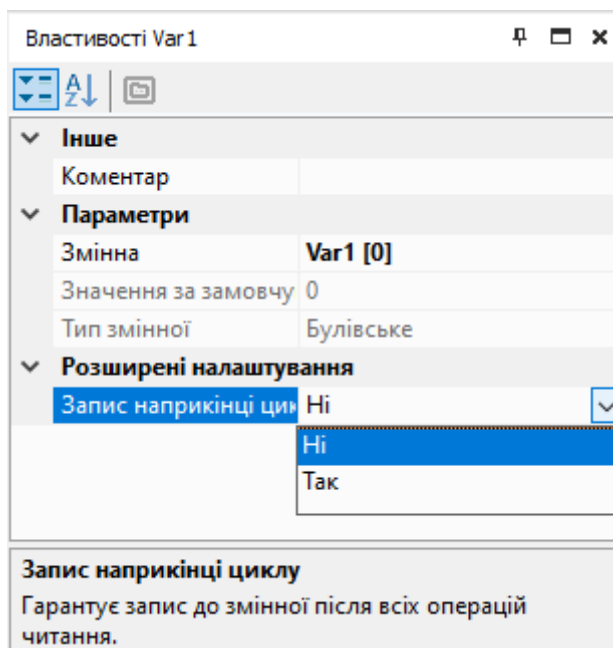



Рисунок 3.15

3.3.3 Блок константи

Для включення до програми жорстко заданого числового значення використовується блок константи. Для розміщення блоку константи на схемі слід натиснути кнопку  на панелі вставки, потім – у місце на схемі, де має бути розташований блок.

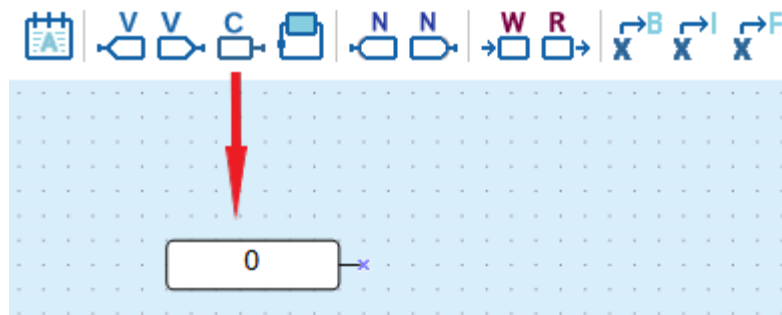


Рисунок 3.16

На панелі властивостей вибраного блоку вказується тип даних для константи та вводиться значення. Значення буде відображено у блоці на схемі.

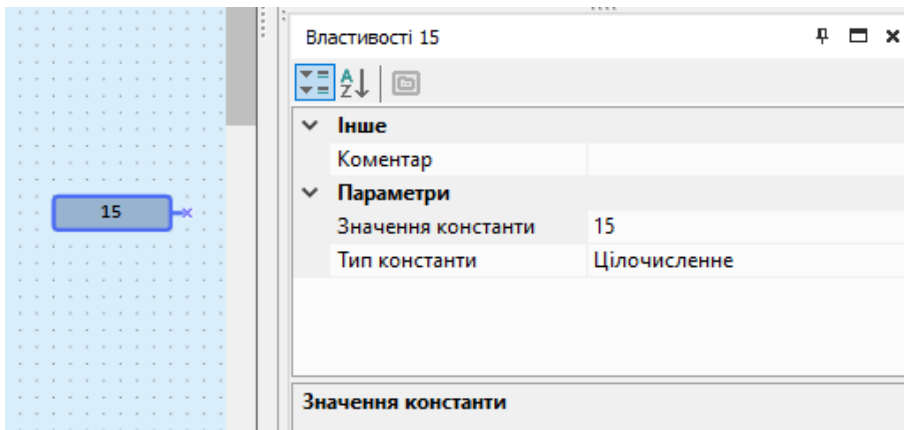


Рисунок 3.17

Значення константи можна відредагувати подвійним натисканням на блок константи або вибором **Змінити значення** у контекстному меню блоку.

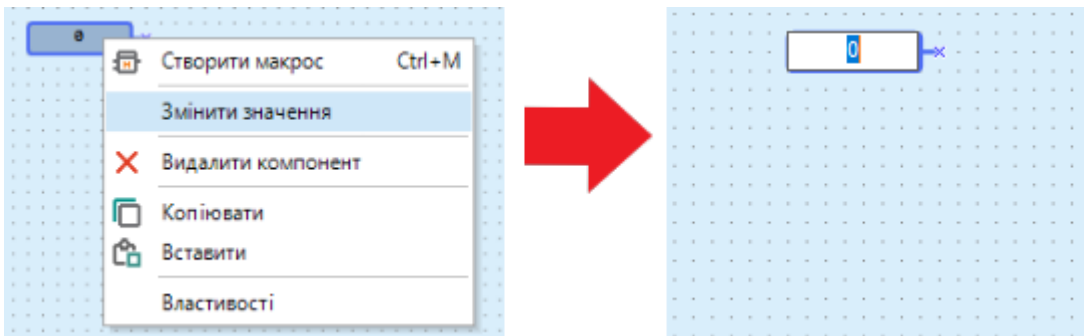


Рисунок 3.18

Допустимі значення для типів константи:

Булевське	від 0 до 1
Цілочисельне	від 0 до 4294967295
З рухомою комою	від $-3,402823E+38$ до $3,402823E+38$

Значення константи не зазнає змін протягом роботи програми.

3.3.4 Лінія затримки

Лінія затримки використовується для створення затримки на один цикл передачі значення з виходу компонента на вхід.


Для створення лінії затримки слід натиснути кнопку  на панелі вставки і побудувати лінію між виходом і входом компонента. Буде проведено червону пунктирну лінію.



Рисунок 3.19

Приклад

На вхід **I1** блоку арифметичного складання (цілочисельний тип) подається константа, значення якої дорівнює 1. На вхід **I2** по лінії затримки подається значення з виходу блоку **Q**, яке було обчислено у попередньому циклі.

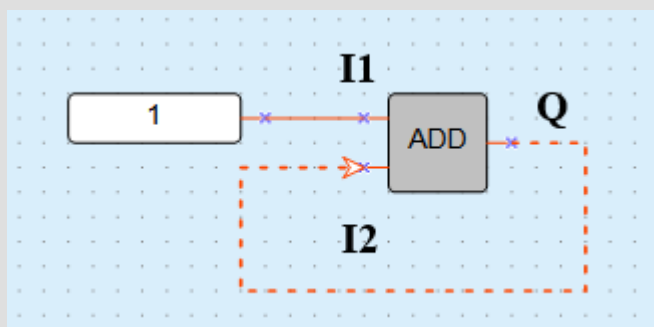


Рисунок 3.20



Значення на входах та виходах блоку:

№ циклу	1	2	3	4	5	6	7	8	9	10
I2	0	0	1	1	2	2	3	3	4	4
Q	1	1	2	2	3	3	4	4	5	5

3.3.5 Блок мережевої змінної

Блоки мережевих змінних слугують для передання та отримання значень за інтерфейсами зв'язку та використання їх в програмі.

Для додавання на схему блоку мережевої змінної слід натиснути на кнопку:

-  – передання значення із програми по мережі;
-  – отримання значення у програму по мережі.

Прив'язка змінної до блоку мережевої змінної

Для прив'язки змінної до блоку мережевої змінної слід:

1. Вибрати на схемі блок мережевої змінної.

- Натиснути на кнопку «...» у полі **Змінна** у властивостях блоку мережевої змінної або двічі натиснути на блок мережевої змінної на схемі.

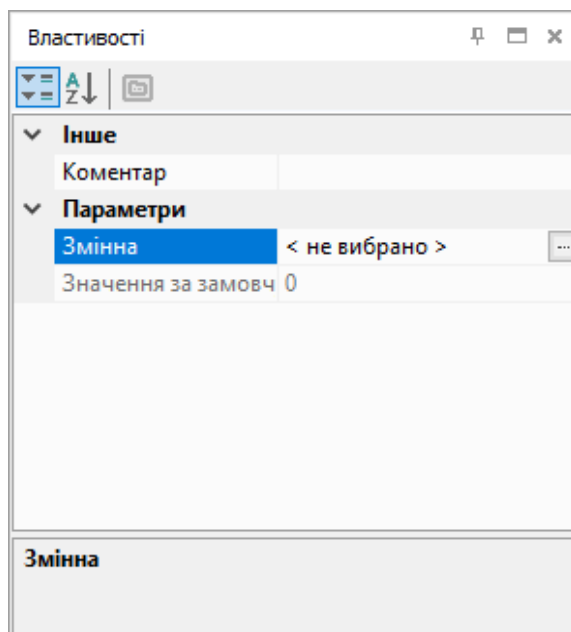


Рисунок 3.21

- У [таблиці змінних 5](#), що відкрилася, вибрати або створити нову змінну для прив'язки і натиснути кнопку **ОК**.

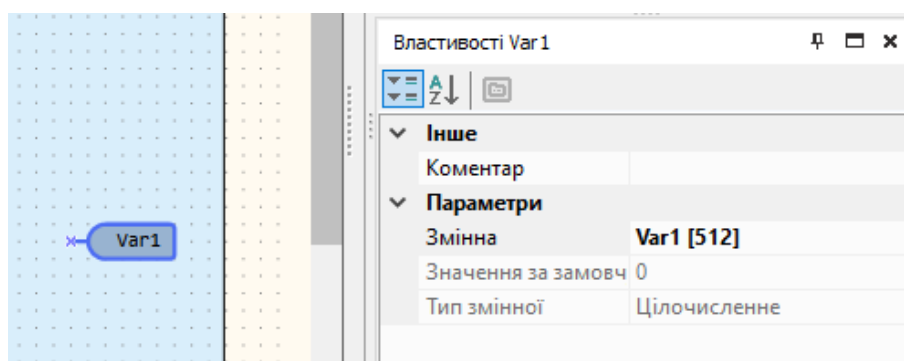


Рисунок 3.22

- Приєднати вихід/вхід блоку мережевої змінної до необхідного компонента.

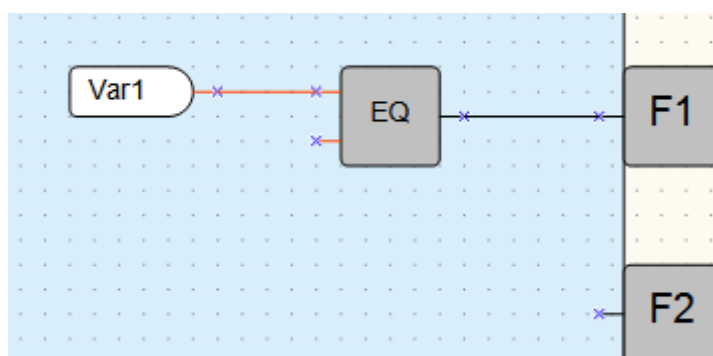


Рисунок 3.23

Вищеописаний алгоритм роботи зі змінними інтерфейсів зв'язку зручний, якщо користувач починає роботу з проектом з налаштування пристрою. Якщо список параметрів Майстра мережі Modbus вже відомий, роботу зручніше починати з визначення змінних у [таблиці змінних 5](#). Докладніше див. розділ [Робота за протоколом Modbus 4.3.2.1](#).



**ПРИМІТКА**

Не можна прив'язати змінну до блоку, якщо в [налаштуваннях пристрою 4](#) немає інтерфейсів зв'язку.

3.3.6 Блок читання/запису у ФБ

Блок читання/запису у ФБ використовується для задання чи читання значення параметра ФБ у процесі виконання програми.

На схему можуть бути додані блоки:

-  – запису значення у ФБ;
-  – читання значення з ФБ.

Приклад

Залежно від значення на вході **I1** у функціональному блоці **BLINK1** значення параметра **Тривалість вимк. стану** матиме значення **2** або **10**.

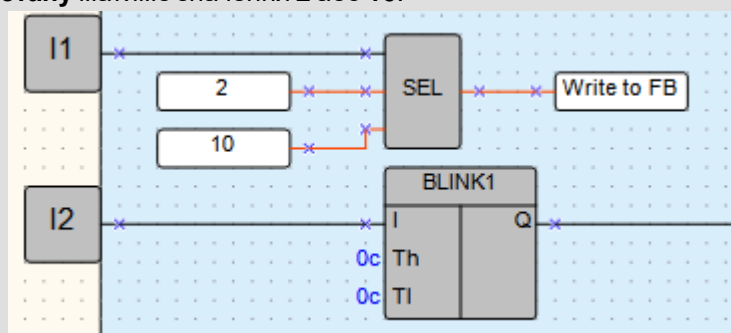


Рисунок 3.24

Для налаштування блоку запису значення у ФБ слід на панелі властивостей блоку вибрати ФБ та ім'я змінної, в яку буде записуватися значення.

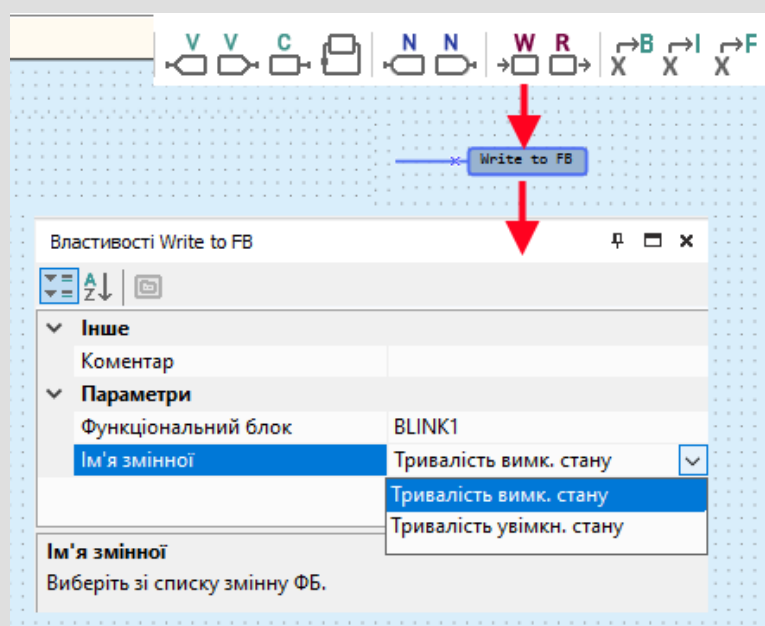


Рисунок 3.25

3.3.7 Блок перетворення

Лінія зв'язку між компонентами програми може бути створена тільки для входу та виходу одного й того ж типу: булевського, цілочисельного чи з рухомою точкою. Для створення лінії зв'язку між входом та виходом різних типів слід використовувати блоки перетворення.

Щоб додати блок перетворення на схему, слід вибрати потрібний тип блоку на панелі вставки і потім натиснути на те місце схеми, де він повинен бути розташований.

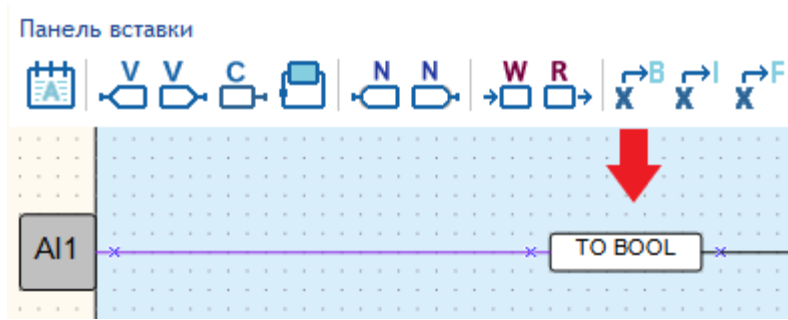



Рисунок 3.26

Типи блоків перетворення:

	<p>Перетворювач на булевське значення</p>	<p>Конвертація значення сигналу на булевське значення. Якщо значення на вході блоку більше 0 (False), то на виході блоку буде 1 (True).</p>
	<p>Перетворювач на цілочисельне значення</p>	<p>Конвертація значення сигналу на цілочисельне значення. Якщо на вході блоку булевське значення 0 (False), то на виході буде 0, якщо на вході 1 (True), то на виході буде 1. Якщо на вході блоку значення числа з рухомою комою, то на виході воно округляється з нестачею. Від'ємні значення з рухомою комою конвертуються в 0.</p>
	<p>Перетворювач на число з рухомою комою</p>	<p>Конвертація значення сигналу в число з рухомою точкою. Для перетворення цілочисельного в число з рухомою точкою додається дробова частина (int = 5 → float = 5,0000). Якщо на вході блоку булевське значення 0 (False), то на виході буде 0,0000, якщо на вході блоку 1 (True), то на виході буде 1,0000.</p>

3.3.8 Перенумерувати компоненти

Щоб перевизначити порядкові номери компонентів на схемі, натисніть кнопку  на панелі інструментів або виберіть у головному меню **Сервіс** → **Перенумерувати компоненти**. Однотипні компоненти нумеруються по порядку зверху донизу та зліва направо.

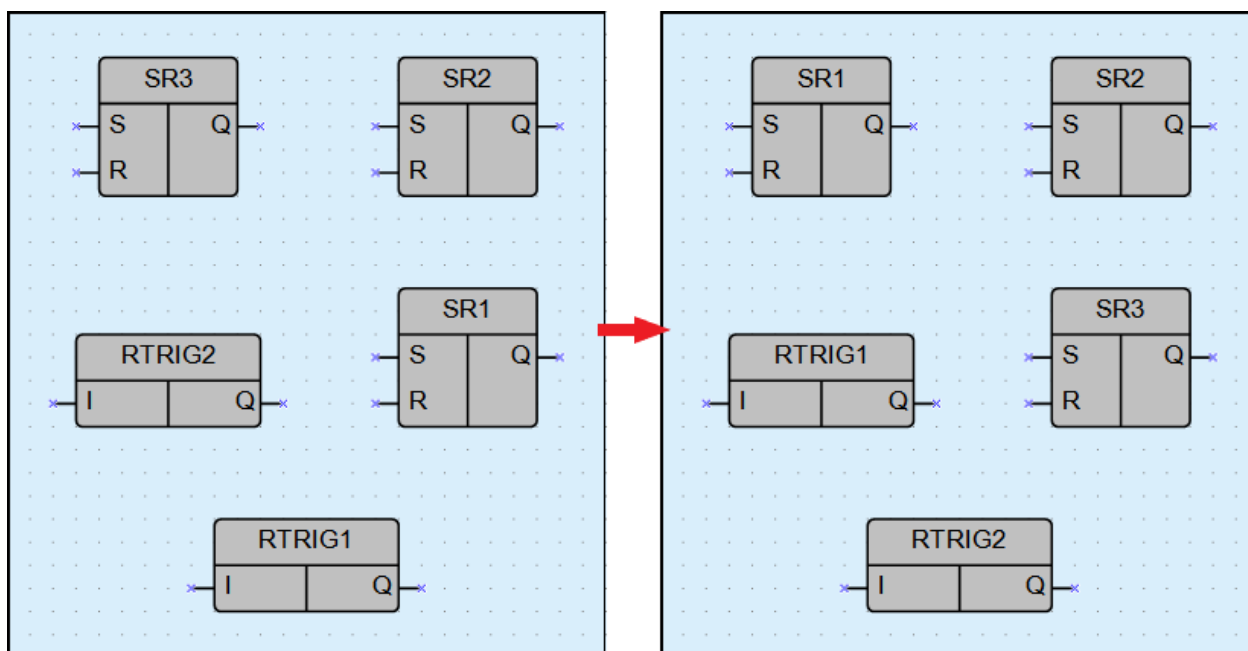



Рисунок 3.27

3.3.9 Порядок виконання

Значення всіх виходів та зворотних зв'язків у програмі обчислюються у певному порядку. Щоб змінити порядок обчислення, слід натиснути кнопку  на панелі налагодження та у спливному меню вибрати **Зворотні зв'язки** або **Виходи**.

AQLogic перейде в режим налаштування порядку виконання – на виходах та зворотних зв'язках відобразяться порядкові номери черги виконання.

Для зміни черговості виконання слід двічі натиснути ЛКМ на потрібному виході або зворотному зв'язку і ввести число, яким за порядком має виконуватися ця операція.

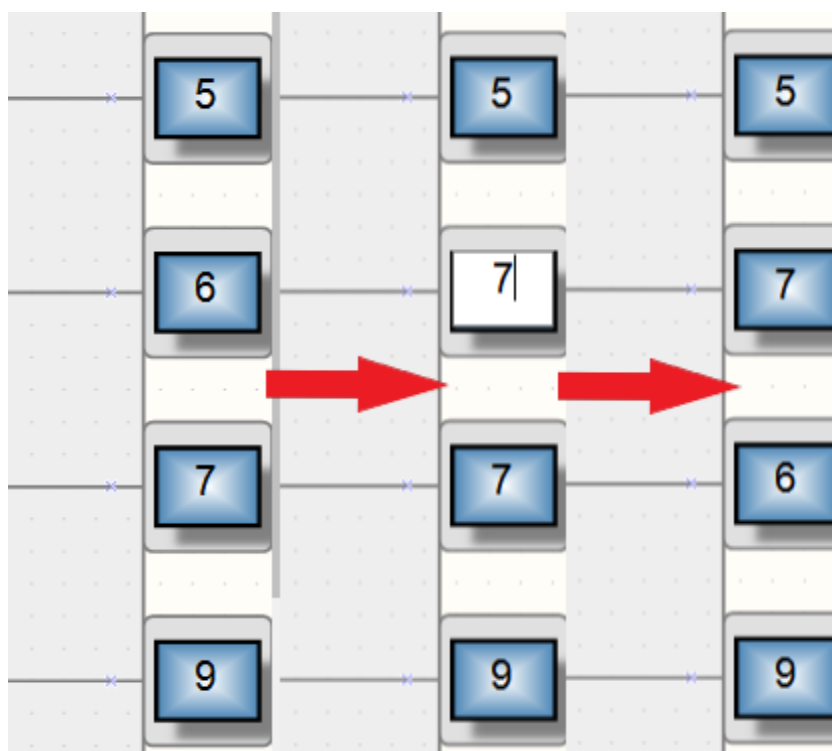



Рисунок 3.28

Після встановлення необхідної черговості слід включити режим налаштування порядку виконання повторним натисканням кнопки .

3.4 Програмування дисплея пристрою

Якщо пристрій оснащено дисплеєм, то можна запрограмувати виведення на нього інформації програми за певним алгоритмом – тобто створити **екран**. На екрані може відображатися текст та значення змінних програми.

Для програмування екрана використовується [Менеджер екранів 2.8](#). Щоб відкрити менеджер екранів, слід натиснути на заголовок вкладки у лівій частині робочої області.

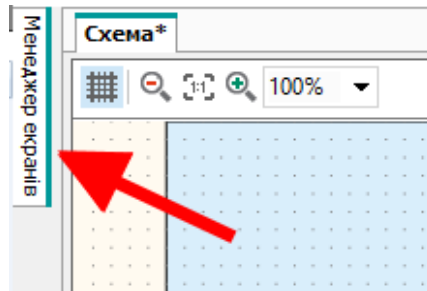


Рисунок 3.29

За умовчанням у Менеджері екранів відображається один екран.

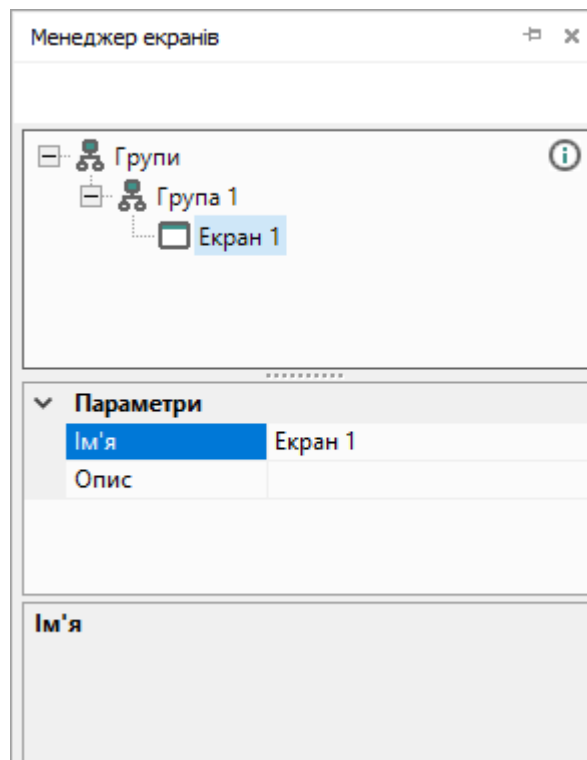


Рисунок 3.30

Додавання екранів

У менеджері екранів можна налаштувати кілька екранів, переключення між якими на дисплеї пристрою здійснюватиметься натисканням кнопки або за подією.

Щоб додати екран, слід натиснути ПКМ на елемент **Група 1** і в контекстному меню вибрати **Додати екран**.

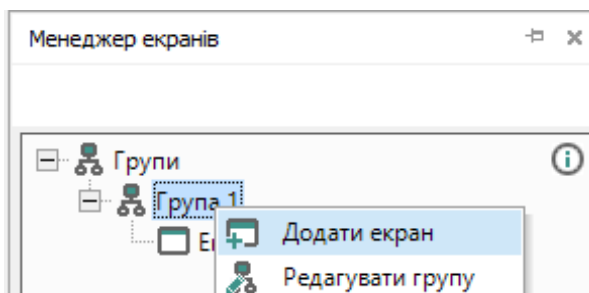


Рисунок 3.31

Для видалення екрана слід натиснути ПКМ на потрібний екран і в контекстному меню вибрати **Видалити**.

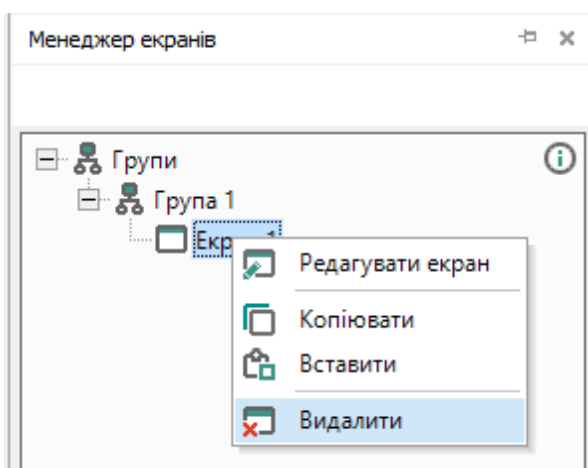


Рисунок 3.32

Для зміни позиції екрана його слід перетягнути, утримуючи клавішу **Shift**, на нове місце.

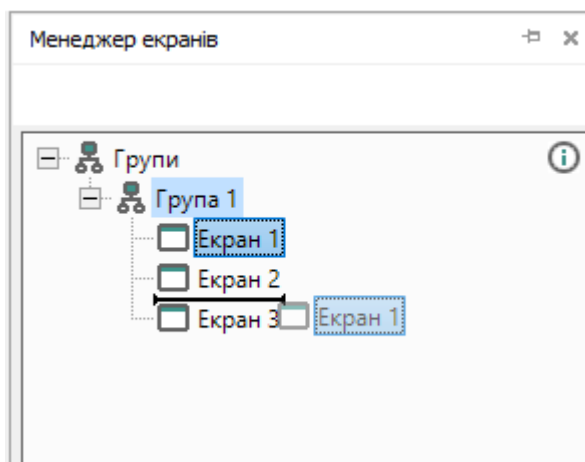


Рисунок 3.33

При перенесенні нове положення відобразиться у вигляді горизонтального маркера.

Для пристроїв з графічним кольоровим РКІ доступне створення груп екранів. У властивостях групи екранів задається ім'я та опис.

Властивості екрана

Для відображення властивостей екрана на панелі властивостей слід двічі натиснути на потрібний екран менеджера екранів.

Властивості екрана:

- **Ім'я** – для відображення в менеджері екранів та в заголовку редактора екранів;
- **Опис** – текстовий опис екрану.

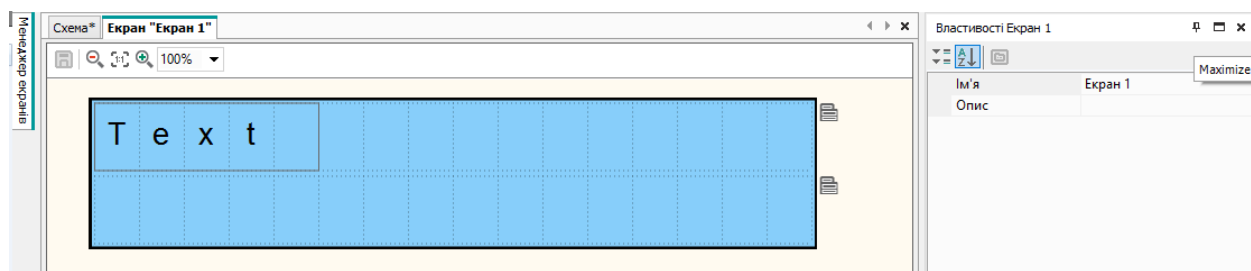


Рисунок 3.34

Перехід до екрана

Якщо у проекті беруть участь два або більше екранів, необхідно налаштувати переходи (умови переключення) між ними.

Для створення переходу слід:

1. Натиснути ПКМ по елементу **Група 1** у дереві менеджера екранів та вибрати **Редагувати групу** в контекстному меню. Відкриється вкладка редактора групи екранів.
2. Вибрати стартовий екран у редакторі групи екранів.
3. У властивостях екрана натиснути «...» у параметрі **Перехід до екрану**. Відкриється меню створення переходу.
4. У випадковому меню **Список екранів** вибрати інший екран для переходу до нього.

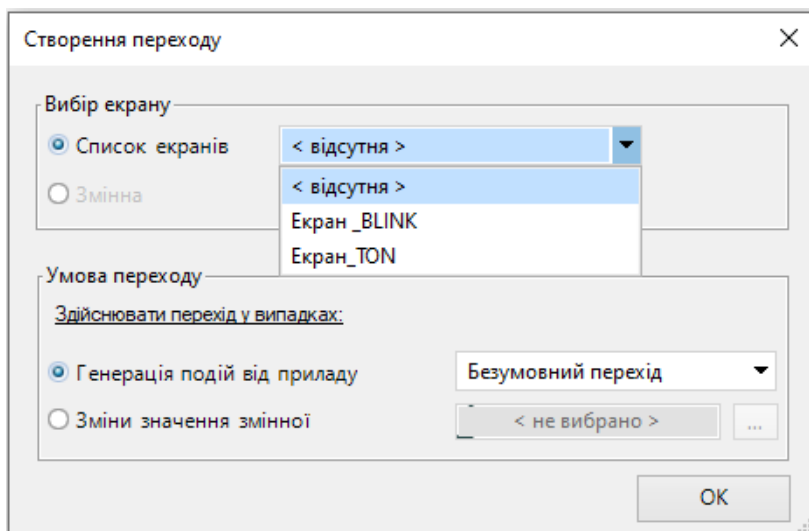


Рисунок 3.35

5. У полі **Умова переходу** вибрати пункт **Генерація подій від приладу** і у випадному меню вказати кнопку пристрою для виконання переходу або вибрати пункт **Зміни значення змінної** для вказівки змінної булевського типу, у разі зміни значення якої буде виконано перехід.

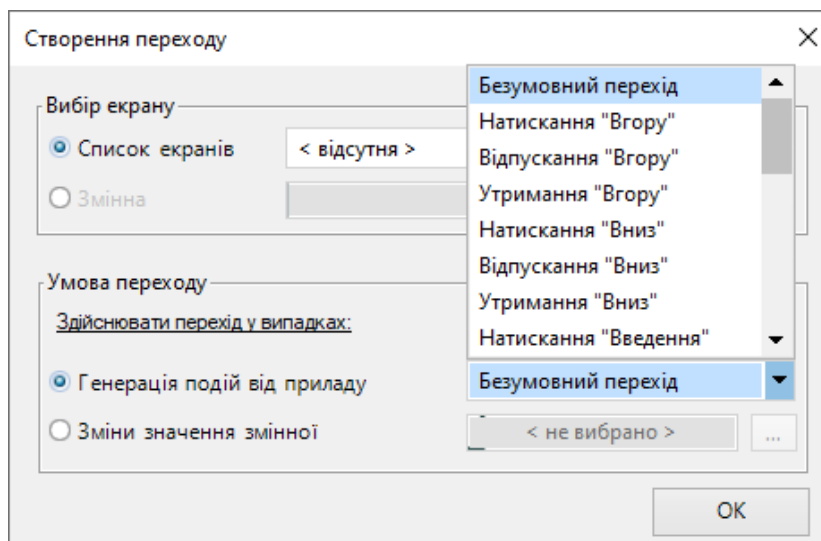


Рисунок 3.36

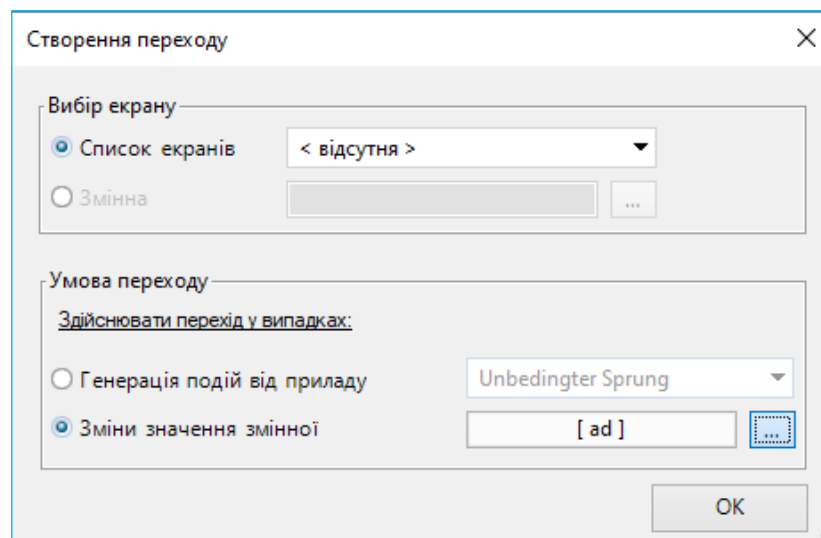


Рисунок 3.37

**ПРИМІТКА**

Безумовний перехід слугує для налаштування переходу, якщо жодна з подій або змін значення змінної не відбувається, наприклад, для переходу на екран аварій.

6. Після створення перехід буде відображено на схемі в редакторі групи екранів та властивостях екрану.

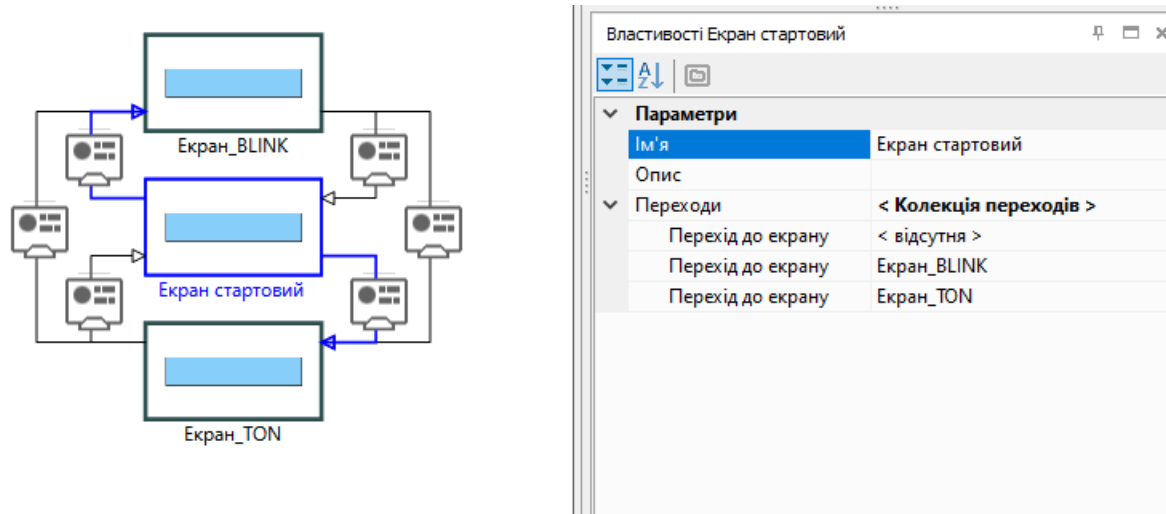


Рисунок 3.38

Перехід від екрана до екрана може відбуватися за різними подіями, для цього можна налаштувати кілька (**колекцію**) переходів. Переходи додаються аналогічно до опису вище.

Для редагування або видалення переходу слід вибрати його у властивостях екрану та налаштувати відповідні параметри. Перехід буде видалено, якщо в полі **Список екранів** меню створення переходу стоятиме значення **< відсутня >**.

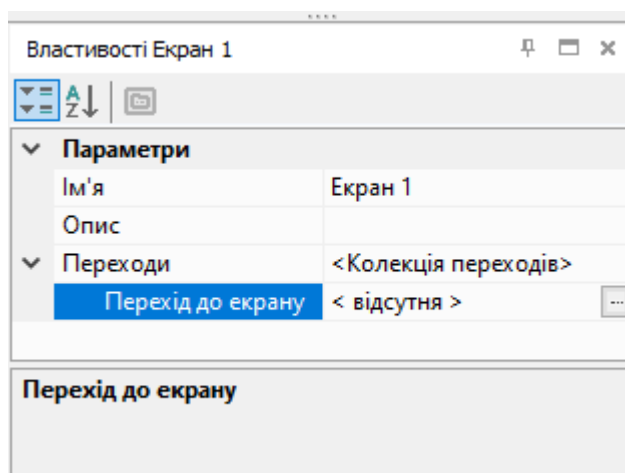



Рисунок 3.39

Редактор екранів

Для налаштування виведення інформації на екран слід відкрити редактор екрана подвійним натисканням по екрану менеджера екранів або натисканням на кнопку  **Редагувати екран** з контекстного меню екрана.

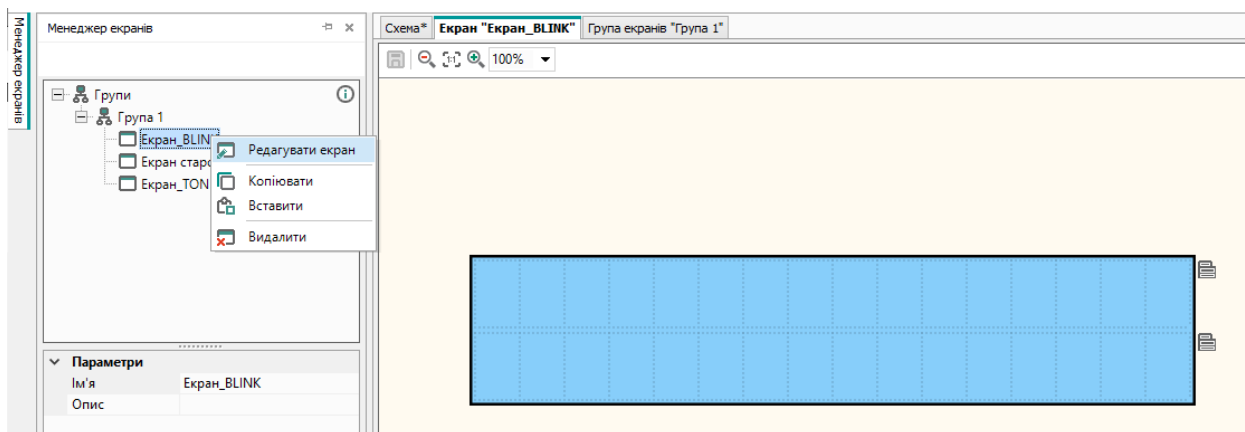



Рисунок 3.40

Екран може містити кілька рядків, перехід до яких здійснюється за допомогою кнопок пристрою. Для додавання або видалення рядків слід натиснути кнопку  **Контекстне меню рядка** і вибрати необхідну команду.

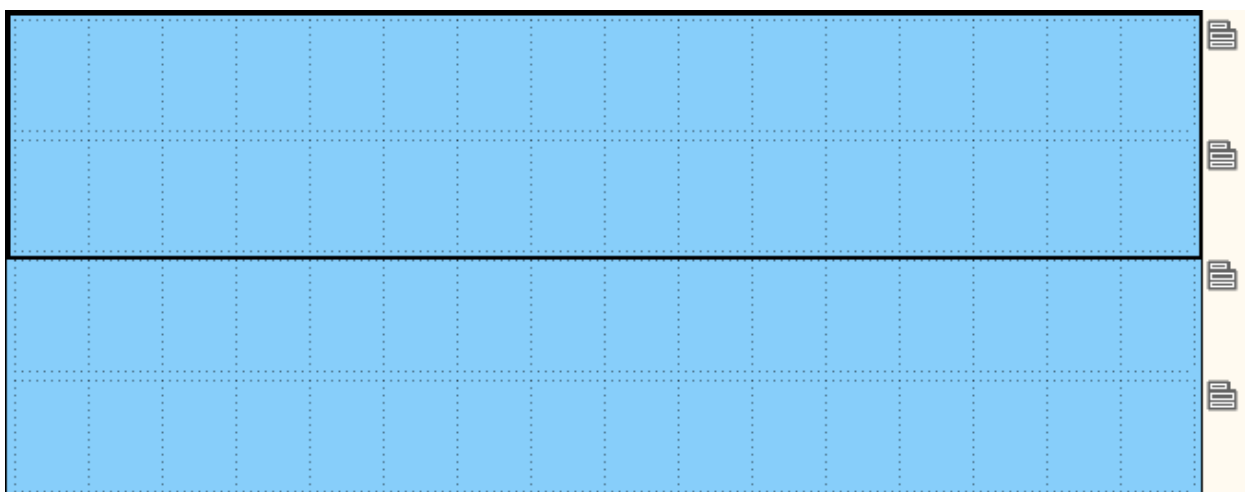


Рисунок 3.41

У рядках розміщуються елементи керування, які перетягуються з панелі **Бібліотека компонентів** методом drag&drop. Опис блоків наведено в розділі [Елементи керування 6.5](#).

**ПРИМІТКА**

Набір символів реалізовано в рамках кодування Windows-1251.

Копіювання екранів

У менеджері екранів можна копіювати екрани для вставлення в поточний або інший проект. Для копіювання виділених екранів слід вибрати команду **Копіювати** в контекстному меню екрана або групи екранів чи натиснути клавіші **Ctrl + C**. Кілька екранів можна виділити за допомогою клавіші **Ctrl** або **Shift**.

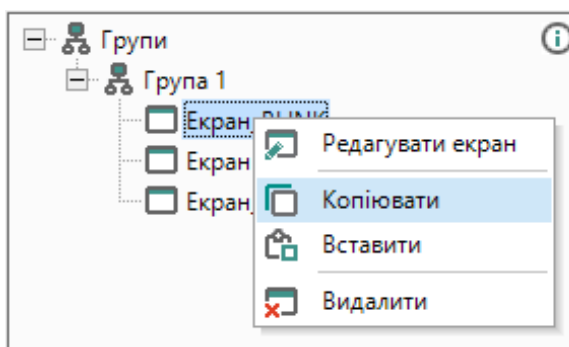


Рисунок 3.42


Для вставки скопійованих екранів слід вибрати в контекстному меню екрана або групи екранів команду **Вставити** чи натиснути клавіші **Ctrl + V**.

Разом з екраном копіюються всі розміщені на екрані елементи керування та властивості екрану. Прив'язані до екрану змінні копіюються в інший проєкт згідно з розділом [Копіювання змінних 5.4](#).

Якщо між виділеними екранами є переходи, вони копіюються разом з екранами. Якщо екран має перехід за межі виділеної області, то такий перехід видаляється.






3.5 Режим симулятора

Коректність роботи створеної програми перевіряється за допомогою режиму симулятора, в якому моделюється зміна станів виходів залежно від зміни станів входів. Симуляція дозволяє проаналізувати стан сигналів усередині програми.

Щоб перейти в режим симуляції, натисніть кнопку  на панелі інструментів або виберіть у головному меню **Сервіс** → **Режим симулятора**. Відкриється панель симуляції.

Панель симуляції



	Пуск	Запуск симуляції у реальному часі
	Крок на один цикл	Покрокова симуляція, виконує 1 цикл програми
	Пауза	Пауза симуляції, повторне натискання продовжує симуляцію
	Зупинка симулятора	Зупинка симуляції, переводить режим симуляції в початковий стан
	Період, мс	Поле введення для задавання періоду поновлення інформації на схемі в мілісекундах
	Час циклу	Поля введення для задавання часу циклу виконання програми у режимі симулятора та вибору одиниць вимірювання часу циклу
	Вікно перегляду	Відкрити/закрити вікно для перегляду значень змінних на кожному кроці виконання програми



УВАГА

Час циклу в режимі симуляції не той самий, що **час циклу 7.2**, який розраховується безпосередньо в пристрої. **Час циклу** може бути вибраний довільно і потрібний для симуляції роботи часових ФБ – [TON 6.2.2.2](#), [TOF 6.2.2.3](#), [BLINK 6.2.3.1](#). Усі уставки часових ФБ будуть відпрацьовані у проміжних кроках симулятора.



Симулювання роботи блоків годинника

Для пристроїв з функцією годинника реального часу відкривається додаткова панель, за допомогою якої можна симулювати поведінку ФБ [CLOCK 6.2.2.4](#) та [CLOCK WEEK 6.2.2.5](#) у часі.

Дата і час:

Послідовність та правила роботи в режимі симуляції

Порядок симуляції програми:

1. Запуск симулятора в одному з режимів: реального часу () або в покроковому () .

2. Задання значень вхідних сигналів на блоках програми:

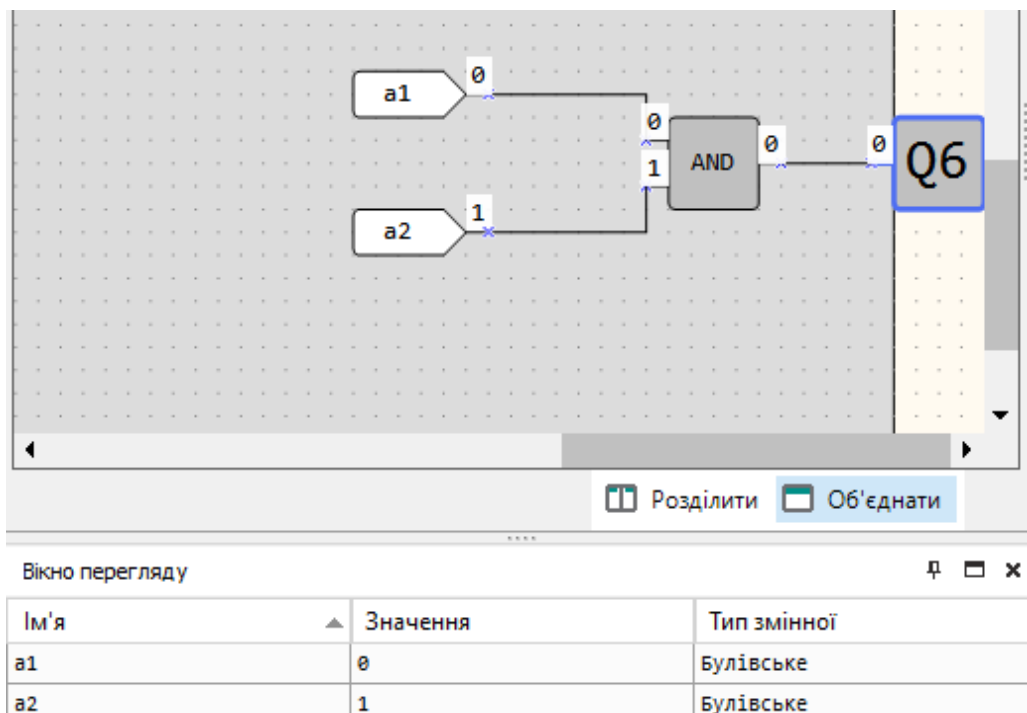


Рисунок 3.43


3. Підбір значень параметрів **Період, мс** та **Час циклу** для зручності симуляції.

4. Вихід з режиму симуляції для коригування програми.

Режим симуляції працює за такими правилами:

- програма (вкладка **Схема**) та макроси моделюються окремо;
- робота блоків, що не мають зв'язку із жодним з виходів пристрою або вихідним блоком мережевої змінної, не симулюється;
- симуляція не працюватиме для енергозалежних та некоректно прив'язаних змінних.

Вікно перегляду

Вікно перегляду можна відкрити або приховати, натиснувши кнопку  на панелі симуляції. Вікно призначене для перегляду значень змінних або входів/виходів кожного кроку виконання програми.

До запуску симуляції вікно пусте.

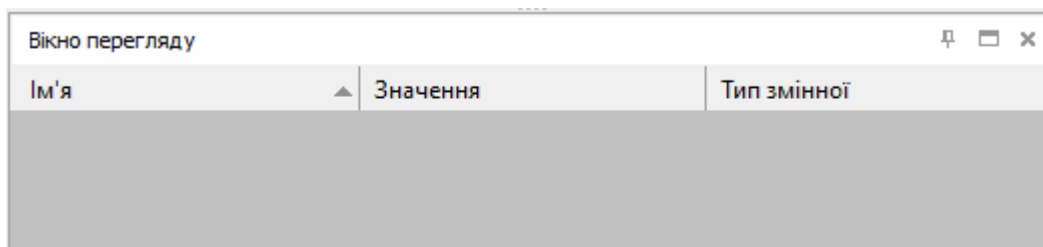


Рисунок 3.44

Для додавання змінної для відображення слід натиснути в порожнє поле колонки **Ім'я**, потім на кнопку «...».

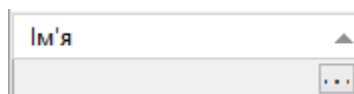


Рисунок 3.45

Відкриється таблиця змінних. У ній можуть бути обрані змінні проекту, входи та виходи.

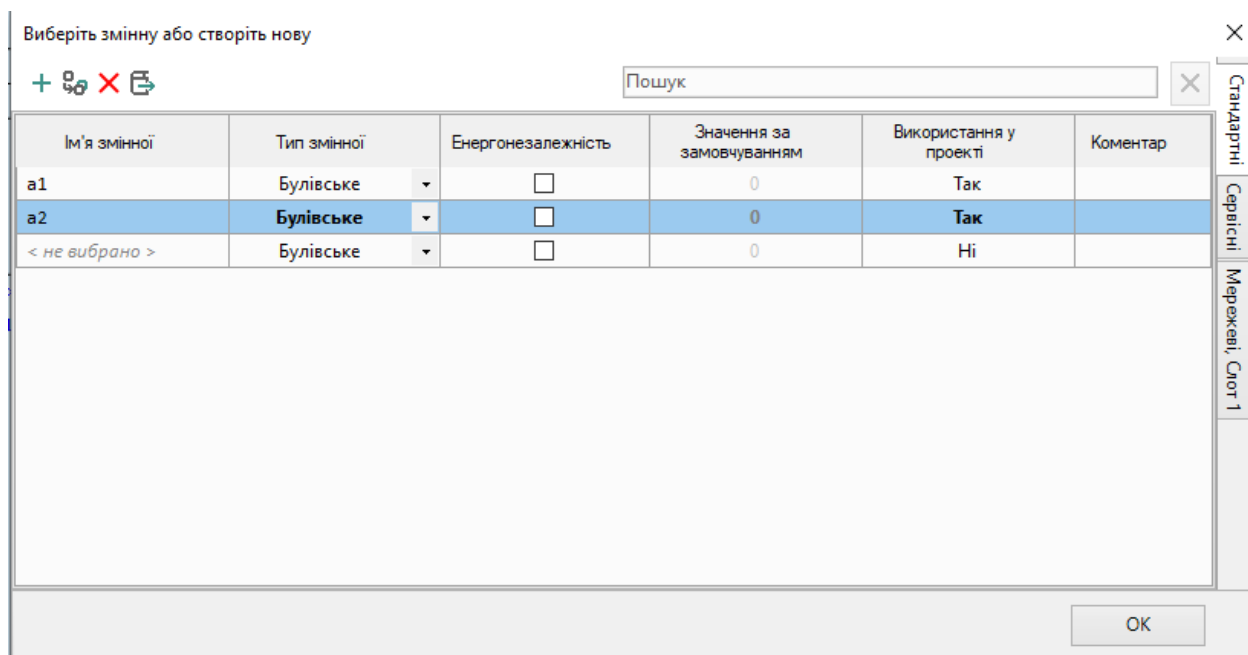


Рисунок 3.46

Вибрані змінні додадуться у вікно перегляду.

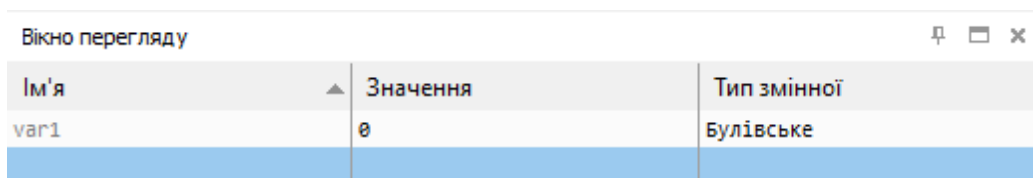


Рисунок 3.47

Також можна додати змінну у вікно перегляду натисканням ПКМ на змінній на схемі і вибором кнопки **Додати у вікно перегляду** в контекстному меню.

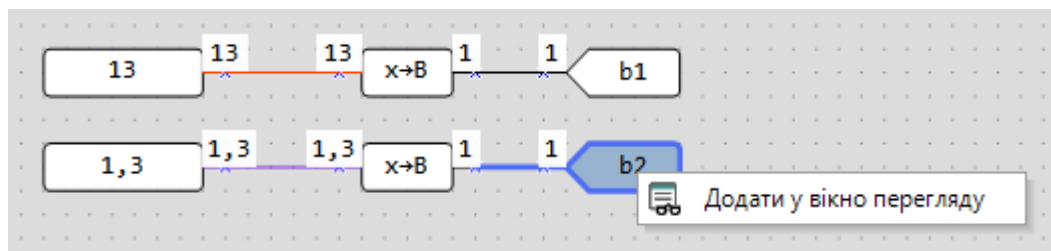


Рисунок 3.48

Під час симуляції значення змінних можна встановити у вікні перегляду в колонці **Значення**.

Для змінних можна вказувати значення на схемі. Подвійне натискання на змінну викликає вікно з полем введення нового значення.

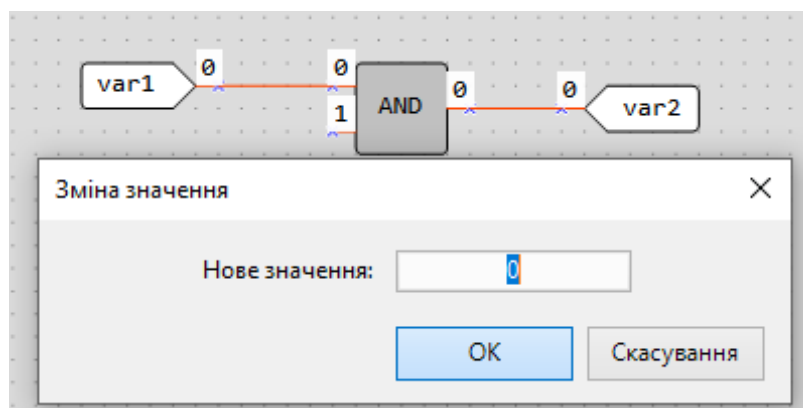


Рисунок 3.49

У режимі симулятора можна змінювати значення входів пристрою, натискаючи на них. Після натискання дискретні входи змінюють колір і булевське значення, для аналогових входів значення задаються аналогічно змінним у вікні з полем введення.

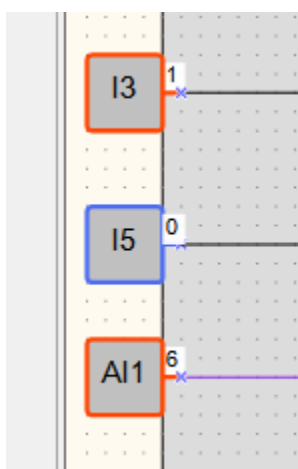


Рисунок 3.50

3.6 Налаштування порту та підключення пристрою

Щоб відкрити вікно **Налаштування підключення приладу**, слід вибрати в головному меню **Прилад** → **Налаштування порту** або натиснути на індикатор **COM** у [рядку станів 2.7](#).

У вікні **Налаштування підключення приладу** вибирається номер COM-порту, що використовується, та адреса пристрою, який підключається, інші налаштування фіксовані і виводяться для довідки.

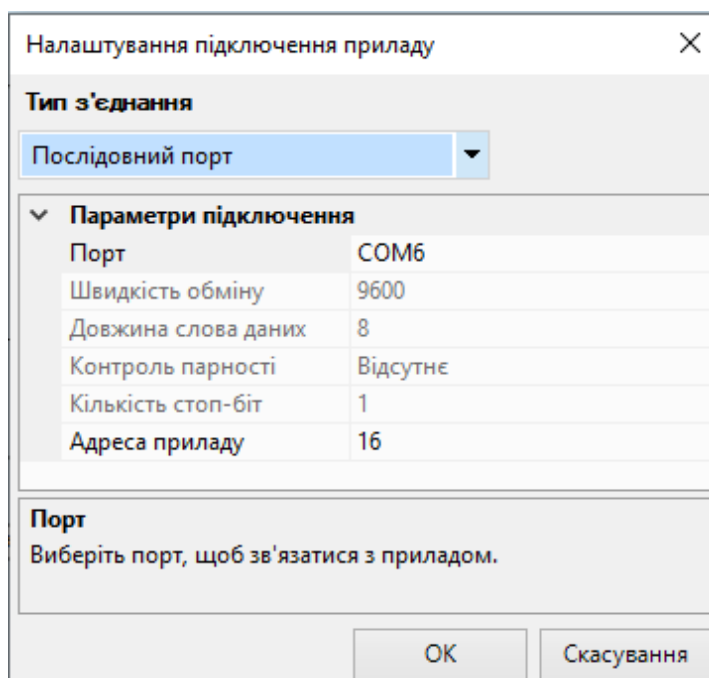


Рисунок 3.51

Підключення пристрою по USB

Для підключення пристрою до ПК через порт USB слід:

1. Встановити драйвер USB, який можна завантажити на сайті aqteck.com.ua.

2. Підключити пристрій до ПК та увімкнути живлення (якщо потрібно). Залежно від моделі пристрою для з'єднання використовується комплект для програмування ПР-КП20 або кабель для програмування з комплекту постачання пристрою. Номер емульованого COM-порту можна дізнатися в Диспетчері пристроїв ОС Windows.

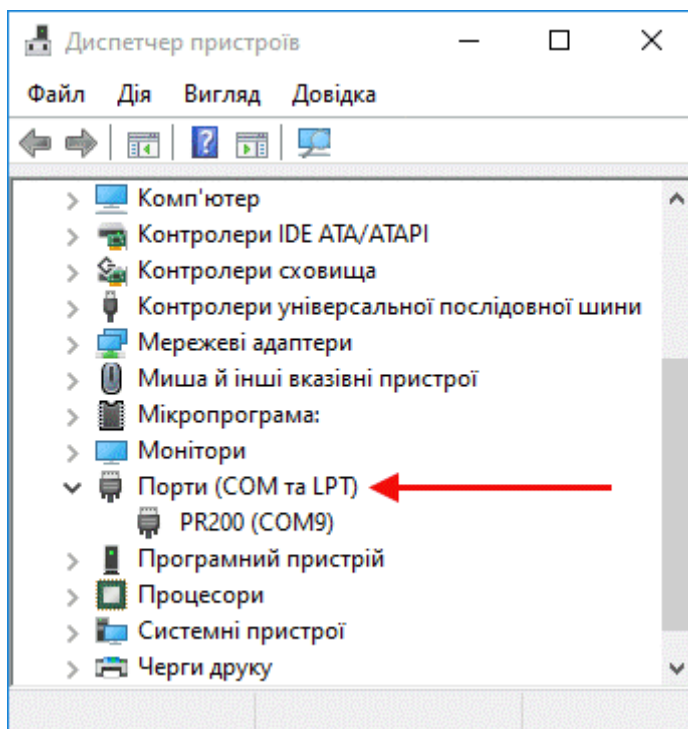


Рисунок 3.52

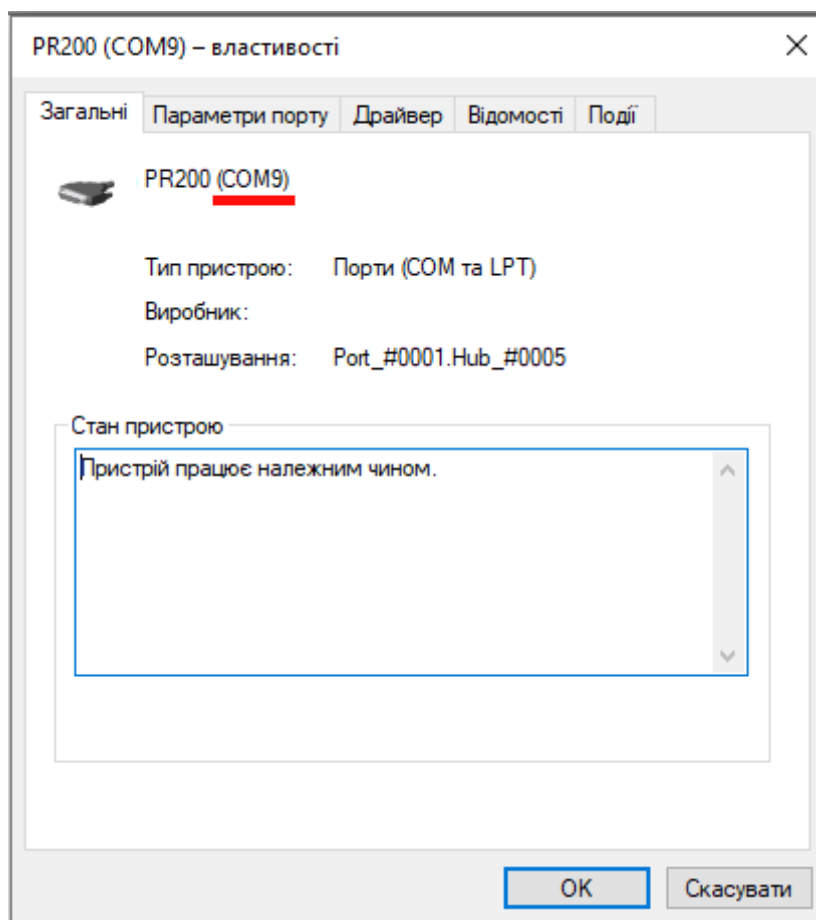


Рисунок 3.53

- У вікні Налаштування порту у випадному списку Тип підключення вибрати Послідовний порт.
- Ввести номер COM-порту та адресу пристрою (за умовчанням – 16) у вікні **Налаштування порту** і натиснути **ОК**. Якщо підключення коректне, то у рядку стану з'явиться найменування підключеного пристрою.

3.7 Запис програми у пристрій

Запис програми у пристрій




УВАГА

Якщо у підключеному пристрої вже є раніше записана програма, вона замінюється новою.

Створена програма записується у ПЗП пристрою.

Для запису програми до пристрою слід:

- [Підключити](#) пристрій до ПК.
- Подати живлення на пристрій, якщо потрібно.
- Налаштувати параметри з'єднання, якщо є потреба.
- Записати програму в пристрій, натиснувши кнопку  на панелі інструментів або вибравши в головному меню **Прилад** → **Записати програму у прилад**.



ПРИМІТКА

Перед записом програми AQLogic перевіряє сумісність із версією вбудованого ПЗ пристрою. Якщо вони не збігаються, то запусниться [оновлення вбудованого ПЗ 7.3](#) пристрою, потім програма запишеться у пристрій.

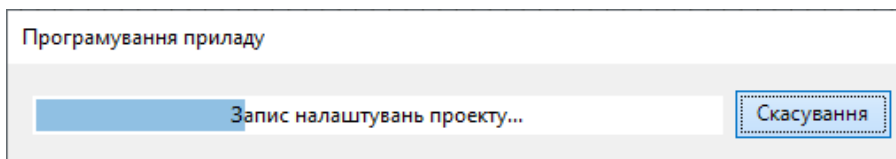


Рисунок 3.54

Якщо записується програма з налаштуваннями для іншої моделі пристрою, з'явиться таке повідомлення:

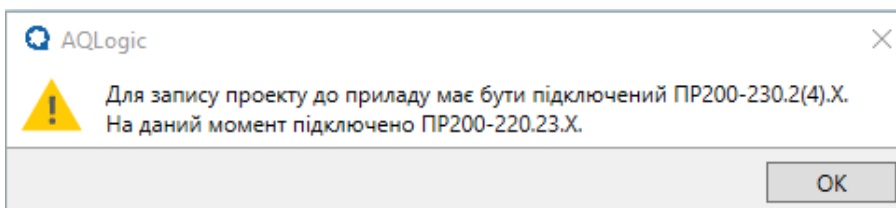


Рисунок 3.55


Переключення AQLogic у режим OFFLINE

Режим OFFLINE слід використовувати, якщо в двох або більше додатках AQLogic на одному ПК налаштовано передання даних в один пристрій. Додатки по черзі займатимуть порт, завантаження у пристрій в такому разі перерветься.

Додатки, які не повинні завантажуватися, слід перевести в режим OFFLINE.

Режим OFFLINE включається/відключається кліком на імені пристрою в [рядку стану 2.7](#) або в головному меню **Сервіс** → **Режим OFFLINE**.

3.8 Онлайн налагодження


Режим онлайн налагодження слугує для тестування програми з реальними значеннями з входів пристрою. Для запуску режиму онлайн налагодження слід натиснути кнопку  на панелі

інструментів, AQLogic перейде в режим онлайн налагодження, в ньому неможливо вносити зміни до проекту. Відкриється панель онлайн налагодження. Онлайн налагодження проводиться лише з підключеним до ПК пристроєм.

Послідовність та правила роботи в режимі онлайн налагодження

Онлайн налагодження можна проводити лише після завантаження програми у пристрій. Програма у пристрої має відповідати програмі на схемі проекту. У пристрої має бути відповідна версія [вбудованого ПЗ 7.3](#).

Для проведення онлайн налагодження слід:

1. Запустити онлайн налагодження натисканням на кнопку .
2. Перевірити логіку роботи програми з реальними значеннями на входах пристрою, які відображаються на схемі.

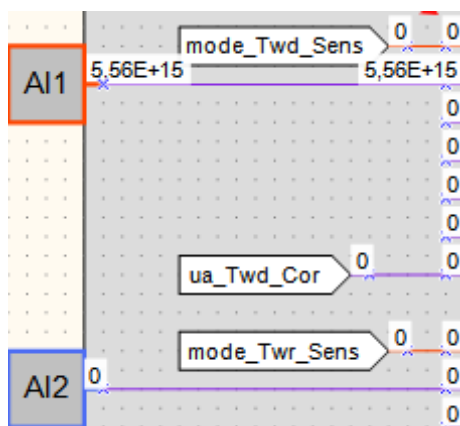



Рисунок 3.56

3. Для внесення коригувань вийти з режиму онлайн налагодження натисканням на кнопку .

Онлайн налагодження доступне лише для схеми проекту (для перевірки роботи макросів слід використовувати [режим симуляції 3.5](#)).

Ручне задання значень

У режимі онлайн налагодження на входи функцій, ФБ та макросів можна задавати власні значення, які не залежатимуть від значень з входів пристрою. Для власного значення в режимі онлайн налагодження слід натиснути на вхід блоку і ввести нове значення.

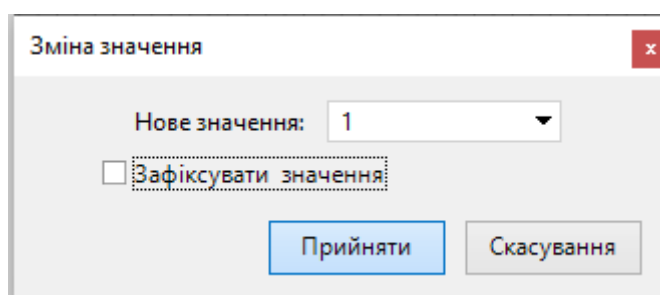


Рисунок 3.57

Якщо встановити галочку **Зафіксувати значення**, то нове значення діятиме протягом одного робочого циклу і по закінченні скинеться до значення пристрою або вихідного значення.

Протягом дії нового значення воно виділяється жовтим кольором.

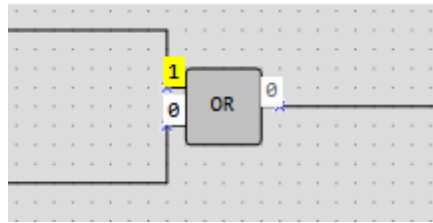


Рисунок 3.58

Якщо зафіксувати нове значення, воно діятиме кожен робочий цикл до зміни вручну на інше чи до відключення фіксації. Фіксація значень скидається у разі виходу з режиму онлайн налагодження. Зафіксоване значення відзначається на схемі в такий спосіб.

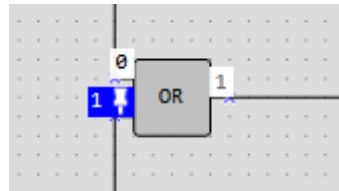


Рисунок 3.59

Усунення несправностей

У разі обриву зв'язку з пристроєм режим онлайн налагодження скинеться через 10 секунд, і пристрій перейде в робочий режим. Якщо встигнути відновити зв'язок, онлайн налагодження продовжиться, але зафіксовані значення будуть скинуті.

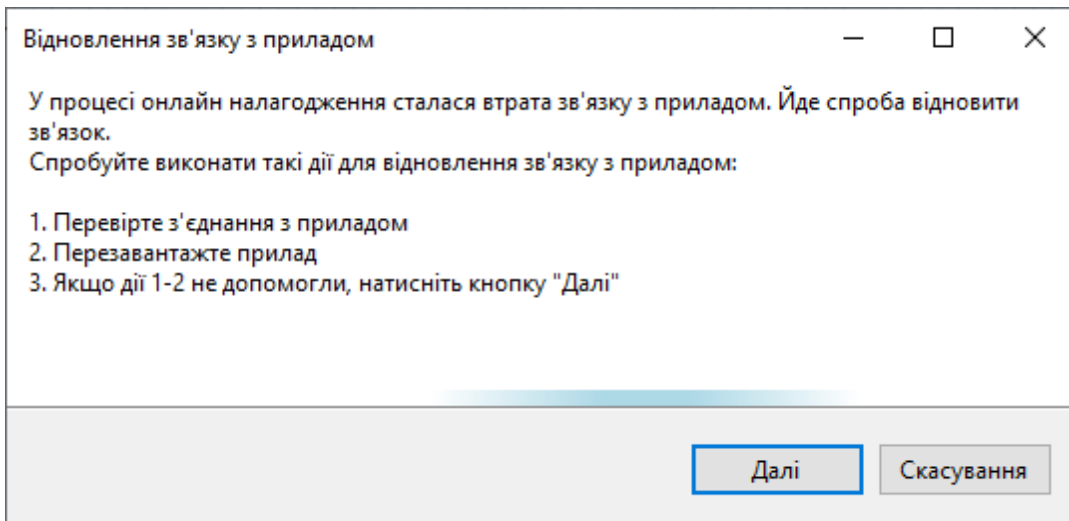


Рисунок 3.60



ПРИМІТКА

Для кожної модифікації пристрою є межа значень, що передаються в режимі онлайн налагодження. Якщо на схемі відображаються порожні комірки значень, значить спрацьовує обмеження, і слід збільшити масштаб схеми, щоб менше значень потрапляли до «видимого вікна». Зафіксовані значення залишаються зафіксованими, якщо не потрапляють у «видиме вікно», але скорочують межу значень, що передаються, оскільки займають області пам'яті.

3.9 Відомості про проект

Для запуску вікна **Відомості про проект** слід вибрати в головному меню **Файл** → **Відомості про проект....**

Вкладка Загальні

На вкладці **Загальні** міститься інформація про версії AQLogic, в яких створювався та змінювався проект.

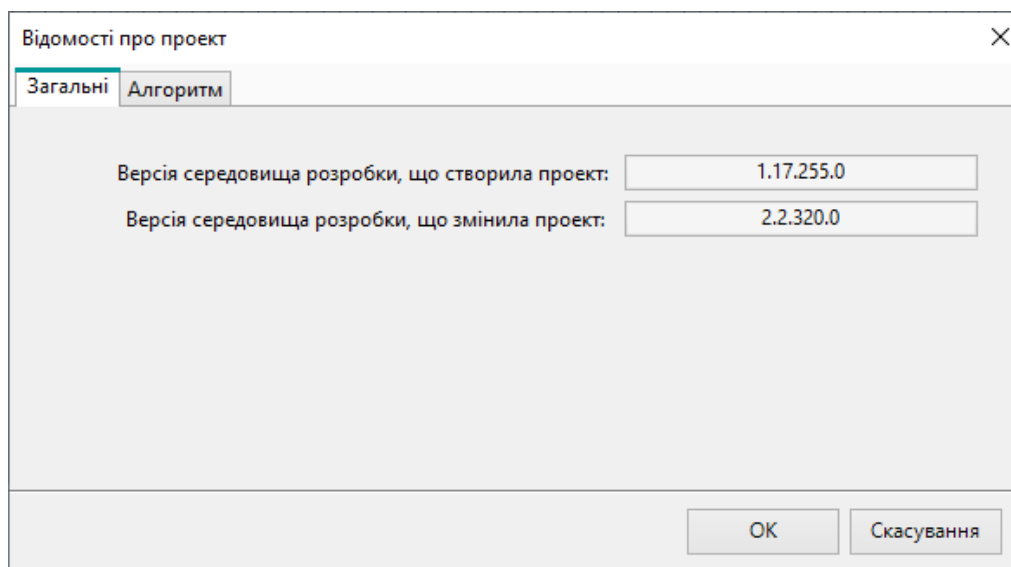


Рисунок 3.61

Вкладка Алгоритм



ПРИМІТКА

Вкладка **Алгоритм** доступна лише для пристроїв ПР200.

На вкладці **Алгоритм** можна вказати і переглянути інформацію про групу, номер та версію програми для відображення у вікні [інформації 7.1](#) про підключений пристрій після запису.

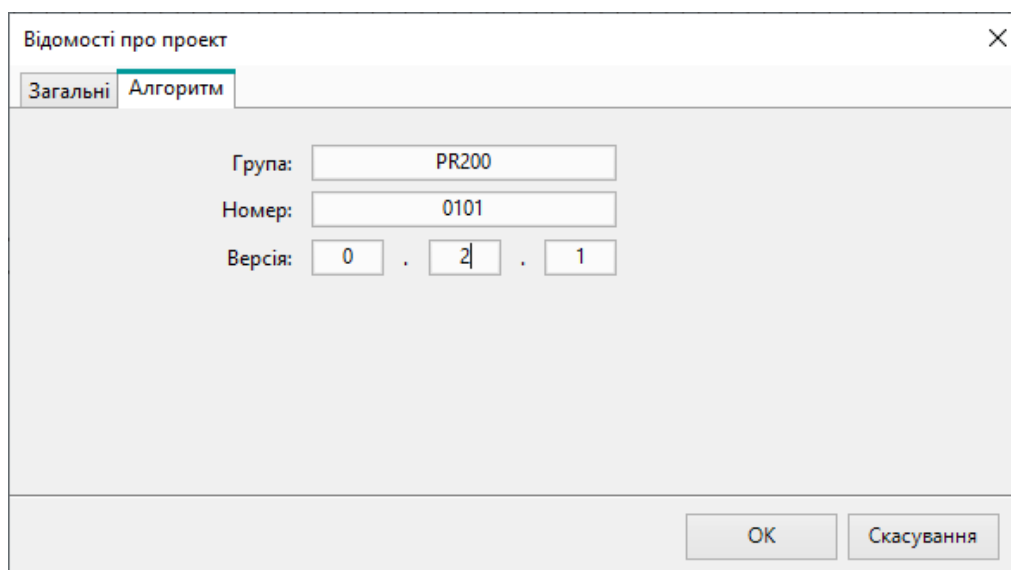


Рисунок 3.62

- Група – визначає групу (родину) алгоритмів;
- Номер – визначає номер та дозволяє відрізнити схожі алгоритми в рамках групи;
- Версія – визначає версію алгоритму. Рекомендується для задання версії використовувати формат [Semantic Versioning](#).

3.10 Менеджер компонентів

Менеджер компонентів слугує для завантаження в проект компонентів з онлайн-бази компанії АКУТЕК (вимагає підключення до Інтернету) та керування компонентами, що зберігаються на ПК.

Для запуску менеджера компонентів слід вибрати головне меню **Файл** → **Менеджер компонентів**.

Вкладка Онлайн-база

На вкладці **Онлайн-база** відображається список компонентів з онлайн-бази компанії АКУТЕК.

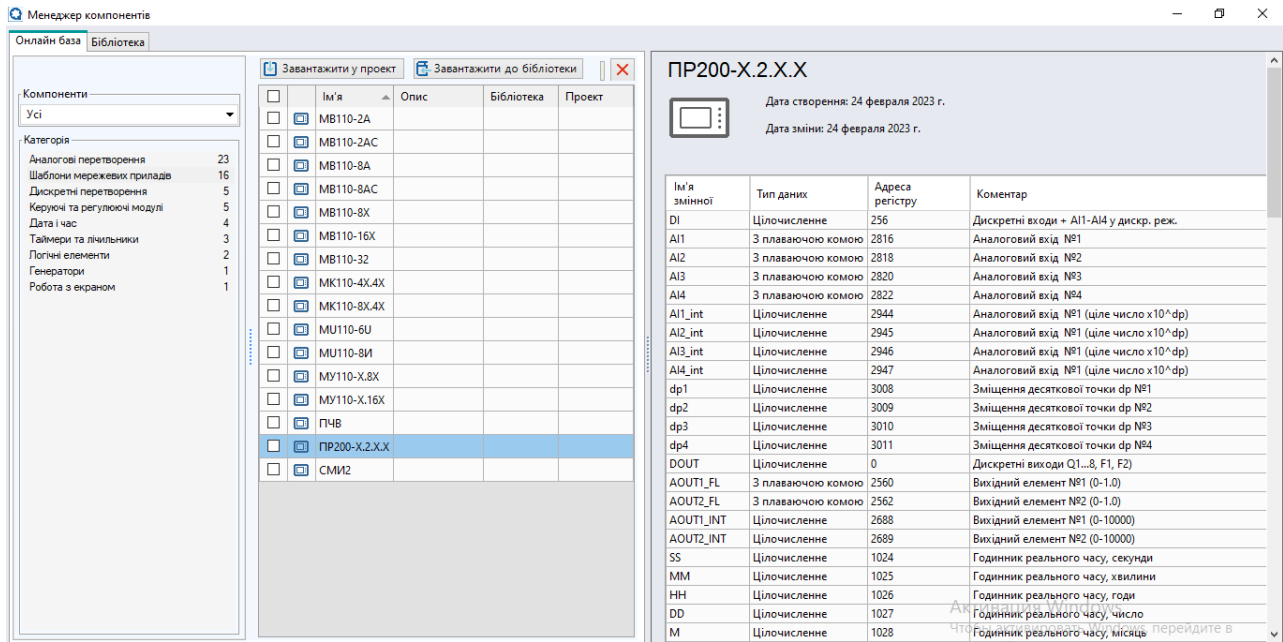


Рисунок 3.63

Випадне меню **Компоненти** дозволяє відфільтрувати список за типом:

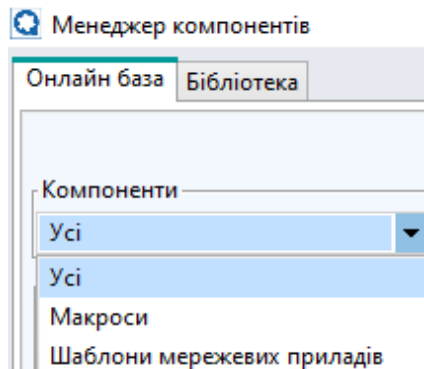


Рисунок 3.64

Макроси додатково розділені на категорії залежно від призначення:

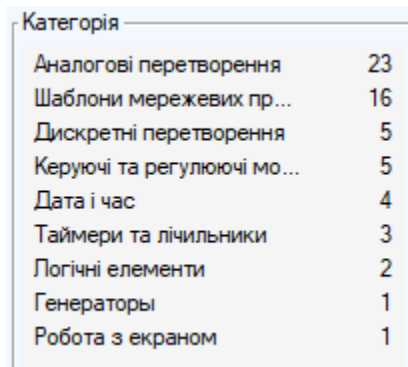


Рисунок 3.65

Повний опис компонента можна прочитати, виділивши його у списку. Опис компонента відображається у правій частині вікна. Для відображення повної документації до компонента потрібен Acrobat Reader.

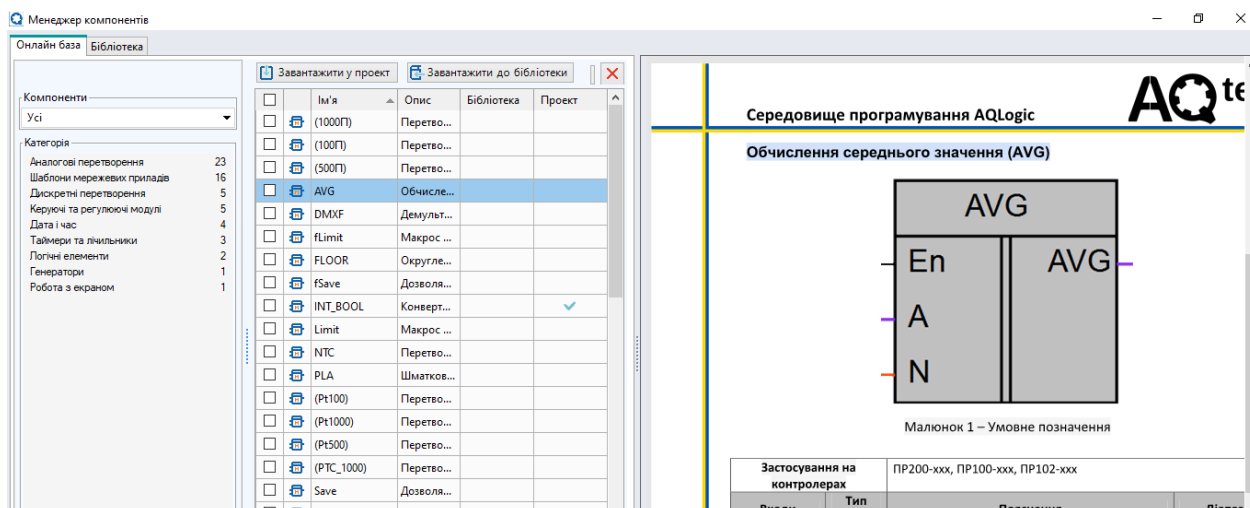


Рисунок 3.66

Щоб компоненти стали доступними без підключення до Інтернету, слід їх відзначити галочкою та натиснути кнопку **Завантажити до бібліотеки** – компоненти завантажуться у локальне сховище та відобразатимуться у вкладці **Бібліотека**.

Щоб одразу завантажити компоненти в поточний проект, слід їх відзначити галочкою та натиснути кнопку **Завантажити у проект** – компоненти будуть додані до панелі **Бібліотека компонентів** у розділ **Макроси проекту**.

Успішне завантаження компонентів у локальне сховище або поточний проект буде відображено у списку зеленою галочкою у графах **Бібліотека** та **Проект** відповідно.

Щоб підключити мережевий шаблон до конфігурації пристрою, слід скористатися рекомендаціями розділу [Режим Master 4.3.2.2](#).

Вкладка Бібліотека

На вкладці **Бібліотека** відображається вміст локального сховища компонентів на ПК користувача. Для завантаження компонентів у поточний проект слід виділити необхідні компоненти та натиснути кнопку **Завантажити у проект** – компоненти будуть додані до панелі **Бібліотека компонентів** у розділ **Макроси проекту**. Успішне завантаження компонентів у поточний проект буде відображено у списку зеленою галочкою у графі **Проект**.

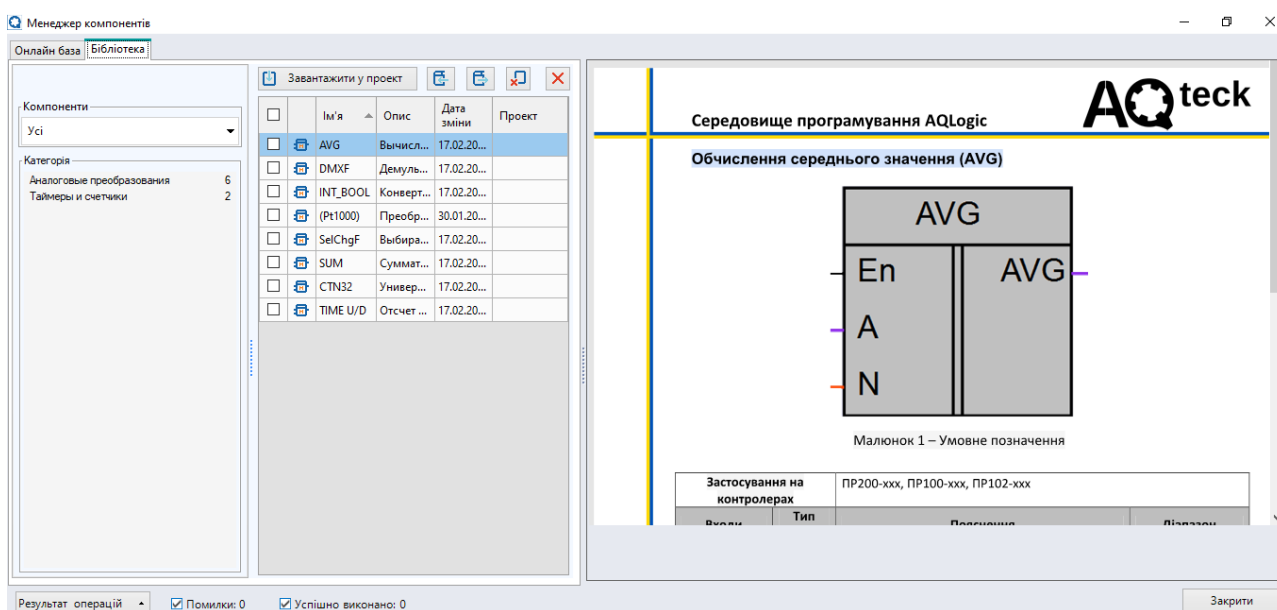





Рисунок 3.67



ПРИМІТКА

Локальне сховище компонентів розташоване за адресою:
 C:\Users\ім'я користувача\Documents\AQLogic \Library\


Менеджер компонентів підтримує такі функції керування компонентами:

-  – імпорт файлу в локальне сховище;
-  – експорт файлу з локального сховища;
-  – видалення вибраних компонентів.

Імпорт компонента

За допомогою імпорту можна завантажити компоненти, що зберігаються на ПК, але не входять до складу локального сховища.


Для імпорту компонента слід:

1. Натиснути кнопку .
2. У вікні, яке з'явилося, вибрати файли для імпорту.
3. Натиснути кнопку **Відкрити**. Файли з'являться на вкладці **Бібліотека**.

Експорт компонента

За допомогою експорту можна зберегти на ПК компоненти для подальшої зміни чи передачі.

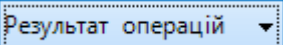
Для експорту компонента слід:

1. Позначити галочкою потрібний компонент у вкладці **Бібліотека**.
2. Натиснути кнопку .
3. У вікні, що з'явилося, вибрати шлях для експорту.
4. Натиснути кнопку **ОК**. Компонент збережеться у вигляді файлу за вказаним шляхом.

Інше

Для документації до компоненту доступні функції відповідно до встановленої версії Acrobat Reader, у тому числі збереження файлу у форматі pdf та друк.

Для відображення результатів операцій у менеджері компонентів слід натиснути кнопку

 – у нижній частині вікна розкриється рядок повідомлень.

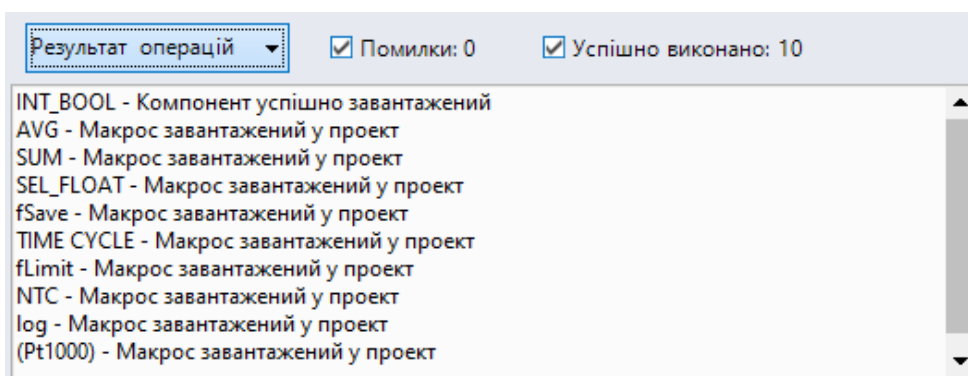


Рисунок 3.68

3.11 Робота з макросами

Макрос – це алгоритм користувача для використання в проекті.

Основні дії з макросами:

- **збереження** у **Бібліотеці компонентів 6** для подальшого використання у проекті;
- **експортування** до файлу для використання в інших проектах;
- **імпортування** створених раніше макросів;
- **завантаження** з онлайн-бази до проекту.

Створення макроса

Для створення макроса слід:

1. Вибрати у головному меню **Файл** → **Створити макрос**. У діалоговому вікні, що відкрилося, задати кількість входів і виходів у макросі:

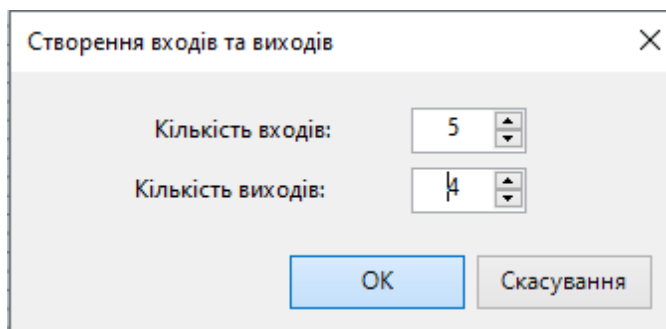


Рисунок 3.69



ПРИМІТКА

Кількість входів та виходів можна змінити після створення макросу

2. Розробити алгоритм роботи макросу у вкладці **Редактор макросів**, аналогічно до розробки програми на схемі.

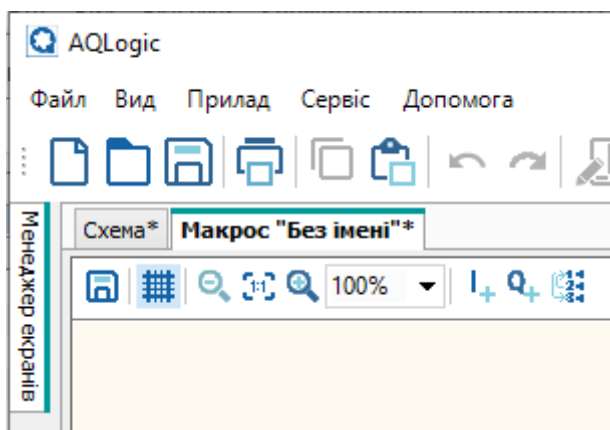


Рисунок 3.70

Кількість входів та виходів можна збільшити за допомогою контекстного меню редактора або

за допомогою кнопок  та .

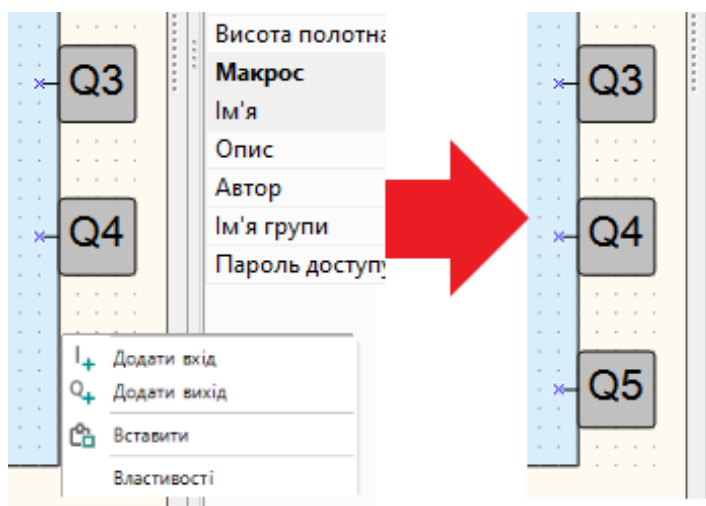


Рисунок 3.71

Щоб видалити вхід або вихід, слід натиснути ПКМ і вибрати команду **Видалити**.

- Задати ім'я та опис макросу на панелі властивостей.

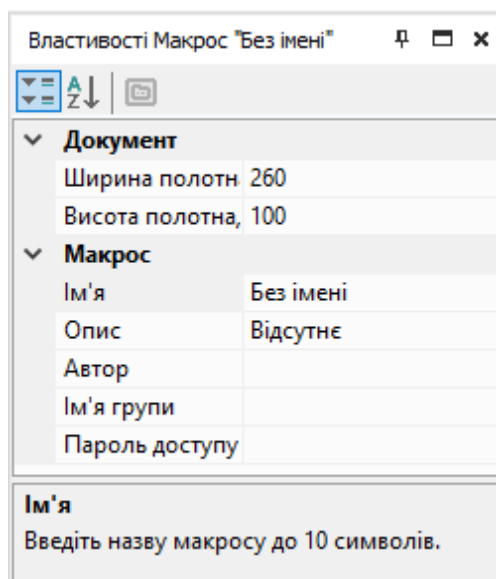


Рисунок 3.72

Введене ім'я буде вказано в заголовку редактора макросів та на схемі.

- Для визначення типу та імені входів/виходів вибрати вхід або вихід і налаштувати його на панелі властивостей.

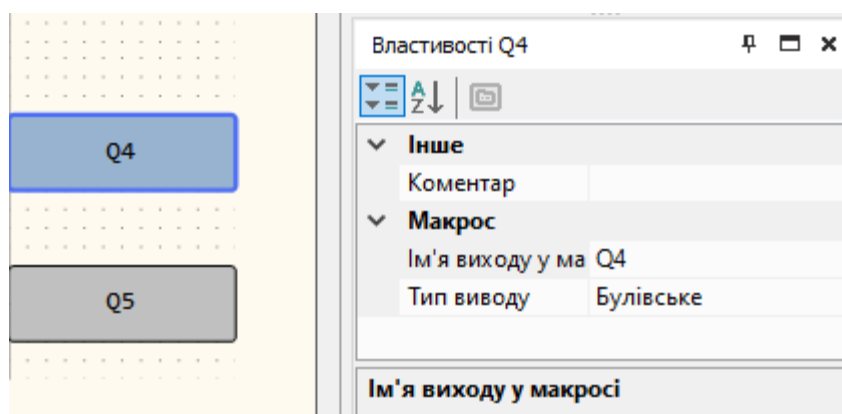


Рисунок 3.73

5. Далі можна задати «видимість» із програми уставок ФБ, використаних у макросі.

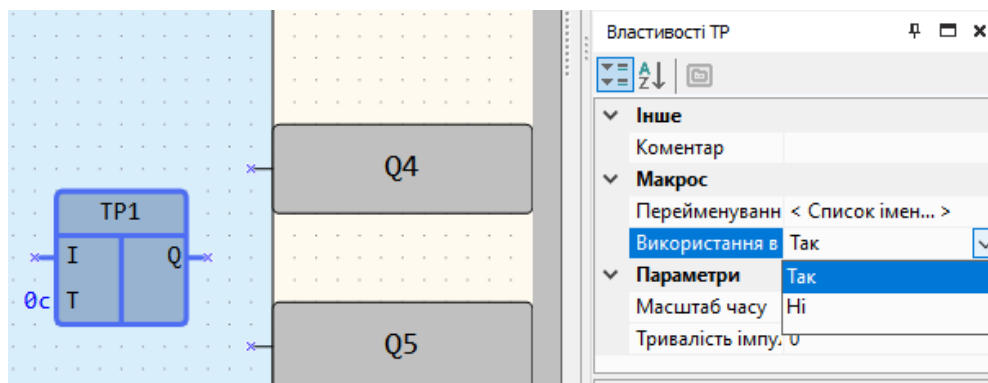


Рисунок 3.74

Щоб уставки вибраного ФБ були доступні під час використання ФБ у програмі, слід налаштувати на панелі властивостей блоку параметр **Використання в макросі – Так**.

Після вибору варіанта **Так** з'явиться випадний список **Перейменування змінних** з переліком доступних змінних. Наприклад, для ФБ BLINK:

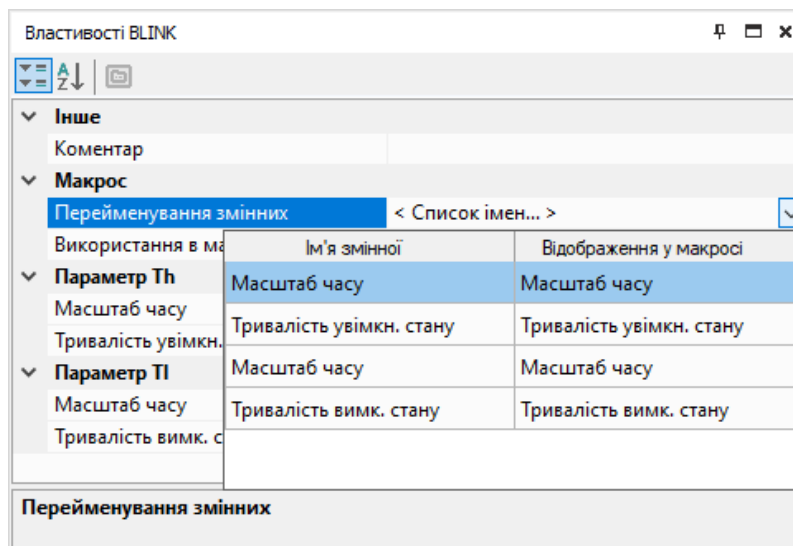


Рисунок 3.75

6. Після створення макросу запустити [режим симулятора 3.5](#) та перевірити логіку його роботи.

7. Перед збереженням макросу можна заповнити поля: **Ім'я**, **Опис**, **Автор**, **Ім'я групи** та **Пароль доступу**.

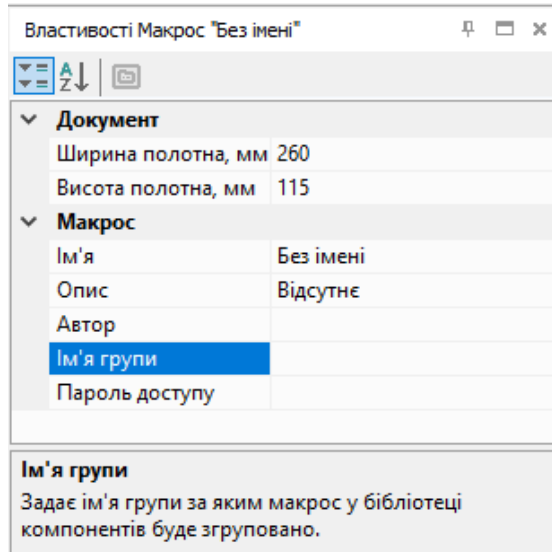



Рисунок 3.76

Рекомендується задавати коротке і зрозуміле ім'я макросу. Вміст поля **Опис** відображається під зображенням макросу на панелі **Бібліотека компонентів**.

Якщо пароль не заданий (поле **Пароль** не заповнене), доступ до редагування макросу відкритий для всіх.

Якщо поле **Ім'я групи** не заповнене, макрос зберігається в бібліотеці проекту в групі **Інші**.

Макрос можна зберегти, вибравши в головному меню **Файл** → **Зберегти макрос як** або натиснувши кнопку  на панелі редактора макросів.

Збережений макрос доступний лише для відкритого проекту, для відкриття макросу в інших проектах його слід експортувати.

8. Вибрати розділ **Макроси проекту** на панелі **Бібліотека компонентів** і перенести макрос на полотно проекту.

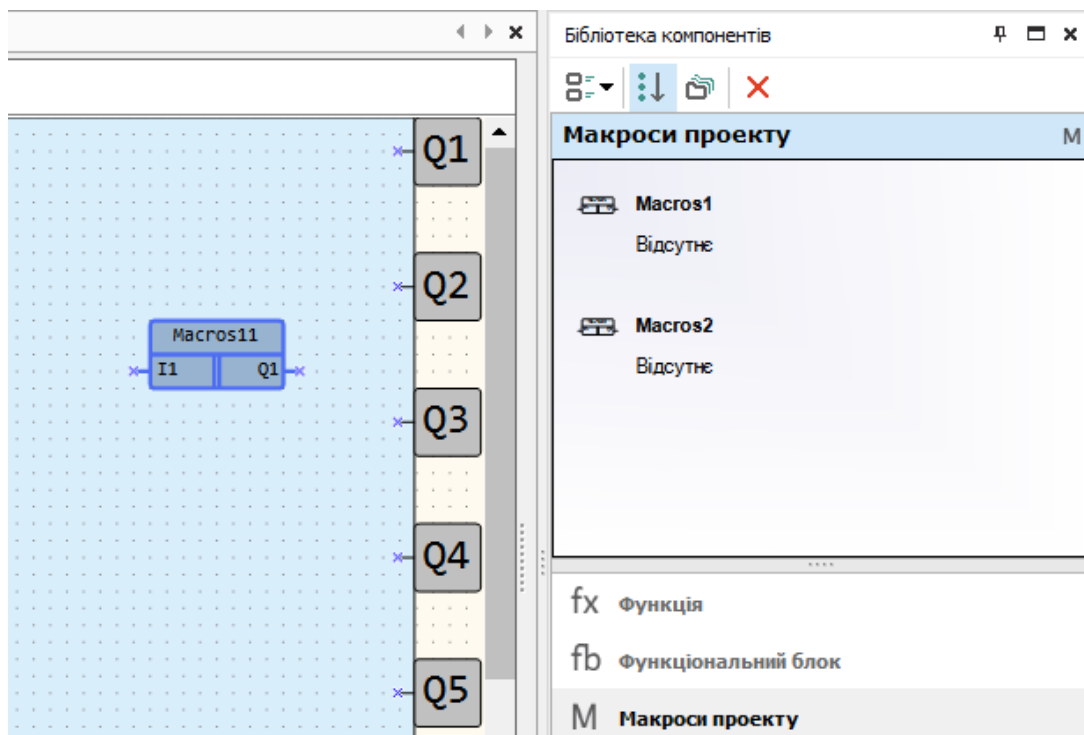


Рисунок 3.77

Створення макросу на схемі

Макрос можна створити, виділивши елементи на схемі проекту та вибравши в контекстному меню **Створити макрос**. На місці елементів з'явиться блок макросу.

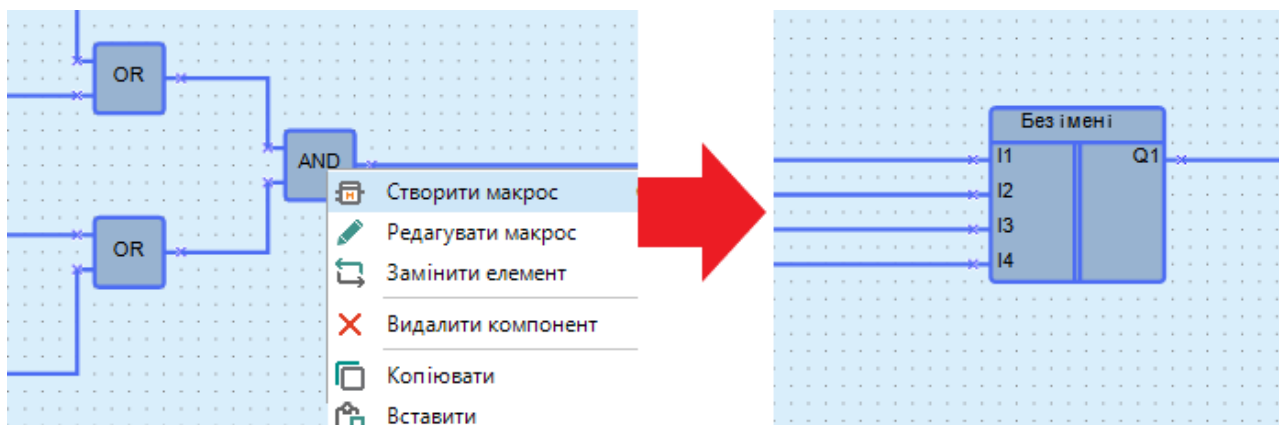


Рисунок 3.78

Входи і виходи макросу називаються згідно з підключеними під час створення макросу елементами і блоками. Якщо до виходу макросу приєднано кілька елементів, то вихід називається за умовчанням (Q_n , де n – порядковий номер виходу макросу).

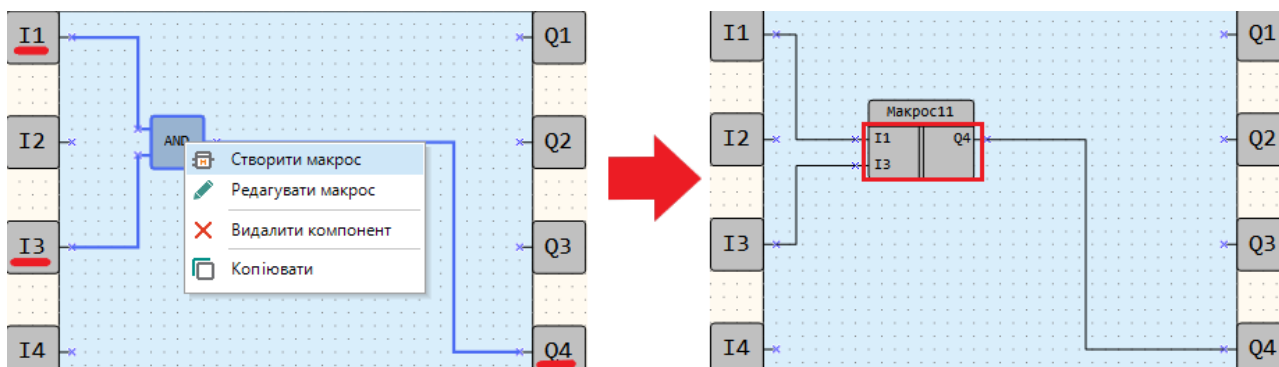


Рисунок 3.79

Створений макрос та назви входів/виходів можна відредагувати в **редакторі макросів**, вибравши у контекстному меню блоку **Редагувати макрос**.

Кількість входів та виходів макросу буде відповідати кількості з'єднаних вхідних та вихідних зв'язків у виділеній ділянці схеми. У разі виділення елементів без зв'язків створюється макрос з одним входом та виходом.

Елементи, які не потрапляють у макрос під час створення на схемі:

- входи і виходи;
- сервісні змінні;
- мережеві змінні;
- ПІД-регулятор.

Якщо вищезазначені елементи були виділені при створенні макросу, вони будуть з'єднані з відповідним входом/виходом блоку макросу. Якщо у виділену область потрапив ФБ, у який записується чи зчитується значення, то у макрос входить блок запису/читання з ФБ, навіть якщо він перебуває поза виділеної області.

У разі потрапляння стандартних змінних в область виділення, вони копіюються всередину макросу під тим самим ім'ям. Змінні в макросі не тотожні змінним на схемі. Якщо всі блоки однієї змінної потрапляють усередину макросу, вони зникають зі схеми.



ПРИМІТКА

Якщо у виділену область потрапляють вхідний і вихідний блоки однієї змінної, яка використовується поза межами області на схемі, то блоки копіюються під тим самим ім'ям всередину макросу і на схемі залишається тільки блок вихідної змінної.

Оновлення макросу у програмі

Якщо у макросу, який використовують, були змінені ім'я, тип, кількість входів/виходів або доданий елемент (змінна, константа) з властивістю **Використання в макросі – Так**, AQLogic

вимагатиме його оновити та позначить кольором на полотні. Макрос вважається зміненим після збереження правок у **Редакторі макросів**.

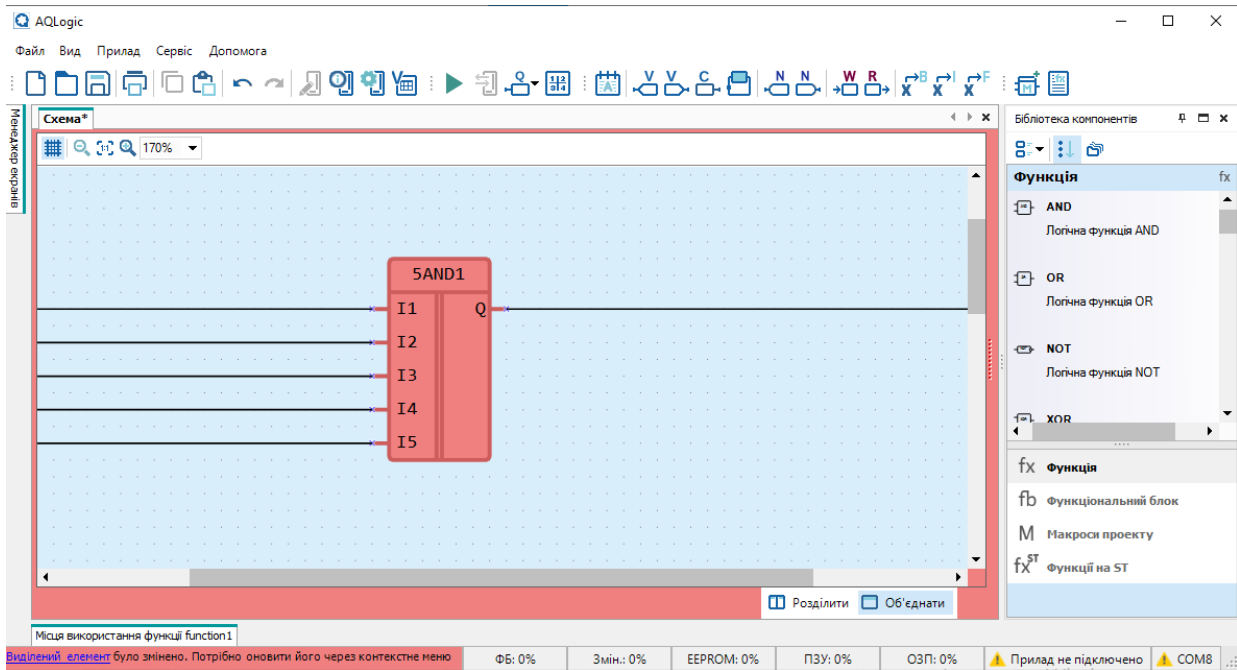


Рисунок 3.80

Для оновлення макросу на полотні слід:

1. Натиснути по відзначеному кольором макросу ПКМ і в меню, яке з'явиться, вибрати **Оновити макрос**.

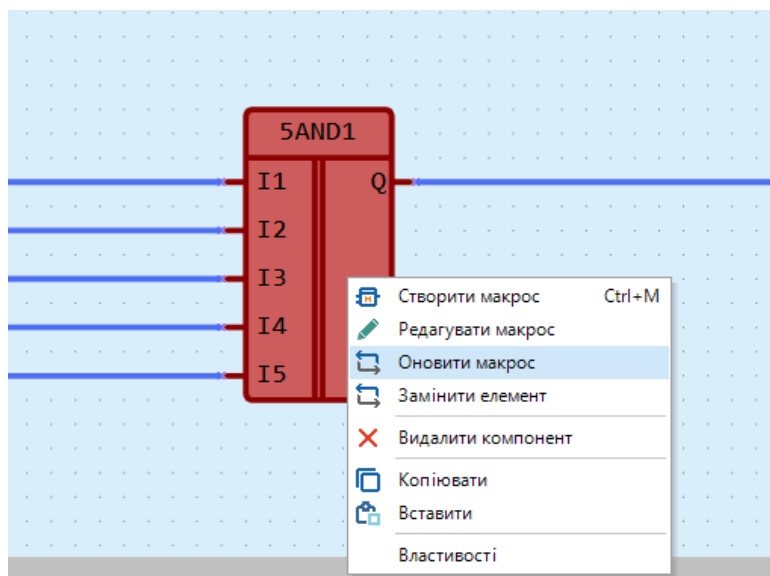


Рисунок 3.81



ПРИМІТКА

У разі вкладеного макросу AQLogic запропонує перейти до нього. Дії із вкладеним макросом аналогічні.

2. Після оновлення макросу AQLogic автоматично перейде до наступного макросу, який потребує оновлення.

Макрос оновлюється у проєкті за такими правилами:

- якщо в макросі змінено тип або ім'я входу/виходу з приєднаним зв'язком, при оновленні зв'язок буде розірвано;
- якщо до макросу додано входи/виходи, при оновленні зв'язки вже існуючих входів/виходів не розриваються.

**ПРИМІТКА**

AQLogic ідентифікує зв'язки з входами/виходами макросу по імені та типу. Якщо в макросі змінити ім'я або тип входу/виходу з приєднаним зв'язком і створити новий вхід/вихід з таким же ім'ям та типом, то при оновленні макросу AQLogic автоматично переєднає зв'язок до нового входу/виходу.

Швидка заміна макросу

Якщо потрібно замінити макрос на інший з макросів проекту, натисніть кнопку **Замінити елемент** у контекстному меню макросу.

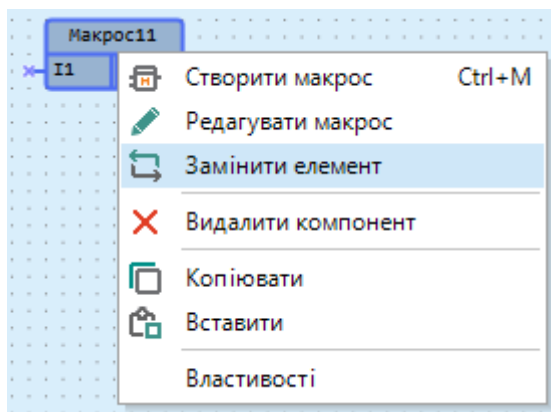


Рисунок 3.82

Відкриється вікно **Замінити макрос**.

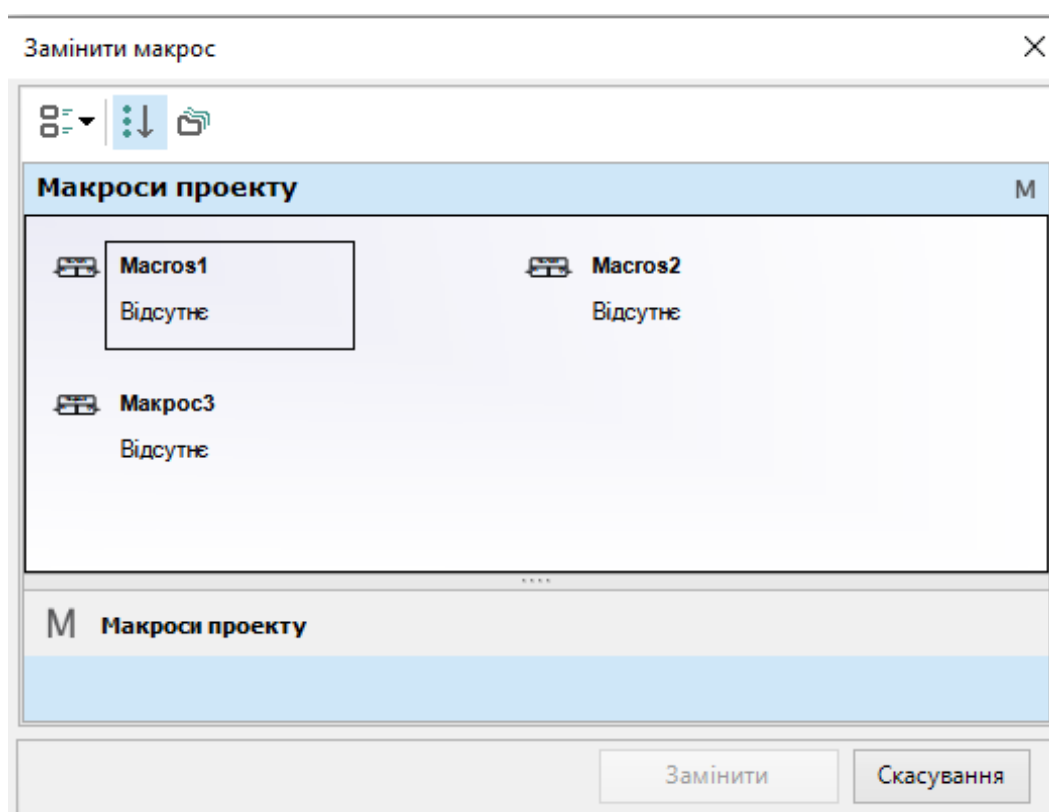


Рисунок 3.83

Якщо в макросі змінено тип або ім'я входу/виходу з приєднаним зв'язком, то при заміні зв'язок буде розірвано, інші зв'язки будуть збережені.

Експорт макросу

Експортувати макрос у файл можна лише при відкритому вікні редактора макросу. Щоб експортувати макрос, виберіть у головному меню **Файл** → **Експорт макросу до файлу**.

Для експорту макросу з панелі бібліотеки компонентів слід:

1. Відкрити макрос у редакторі.

Якщо перед збереженням макрос потрібно відредагувати, його слід перетягнути на полотно проекту і в контекстному меню макросу вибрати **Редагувати** і внести зміни.

2. Вибрати в головному меню **Файл** → **Експорт макросу до файлу**.

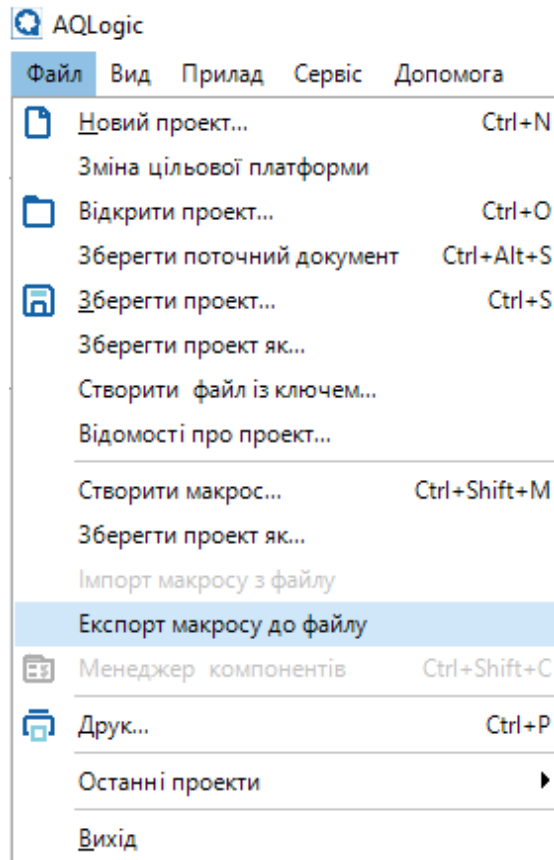


Рисунок 3.84

3. У вікні, що відкрилося, вибрати місце розташування та зберегти файл макросу з розширенням **.tpl*. Після збереження з'явиться повідомлення про успішний експорт макросу.

Імпорт макросу

Якщо для створення програми потрібно використовувати макрос, створений в іншому проекті, необхідний макрос можна імпортувати в проект.

Для імпорту макросу слід вибрати в головному меню **Файл** → **Імпорт макросу з файлу**.

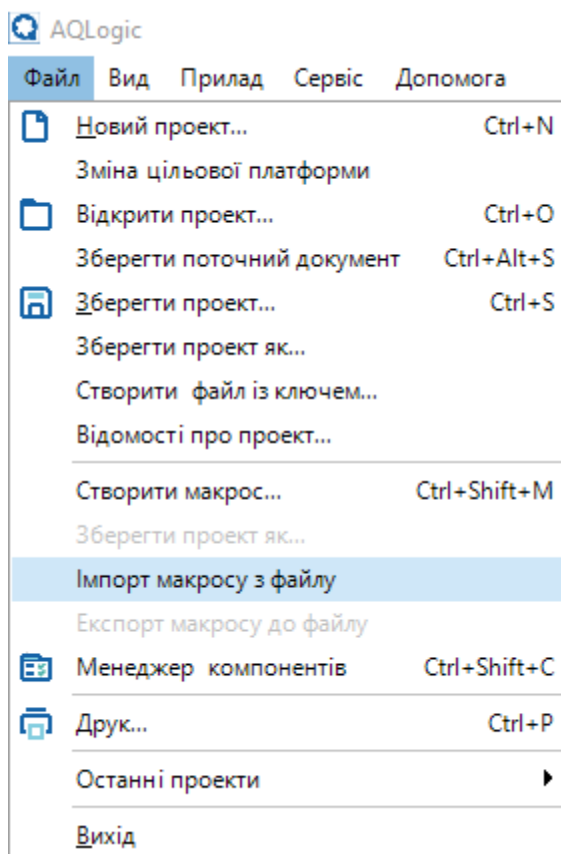




Рисунок 3.85

У вікні, що відкрилося, вибрати потрібний файл і натиснути кнопку **ОК**. Макрос буде додано до панелі **Бібліотека компонентів** у розділ **Макроси проекту**, тепер його можна використовувати в проекті.

Копіювання макросу

Макрос можна скопіювати з проекту до проекту для повторного використання та скорочення часу на розробку. Для копіювання макросу слід виділити блок макросу у вихідному проекті та натиснути кнопку  на панелі інструментів або вибрати команду **Копіювати** у контекстному меню блоку.

Макрос вставляється в інший проект натисканням кнопки  на панелі інструментів або за допомогою вибору команди **Вставити** в контекстному меню полотна. Для копіювання та вставки також можна скористатися комбінаціями клавіш, див. розділ [Поєднання клавіш 10](#).

Після вставки макрос буде доступний у розділі **Макроси проекту** на панелі **Бібліотека компонентів**.

3.12 Створення функції мовою ST

Для пристроїв на новій платформі доступне створення функцій користувача мовою ST.

Створення функції

Для створення функції слід:

1. Вибрати на панелі інструментів **Створити функцію на ST**. Відкриється вкладка **Редактор функції**.

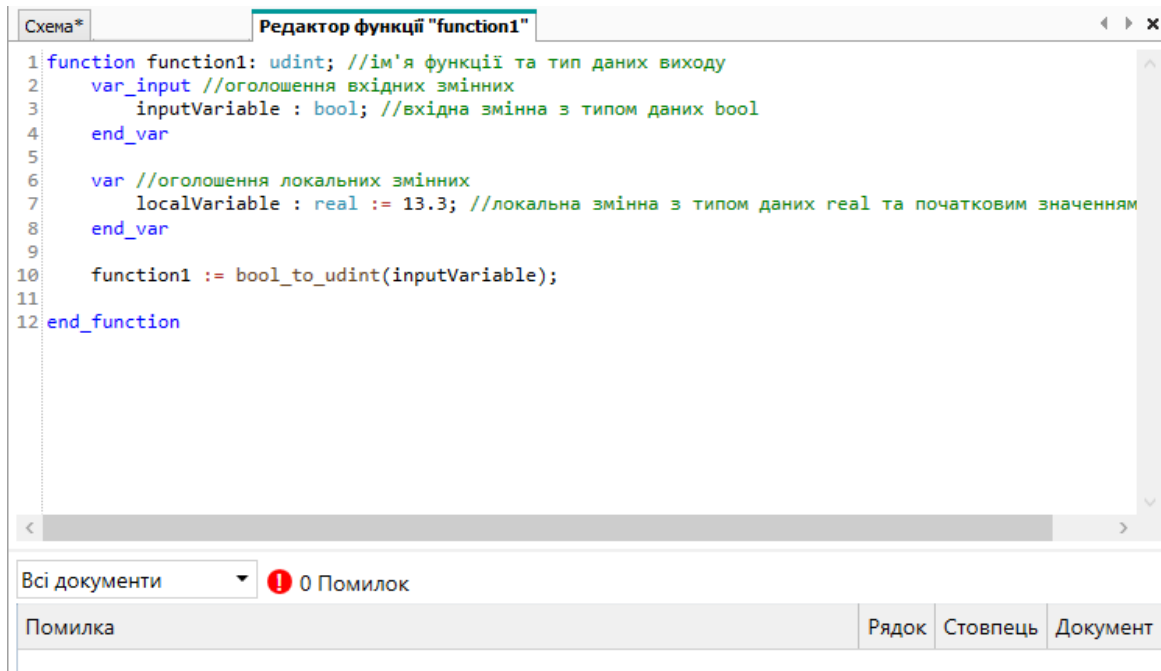


Рисунок 3.86

**ПРИМІТКА**

За умовчанням перші дванадцять рядків редактора заповнені шаблоном функції.

2. Задати ім'я функції та тип вихідних даних у першому рядку редактора.
3. Вказати необхідну кількість входних змінних у блоці оголошення входних змінних (**var_input**).
4. Вказати необхідну кількість локальних змінних у блоці оголошення локальних змінних (**var**).
5. Розробити алгоритм роботи функції відповідно до синтаксису мови ST.

**ПРИМІТКА**

Докладніше див. [Мова програмування ST 11](#).

6. Перейти до вкладки **Схема** або закрити вкладку **Редактор функції**. Функція збережеться автоматично.

7. Вибрати розділ **Функції на ST** на панелі **Бібліотека компонентів** та перенести функцію на полотно проекту.

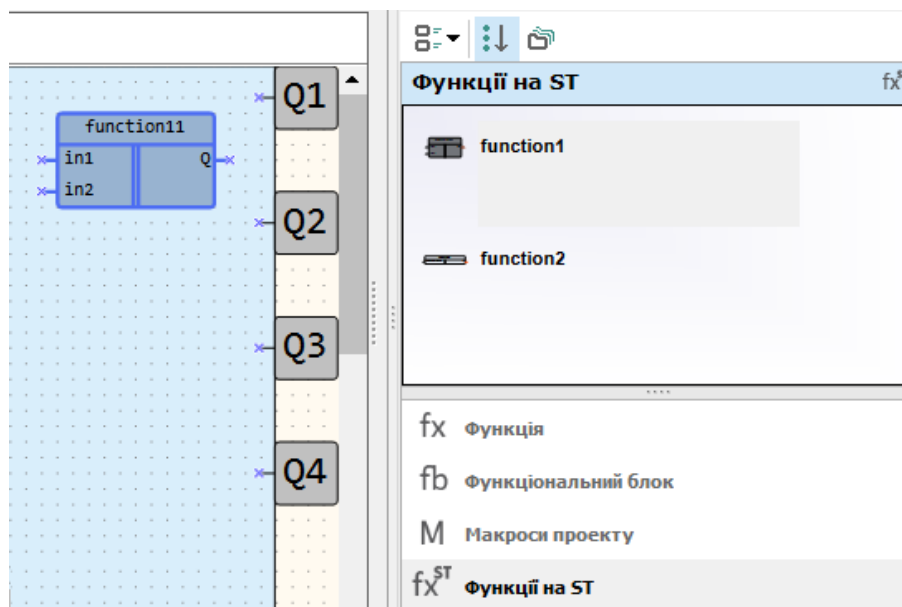


Рисунок 3.87

Редактор функції

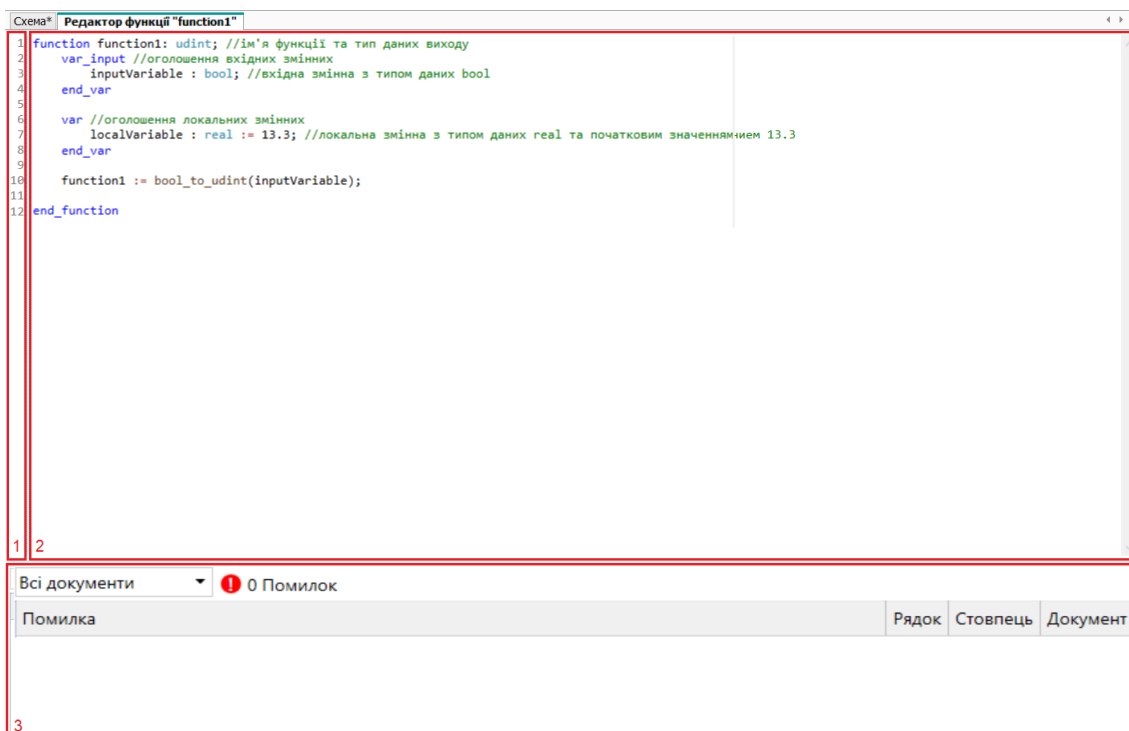


Рисунок 3.88

Елементи редактора функції:

1. **Номери рядків** – заповнюється автоматично за кількістю написаних рядків коду.
2. **Редактор коду** – область редагування коду з функцією підсвічування синтаксису.
3. **Список помилок** — область відображення помилок.

Сніпети

Для зручності написання коду в редакторі функцій реалізовані сніпети. Сніпет — фрагмент коду програми, який автоматично вставляється в код. У разі введення першого символу в редакторі з'явиться контекстне меню з фокусом на першому рядку. Для вибору сніпета використовуються клавіші керування курсором. Щоб вставити вибраний сніпет у код, слід натиснути клавішу **Enter** або **Tab** чи двічі натиснути ЛКМ по пункту списку.

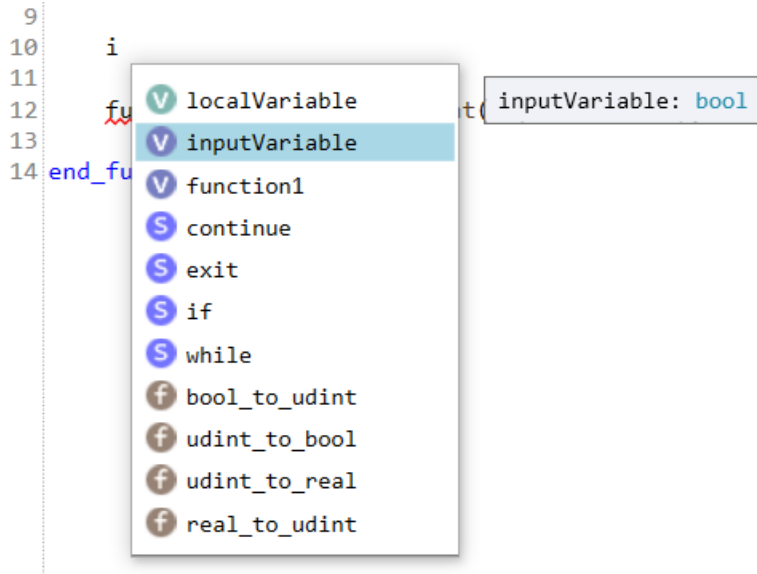


Рисунок 3.89

Список сніпетів формується за групами:

- локальні змінні функції;
- конструкції (while, for тощо);
- вирази (true, false);
- вбудовані функції;
- інші функції проекту.

Усередині груп сніпети розташовані за алфавітом.

Перехід до місця оголошення або використання

Для зручної роботи з функціями та змінними доступний перехід до місць їх оголошення чи використання.

Для переходу до місця оголошення:

1. Встановити курсор на ім'я функції або змінної у коді програми.
2. Натиснути ПКМ на ім'я функції або змінної.
3. У контекстному меню, яке відкрилося, вибрати пункт **Перейти до оголошення**.

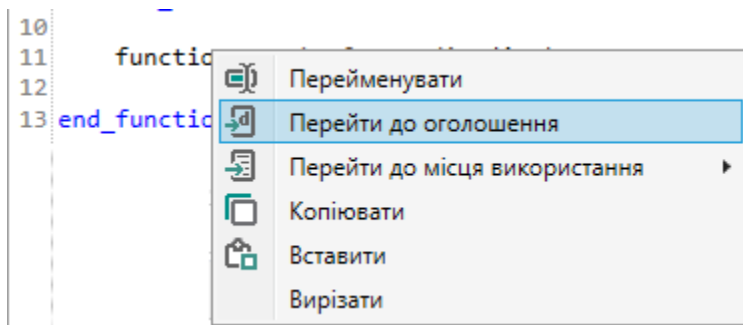


Рисунок 3.90

Для переходу до місця використання:

1. Встановити курсор на ім'я функції або змінної у коді програми.
2. Натиснути ПКМ на ім'я функції або змінної.

- У контекстному меню, яке відкрилося, вибрати пункт **Перейти до місця використання**. Відкриється список місць, де використовується вибрана змінна або функція.

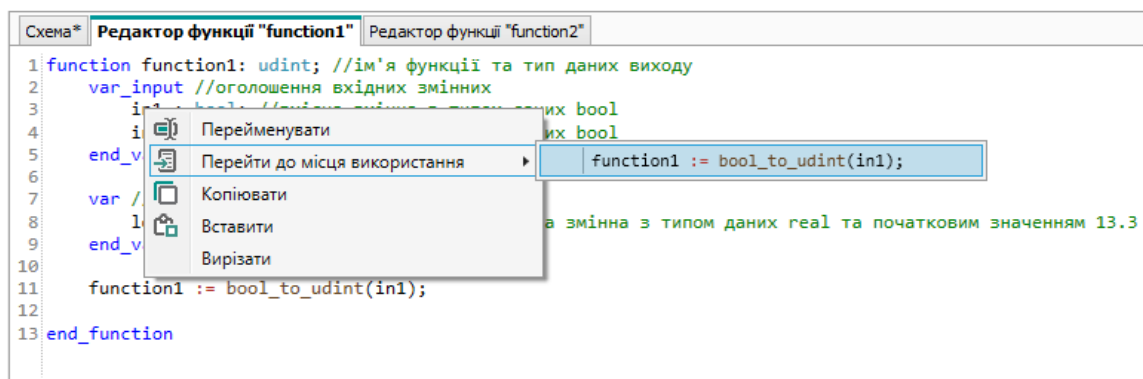


Рисунок 3.91

- Натиснути ЛКМ на обране місце використання. Курсор переміститься на рядок, у якому використовується функція або змінна.

Переіменування змінної або функції

Для зручної роботи з іменами функцій та змінних доступне централізоване переіменування.

Для переіменування слід:

- Встановити курсор на ім'я функції або змінної у коді програми.
- Натиснути ПКМ на ім'я функції або змінної.
- У контекстному меню вибрати пункт **Переіменувати**. Ім'я буде виділено зеленим у всіх місцях використання.

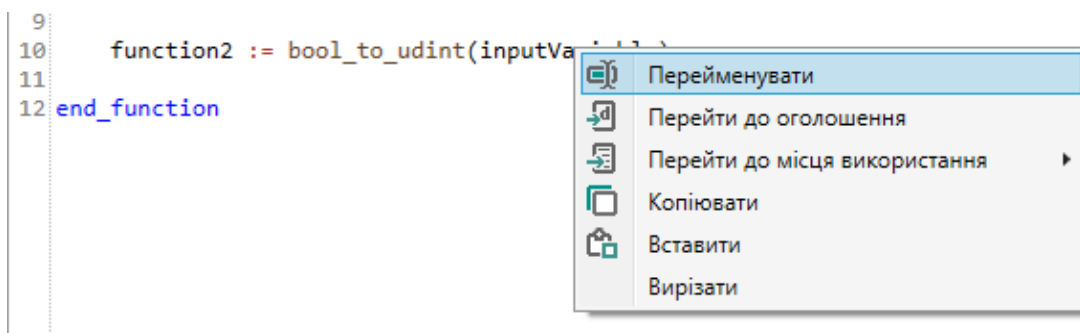


Рисунок 3.92

- Ввести нове ім'я змінної в області оголошення змінних.

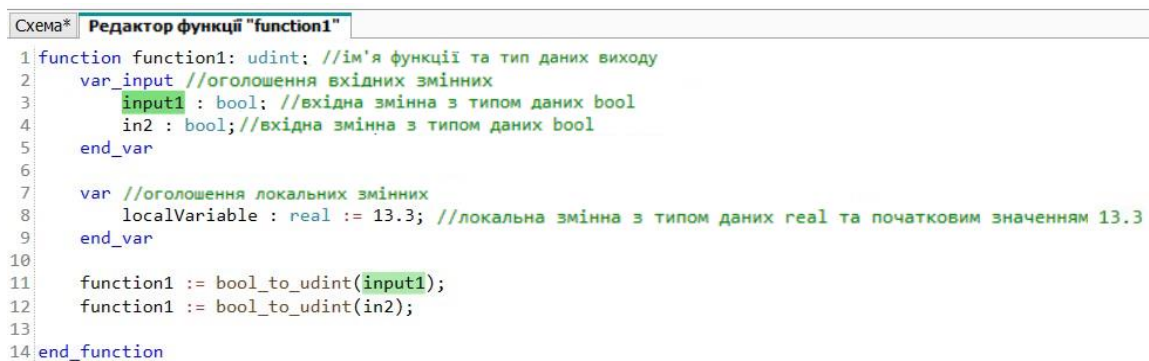


Рисунок 3.93

- Перемістити курсор на інший рядок у редакторі функцій. Змінна буде переіменована у всіх місцях використання.


Список помилок

В області відображення помилок формується таблиця з описом та розташуванням помилок, допущених у кодї. Для переходу на місце помилки слід двічі натиснути ЛКМ по рядку з помилкою.

Помилка	Рядок	Стовпець	Документ
Змінна Variable не оголошена	10	5	function1
Неправильні аргументи функції bool_to_udint	11	18	function1

Рисунок 3.94

4 Налаштування пристрою

Щоб відкрити вікно налаштувань пристрою, слід натиснути кнопку  на панелі інструментів або вибрати в головному меню **Прилад** → **Налаштування приладу**.

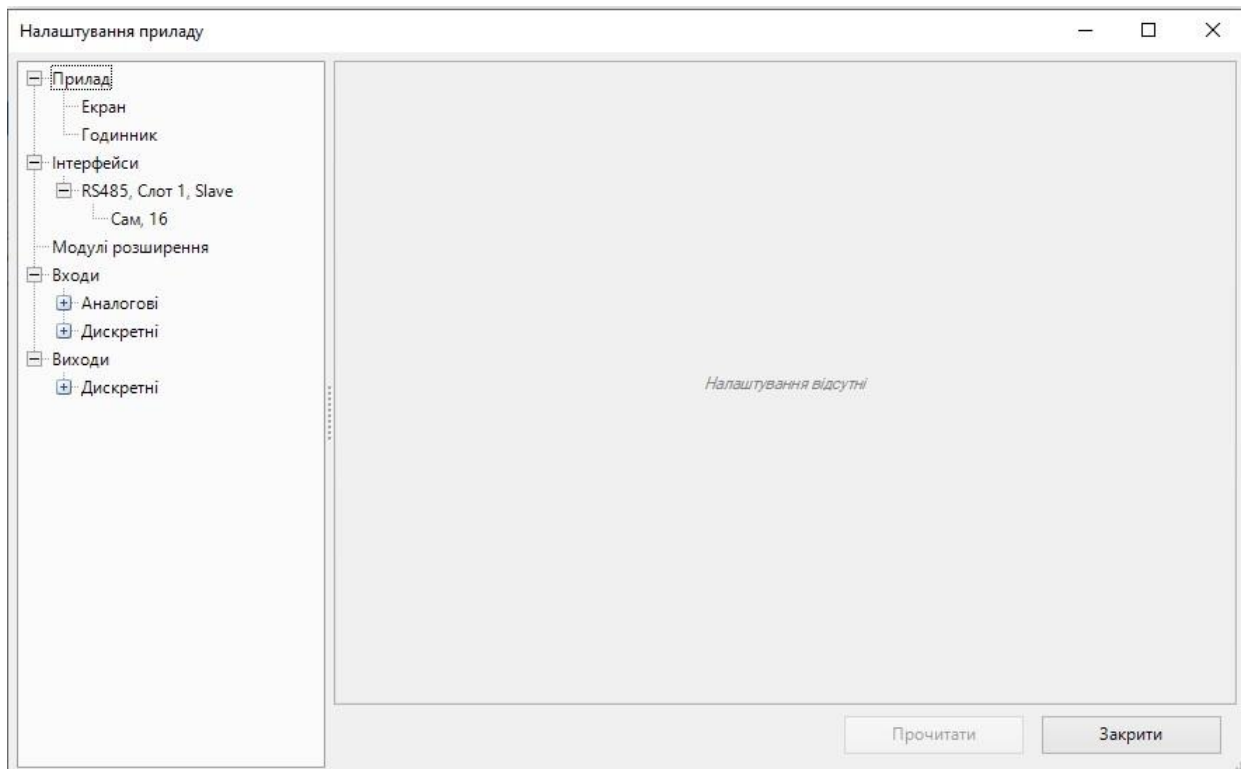


Рисунок 4.1

У лівій частині вікна налаштувань пристрою міститься дерево з розділами:

- [Екран 4.1](#);
- [Годинник 4.2](#);
- [Інтерфейси 4.3.1](#);
- [Модулі розширення 4.4](#);
- [Входи та виходи 4.5](#).



ПРИМІТКА

Перелік розділів може бути різним для різних моделей пристроїв.

У правій частині вікна параметрів пристрою відображаються налаштування вибраного розділу.

Усі встановлені налаштування для пристрою, за винятком годинника, зберігаються разом з проектом. Налаштування пристрою не потребує підключення пристрою.

4.1 Екран

Налаштування екрана доступні лише для пристроїв з дисплеєм.

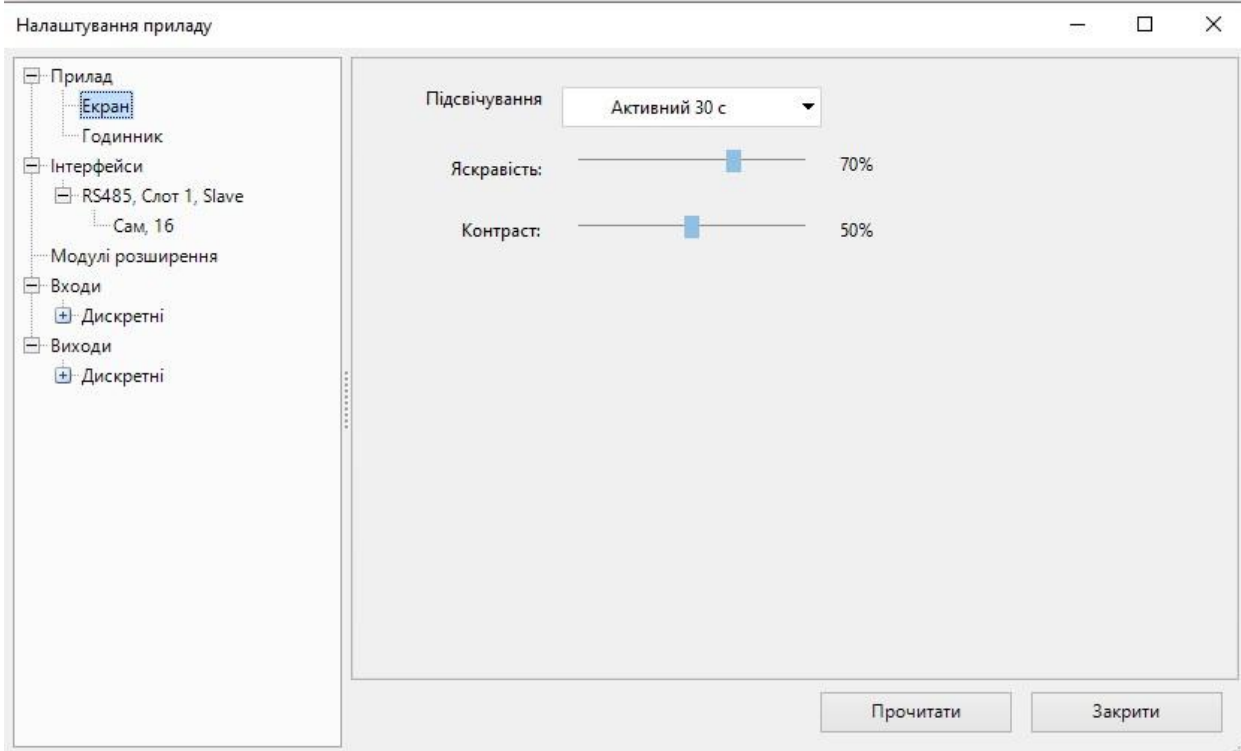


Рисунок 4.2

Перелік параметрів залежить від типу пристрою.

Підсвічування — у випадному меню можна вибрати тривалість роботи підсвічування з моменту останньої дії користувача.

Яскравість — налаштування яскравості екрана, від 0 до 100 %.

Контраст — налаштування контрастності символів, що відображаються, від 0 до 100 %.

Кнопка **Прочитати** використовується для зчитування поточних налаштувань екрана з підключеного пристрою.

4.2 Годинник

У розділі **Годинник** налаштовується годинник реального часу пристрою. Налаштування годинника реального часу потрібні для роботи системних змінних та ФБ програми.

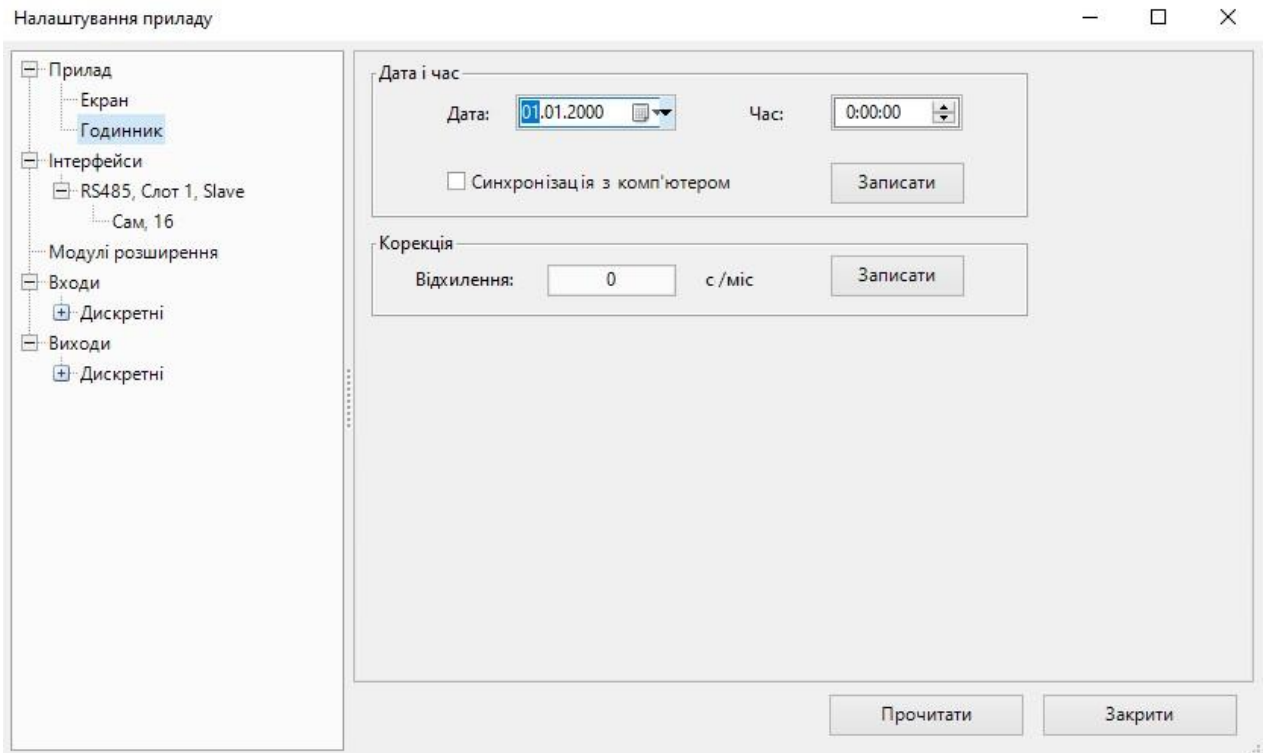


Рисунок 4.3

Дата та час

Дата та час задаються у відповідних полях.

Для синхронізації годинника реального часу пристрою з годинником ПК слід відзначити галочку **Синхронізація з комп'ютером** – поля **Дата** та **Час** стануть неактивними.

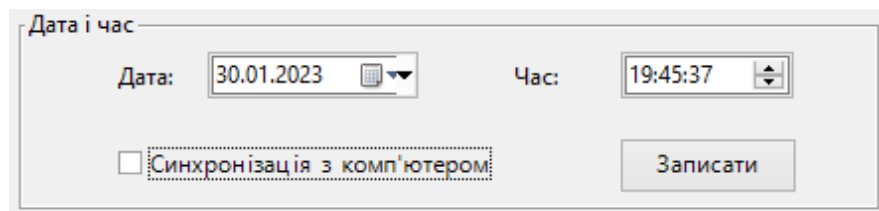


Рисунок 4.4

Щоб задати нові значення для годинника реального часу пристрою, слід натиснути кнопку **Записати**.

Корекція

Поле **Відхилення** слугує для вказівки похибки ходу годинника реального часу пристрою в секундах. Якщо годинник реального часу пристрою поспішає, то значення вказується зі знаком мінус. Коригування на зазначену величину відбувається рівномірно протягом місяця.

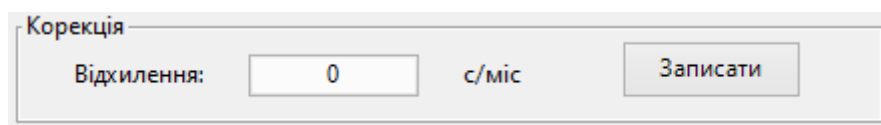


Рисунок 4.5

Щоб задати нове значення корекції для годинника реального часу пристрою, слід натиснути кнопку **Записати**.

Кнопка **Прочитати** використовується для зчитування поточних налаштувань годинника реального часу з підключеного пристрою.

4.3 Обмін по мережі

- [Інтерфейси 4.3.1](#);
- [Modbus 4.3.2](#)

4.3.1 Інтерфейси

У розділі **Інтерфейси** налаштовуються параметри інтерфейсів зв'язку з іншими пристроями в мережі.

Доступний такий інтерфейс: [RS-485 4.3.1.1](#).

Інтерфейси зв'язку працюють за протоколом [Modbus 4.3.2.1](#).

Додати інтерфейс

Щоб додати інтерфейс зв'язку, слід натиснути ПКМ на розділ **Інтерфейси** у дереві налаштувань та вибрати **Додати інтерфейс**.

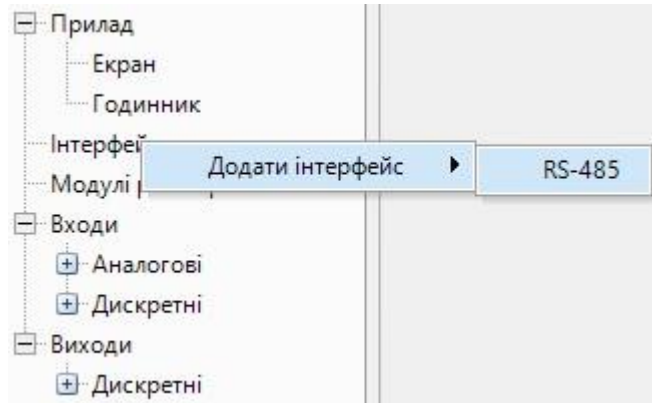


Рисунок 4.6

Інтерфейс буде додано до дерева з параметрами за умовчанням. Ім'я інтерфейсу складається з номера слота, що займається в пристрої, і режиму інтерфейсу.

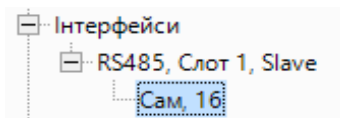


Рисунок 4.7

Кількість інтерфейсів зв'язку для пристрою не може перевищувати кількості слотів у пристрої.

Заміна/видалення інтерфейсу

Для заміни або видалення інтерфейсу слід натиснути ПКМ на імені інтерфейсу в дереві налаштувань та вибрати потрібну дію.

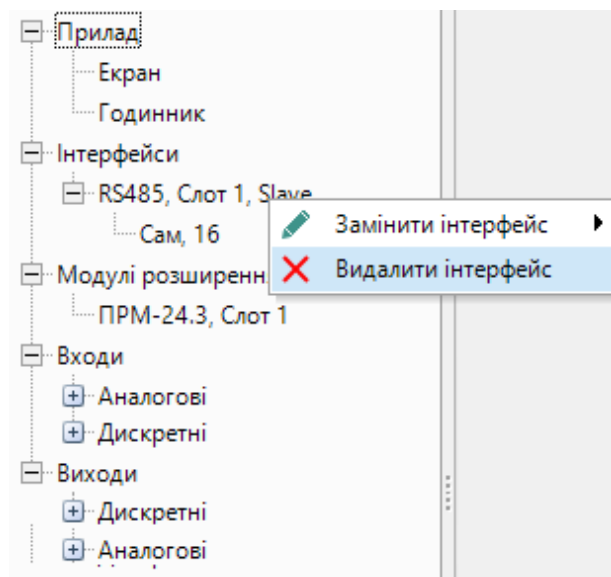


Рисунок 4.8

4.3.1.1 RS-485

Параметри інтерфейсу

Щоб змінити параметри інтерфейсу, його слід вибрати в дереві налаштувань. У правій частині вікна налаштувань пристрою відобразяться параметри інтерфейсу.

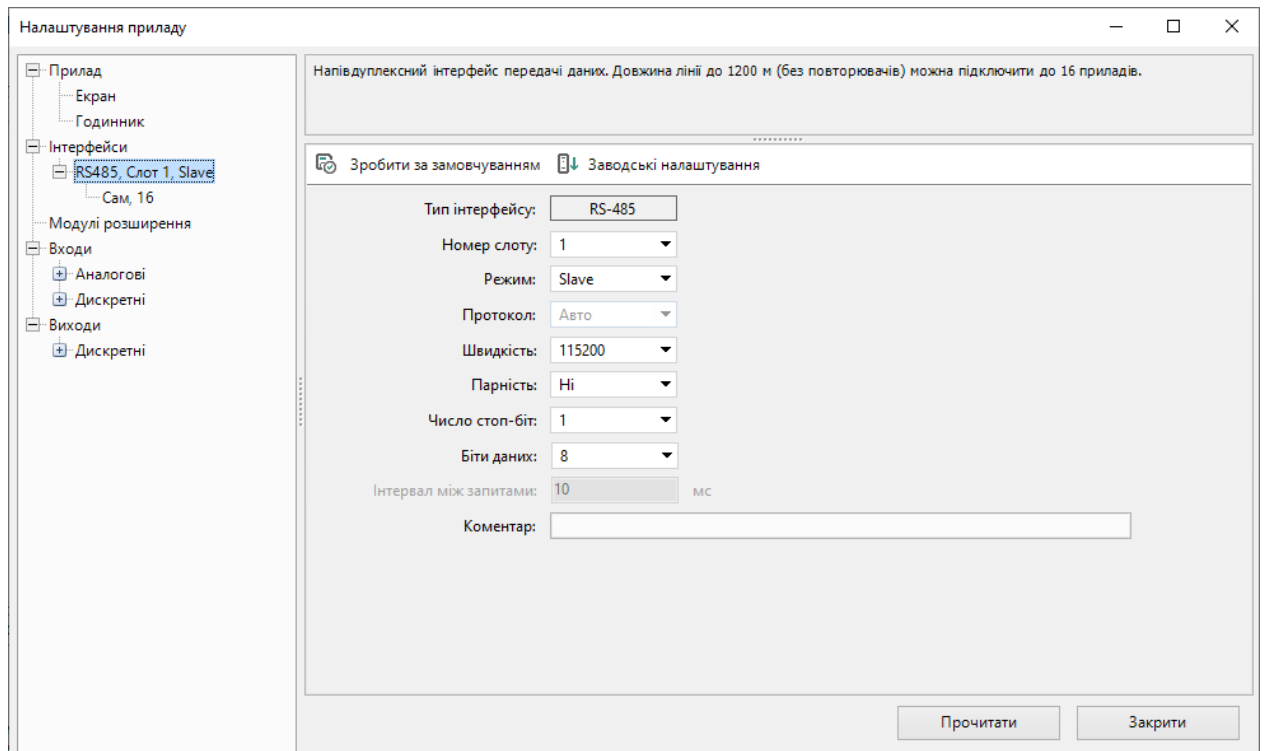




Рисунок 4.9

Номер слота визначає, який слот пристрою займає інтерфейс.

Режим інтерфейсу:

- [Slave 4.3.2.3](#) – пристрій є веденим, значення якого опитуються іншим пристроєм.
- [Master 4.3.2.2](#) – пристрій зчитує та записує значення ведених пристроїв, підключених за інтерфейсом зв'язку.

Кнопка  **Зробити за замовчуванням** — зберігає введені в поля дані як параметри за умовчанням.

Кнопка  **Заводські налаштування** — відображає в полях введення заводські налаштування для моделі підключеного пристрою.

Решта параметрів вводяться відповідно до завдань проекту та технічних характеристик інших пристроїв у мережі.



ПРИМІТКА

Параметри **Протокол** та **Інтервал між запитами** активні тільки у режимі Master.

Кнопка **Прочитати** зчитує поточні параметри підключеного пристрою.

Після задання параметрів інтерфейсу слід натиснути кнопку **Закрити**. Параметри набирають чинності після перезавантаження живлення пристрою або після завантаження програми в пристрій.

4.3.2 Modbus

- [Робота за протоколом Modbus 4.3.2.1](#);
- [Режим Master 4.3.2.2](#);
- [Режим Slave 4.3.2.3](#).

4.3.2.1 Робота за протоколом Modbus

Загальні відомості

Інтерфейси зв'язку працюють за протоколами Modbus-RTU (Master/Slave) або Modbus-ASCII (Master/Slave).

Для організації обміну даними у мережі через інтерфейс зв'язку необхідний Майстер мережі – пристрій у режимі Master. Основна функція Майстра мережі – ініціювати обмін даними.

На лінії зв'язку допускається наявність лише одного пристрою у режимі Master.

Особливості роботи у режимі Master

Якщо пристрій працює в режимі Master, для мінімізації часу отримання даних з усіх приладів рекомендується:

- якщо один або кілька Slave-пристроїв не підключені або недоступні, рекомендується передбачити в програмі блокування опитування цих пристроїв чи скоротити до мінімуму параметр **Таймаут відповіді** для цих пристроїв. В іншому випадку час опитування зростатиме пропорційно кількості непідключених пристроїв і величині встановленого параметра **Таймаут відповіді** для цих пристроїв;
- під час встановлення параметра **Інтервал між запитами** слід враховувати кількість Slave-пристроїв та загальну кількість запитів. Якщо час обробки всіх запитів займає більше часу, ніж встановлено параметром **Інтервал між запитами** цей параметр ігноруватиметься.

AQLogic дозволяє запрограмувати опитування до 16 пристроїв по одному інтерфейсу зв'язку. Кожен пристрій підтримує до 256 змінних. Можна використовувати однакові адреси та імена змінних для кожного пристрою.

Кількість інтерфейсів зв'язку

Кількість інтерфейсів зв'язку пристрою залежить від кількості слотів підключення. Якщо пристрій підтримує роботу до двох інтерфейсів зв'язку, їх можна налаштувати на роботу у двох незалежних мережах.

Особливості налаштування пристроїв з платою ПР-ИП485

Якщо пристрій з встановленою платою ПР-ИП485 налаштований працювати у режимі Master, рекомендується підключити підтягувальні резистори ліній зв'язку А та В за допомогою встановлення перемичок на платі ПР-ИП485. Для режиму Slave рекомендується відключення підтягувальних резисторів.

Підтягувальні резистори встановлюються для задання певного стану ліній зв'язку, коли у мережі немає передачі. Резистори встановлюються в одному місці мережі, як правило, біля пристрою в режимі Master.

Стандартні коди помилок Modbus

- **01** — прийнятий код функції не може бути оброблений;
- **02** — адреса даних, вказана у запиті, недоступна;
- **03** — значення в полі даних запиту є неприпустимою величиною.

Функції та області даних

При запиті Master звертається до однієї з **областей пам'яті** Slave за допомогою **функції**. **Область пам'яті** характеризується типом значень (біти/реєстри), що зберігаються в ній, і типом доступу (тільки читання/читання і запис).

Таблиця 4.1 – Області даних протоколу Modbus

Область даних	Позначення	Тип даних	Тип доступу
Coils (Регістри прапорів)	0x	Булевий	Читання/запис
Discrete Inputs (Дискретні входи)	1x	Булевий	Тільки читання

Продовження таблиці 4.1

Область даних	Позначення	Тип даних	Тип доступу
Input Registers (Регістри введення)	3x	Цілочисельний	Тільки читання
Holding Registers (Регістри зберігання)	4x	Цілочисельний	Читання/запис

Кожна область пам'яті складається з певної (залежної від конкретного пристрою) кількості комірок. Кожна комірка має унікальну адресу. Для конфігурованих пристроїв виробник надає **карту регістрів**, де міститься інформація про відповідність параметрів пристрою та їх адрес. Для програмованих пристроїв користувач формує таку карту самостійно за допомогою середовища програмування. Існують пристрої, в яких поєднуються обидва розглянуті випадки – у їх карти регістрів є фіксована частина і частина, яку користувач може доповнити відповідно до свого завдання.

У деяких пристроях області пам'яті накладені одна на одну (наприклад, **0x** та **4x**) – тобто користувач зможе звертатися різними функціями до тих самих регістрів.

Функція визначає операцію (читання/запис) та область пам'яті, з якої ця операція буде виконана.

Таблиця 4.2 – Основні функції протоколу Modbus

Код функції	Ім'я функції	Виконувана команда
1 (0x01)	Read Coil Status	Читання значень з кількох регістрів прапорів
2 (0x02)	Read Discrete Inputs	Читання значень з кількох дискретних входів
3 (0x03)	Read Holding Registers	Читання значень з кількох регістрів зберігання
4 (0x04)	Read Input Registers	Читання значень з кількох регістрів введення
5 (0x05)	Force Single Coil	Запис значення в один регістр прапора
6 (0x06)	Preset Single Register	Запис значення в один регістр зберігання
15 (0x0F)	Force Multiple Coils	Запис значень у кілька регістрів прапорів
16 (0x10)	Preset Multiple Registers	Запис значень у кілька регістрів зберігання

У різних документах ідентичні позначення можуть мати різне значення залежно від контексту. Наприклад, префікс **0x** часто використовують як вказівку на шістнадцяткову систему числення, тому в одному випадку **0x30** може позначати «30-й біт області пам'яті coils», а в іншому – «адреса 30 в шістнадцятковій (HEX) системі числення» (при цьому ця адреса може відноситися до будь-якої області пам'яті).

Опитування Slave може бути **одиначним** чи **груповим**. При **одиначному опитуванні** Master зчитує кожен з параметрів Slave окремою командою.

При **груповому опитуванні** Master зчитує однією командою відразу кілька параметрів, чиї адреси в карті регістрів розташовані послідовно і не мають розривів. Групове опитування дозволяє зменшити трафік у мережі та час, що витрачається на опитування пристрою, але в деяких випадках його застосування неможливе (або можливе з обмеженнями) через індивідуальні особливості пристрою.

Час опитування та тайм-аут відповіді

При коректно заданих параметрах регістри опитуються із заданим інтервалом. Якщо вказано менший період опитування, ніж загальний час обміну, коректний обмін за інтерфейсом буде порушено.

Якщо час тайм-ауту відповіді від Slave налаштований менше можливого, відповідь пристрою Slave бракується. Наступний запит закінчується невдачею і дані по інтерфейсу не надходять до регістру Master.

Порядок регістрів та порядок байт

Порядок прямування регістрів/байт важливий при зчитуванні системних змінних формату FLOAT (аналогові входи/виходи) пристрою в режимі Slave.

Змінні AQLogic під час роботи з Modbus мають такі особливості:

- цілочисельний тип (uint) – займає один регістр, налаштування слідування регістрів/біт не впливає на цей тип;
- булевський тип (bool) – займає один біт, можна задавати номер біта регістра;

- тип з рухомою комою (float) – займає два регістри, порядок проходження байт і регістрів важливий.

Автопідлаштування часу циклу програми

Пристрої можуть підлаштовувати час роботи циклу програми залежно від складності алгоритму. Автопідлаштування часу циклу програми впливає на роботу інтерфейсу зв'язку, оскільки запити обробляються в час циклу, що залишився після виконання програми.

Згідно з алгоритмом підлаштування часу циклу мінімальна кількість викликів Master складає до 50 разів на секунду. Якщо Master не встигає опитати всі пристрої, необхідно внести зміни в алгоритм для оптимізації кількості запитів.

Період опитування та правила його призначення

Пристрій усі запити ставить у чергу. Якщо черга коротка, то пристрій виконає всі цикли запиту-відповіді і зупиниться в очікуванні, поки не підійде до кінця заданий період. Якщо черга довга і не укладається в заданий період, то пристрій опитуватиме всі необхідні параметри з максимально можливим періодом, але цей період буде більше заданого в налаштуваннях.

Часова діаграма опитування:

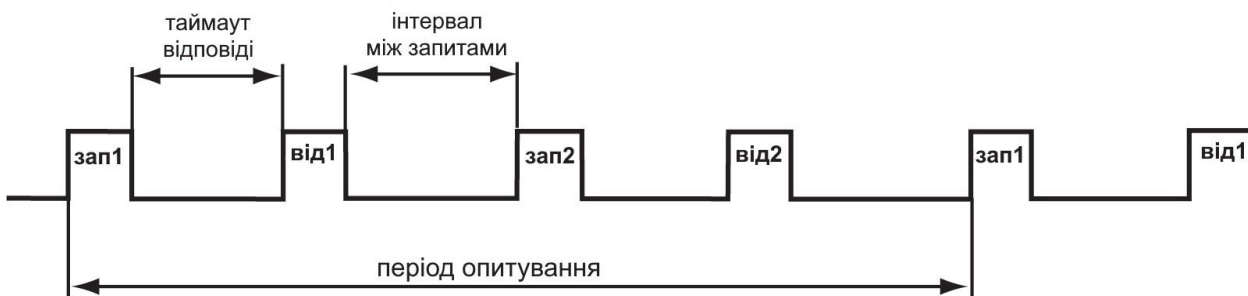


Рисунок 4.10

Для максимальної швидкості 115 200 біт/с опитування двох змінних типу FLOAT в «ідеальних» умовах (коротка лінія зв'язку, відсутність перешкод) в одного підлеглому пристрою слід встановити:

- час відповіді від початку запиту – 2,64 мс;
- наступний запит починається не раніше ніж через 12 мс;
- загальний час на запит 1 регістра FLOAT – 3,4 мс.

У перерахованих умовах пристрій зможе відправляти 83 запити в секунду. Це значення справедливе і для інших пристроїв з подібними часовими характеристиками. У процесі розробки алгоритму, коли логіка ускладнюється, збільшується час циклу і кількість запитів за секунду буде знижуватися.

Значення періоду опитування залежить від алгоритму, як часто та які параметри треба опитувати. Рекомендується виставляти період опитування рівним 1 с. У цьому випадку пристрій зможе опитати до 50 змінних.

Порядок опитування кількох пристроїв на шині

Пристрої опитуються відповідно до сформованої черги. Опитування проходить від найменшої адреси до найбільшої. У прикладі на рисунку нижче першим буде опитано пристрій з адресою 8, останнім – з адресою 32.

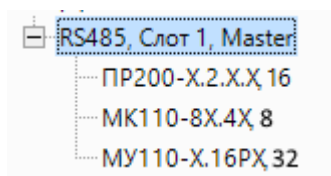


Рисунок 4.11

Можна встановити різний період опитування підлеглих пристроїв. Конкретне значення залежатиме від завдання.

Розрахунок адреси та біта для зчитування булевої змінної в режимі Master

У деяких випадках потрібно обчислити адресу регістра та номер біта підпорядкованого пристрою. Для визначення адреси регістра і номера біта з документації на пристрій, що опитується, береться розташування бітової змінної, наприклад, біт 1400, далі слід:

1. Обчислити адресу регістру: $1400 \div 16 = 87$.

2. Обчислити номер біта $1400 \bmod 16 = 8$.

Отримані адреси та номери вводяться в поля **Регістр** та **Біт** під час налаштування опитування пристрою.

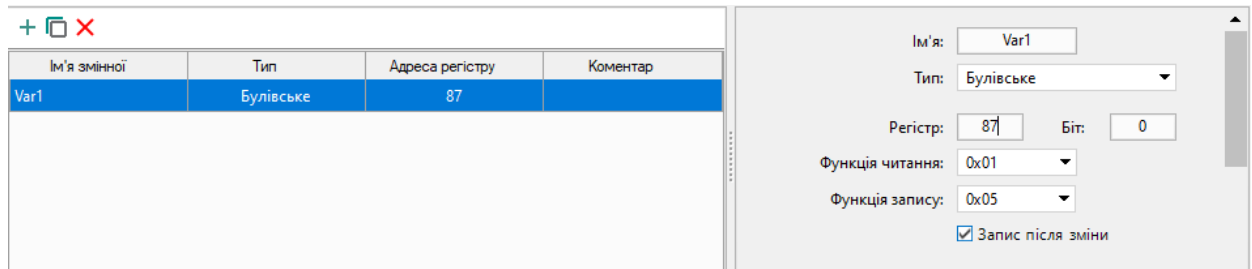


Рисунок 4.12

4.3.2.2 Режим Master

У режимі Master пристрій опитує підключені за інтерфейсом інші підпорядковані пристрої.



ПРИМІТКА

Для інтерфейсу в режимі Master групове опитування підпорядкованих пристроїв не підтримується.

Для опитування слід додати і налаштувати пристрої. Щоб додати пристрій, слід натиснути ПКМ на ім'я інтерфейсу і вибрати **Додати пристрій** в контекстному меню.

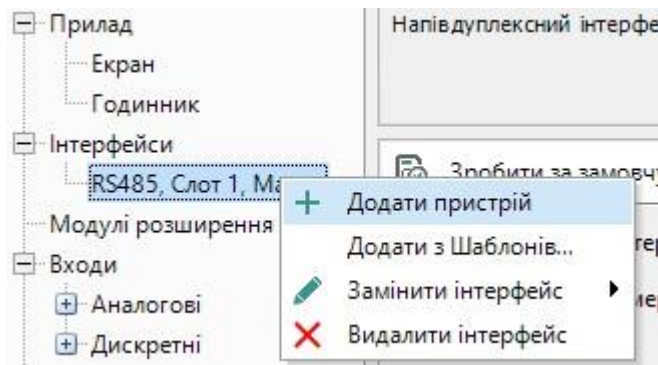


Рисунок 4.13

Доданий пристрій буде відображено в дереві як відгалуження інтерфейсу. Назва пристрою складається з імені та адреси пристрою.

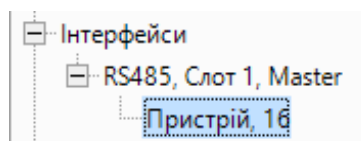


Рисунок 4.14

Параметри пристрою, який опитують

Для зміни параметрів пристрою слід натиснути на його назву в дереві налаштувань – у правій частині вікна налаштувань пристрою відобразяться параметри.

Ім'я:	<input type="text" value="Пристрій"/>	Адреса:	<input type="text" value="16"/>
Період опитування,	<input type="text" value="100"/>	Кількість спроб:	<input type="text" value="3"/>
Таймаут відповіді, мс:	<input type="text" value="100"/>		
Статус:	<input type="text" value=" < не вибрано > ..."/>	Опитування:	<input type="text" value=" < не вибрано > ..."/>
	<input type="checkbox"/> Старшим регістром уперед	<input checked="" type="checkbox"/> Старшим байтом уперед	
Float:	<input type="text" value="2"/> <input type="text" value="1"/>	<input type="text" value="4"/> <input type="text" value="3"/>	
Коментар:	<input type="text"/>		

Рисунок 4.15

Параметри пристрою, який опитують:

- **Ім'я** — ім'я пристрою для відображення у дереві налаштувань;
- **Адреса** — мережева адреса пристрою;
- **Період опитування, мс** — часовий інтервал, через який повторюється опитування. Допустимий діапазон від 0 до 65535 мс;
- **Таймаут відповіді, мс** — час, після вичерпання якого спроба опитування вважається невдалою. Допустимий діапазон від 0 до 65535 мс. Таймаут відповіді рекомендовано задавати в діапазоні 300–500 мс;
- **Кількість спроб** — кількість невдалих спроб опитування, при досягненні якої змінюється Статус пристрою. Допустимий діапазон від 0 до 255;
- **Статус** — зазначення булевської змінної, у яку записується статус пристрою: 1 — пристрій функціонує нормально, 0 — зв'язок з пристроєм втрачено. Змінна вибирається у [таблиці змінних 5](#), яку можна викликати натисканням кнопки «...»;
- **Опитування** — указання булевської змінної, яка визначає доступ до опитування: 0 — заборона опитування, 1 — дозвіл опитування. Змінна вибирається у [таблиці змінних 5](#), яку можна викликати натисканням кнопки «...»;

**ПРИМІТКА**

Параметри **Статус** та **Опитування** слугують для відображення та керування зв'язком пристрою з опитуваними приладами. Прив'язані змінні можна використовувати для візуалізації або усунення обриву зв'язку у логіці роботи програми.

- **Старшим регістром уперед** — визначає черговість послідовності регістрів під час роботи з дворегістровими змінними;
- **Старшим байтом уперед** — визначає черговість слідування байтів у послідовності;
- **Коментар** — текстовий опис пристрою.

Приклад

Слід зчитати число «-250.123».

Залежно від налаштування порядку слідування регістрів/байт воно буде передаватися по-різному.

Налаштування порядку слідування регістрів/байт	Число, що передається
<input type="checkbox"/> Старшим регістром уперед <input type="checkbox"/> Старшим байтом уперед Float: 2 1 4 3 	Молодший байт Старший байт Молодший байт Старший байт 01111101 00011111 01111010 11000011 Молодший регістр Старший регістр
Рисунок 4.16	Рисунок 4.17
<input type="checkbox"/> Старшим регістром уперед <input checked="" type="checkbox"/> Старшим байтом уперед Float: 2 1 4 3 	Старший байт Молодший байт Старший байт Молодший байт 00011111 01111101 11000011 01111010 Молодший регістр Старший регістр
Рисунок 4.18	Рисунок 4.19
<input checked="" type="checkbox"/> Старшим регістром уперед <input type="checkbox"/> Старшим байтом уперед Float: 2 1 4 3 	Молодший байт Старший байт Молодший байт Старший байт 01111010 11000011 01111101 00011111 Старший регістр Молодший регістр
Рисунок 4.20	Рисунок 4.21
<input checked="" type="checkbox"/> Старшим регістром уперед <input checked="" type="checkbox"/> Старшим байтом уперед Float: 2 1 4 3 	Старший байт Молодший байт Старший байт Молодший байт 11000011 01111010 00011111 01111101 Старший регістр Молодший регістр
Рисунок 4.22	Рисунок 4.23

Змінні пристрою, який опитують

У нижній частині вікна є таблиця змінних пристрою, який опитують.

Щоб додати змінну, слід натиснути кнопку . Змінну буде поміщено в таблицю, як на рисунку нижче.

Ім'я змінної	Тип	Адреса регістру	Коментар
Var1	Булівське	0	
Var2	Булівське	0	

Рисунок 4.24

Щоб створити декілька змінних зі схожими налаштуваннями, слід вибрати змінну в таблиці і натиснути кнопку **Тиражувати**.

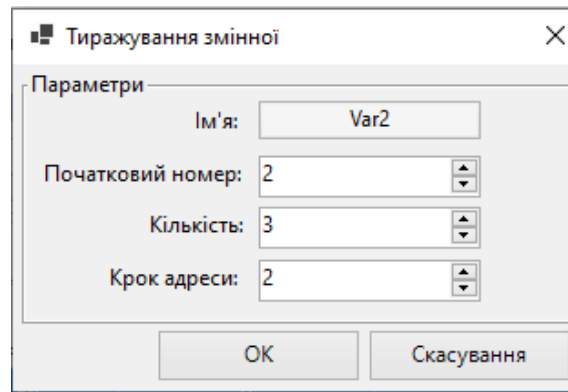


Рисунок 4.25

Налаштування тиражування змінної:

- **Ім'я** — ім'я змінної, яку буде тиражовано;
- **Початковий номер** — початкове число, яке буде додане до імені змінної, яку тиражують;
- **Кількість** — кількість змінних, які слід додати;
- **Крок адреси** — крок, з яким буде збільшуватися значення параметра **Регістр**.

Змінні будуть додані до таблиці.

Ім'я змінної	Тип	Адреса регістру	Коментар
Var1	Булівське	0	
Var22	Булівське	0	
Var23	Булівське	0	
Var24	Булівське	0	

Рисунок 4.26

Для видалення з таблиці слід виділити необхідну змінну і натиснути кнопку **X**.

Властивості змінної пристрою, який опитують

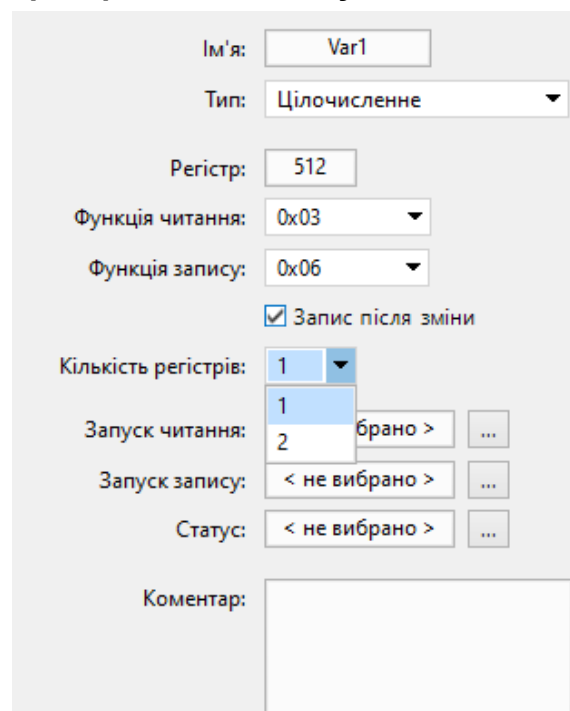


Рисунок 4.27

Властивості змінної пристрою, який опитують, налаштовуються справа від таблиці:

- **Ім'я змінної** — найменування для відображення у [таблиці змінних 5](#);
- **Тип** — [тип 5.1](#) змінної: булевський, цілочисельний або з рухомою комою;
- **Біт (тільки для змінних булевського типу)** — номер біта для читання;
- **Регістр/Адреса регістру** — значення регістра, до якого звертається пристрій, відображається у таблиці;
- **Функція читання/функція запису** — відключення або вибір функції запису/читання;



ПРИМІТКА

Читання за періодом — період читання встановлюється у налаштуваннях пристрою.
Запис за зміною — якщо цю функцію активовано, то у разі зміни значення змінної майстер ініціює запис значення змінної у Slave-пристрій.

- **Кількість регістрів (тільки для цілочисельних змінних)** — кількість регістрів, яку займає змінна: 1 чи 2;
- **Запуск читання** — призначається змінна булевського типу (командна змінна) для примусового читання змінної, яку налаштовують: якщо значення командної змінної змінюється з 0 на 1, майстер ініціює запит на читання, після успішного виконання запиту значення командної змінної автоматично змінюється з 1 на 0;
- **Запуск запису** — призначається командна змінна булевського типу (командна змінна) для примусового запису змінної, яку налаштовують: якщо значення командної змінної змінюється з 0 на 1, майстер ініціює запит на запис, після успішного виконання запиту значення командної змінної автоматично змінюється з 1 на 0;
- **Статус** — призначається цілочисельна змінна, у яку буде записаний код помилки [Modbus 4.3.2.1](#), у разі її появи;
- **Коментар** — текстовий опис змінної для відображення у [таблиці змінних 5](#);




ПРИМІТКА

Читання та запис за командою дозволяє примусово читати чи записувати значення змінної і більш гнучко налаштовувати читання та запис.
 Можна комбінувати всі варіанти читання та запису. Наприклад, можна налаштувати читання значень за періодом та за допомогою команди, а запис за зміною та за командою.

Шаблони мережевих пристроїв

Для інтерфейсів у режимі Master доступне створення та використання шаблонів мережевих пристроїв. Параметри та змінні налаштованого пристрою можуть бути збережені у вигляді шаблону.

Щоб зберегти шаблон, слід натиснути кнопку  **Зберегти пристрій як шаблон**. У вікні провідника, яке відкриється, слід вибрати розташування та ввести ім'я файлу. Файл буде збережено з розширенням *.dvtp. Збережений шаблон можна використовувати для інших інтерфейсів та проектів.

Для використання збереженого шаблону слід натиснути ПКМ на найменування інтерфейсу в дереві налаштувань і вибрати **Додати з Шаблонів...** У вікні провідника, яке відкриється, слід знайти розташування файлу і завантажити. Шаблон застосовується до інтерфейсу разом з усіма параметрами.

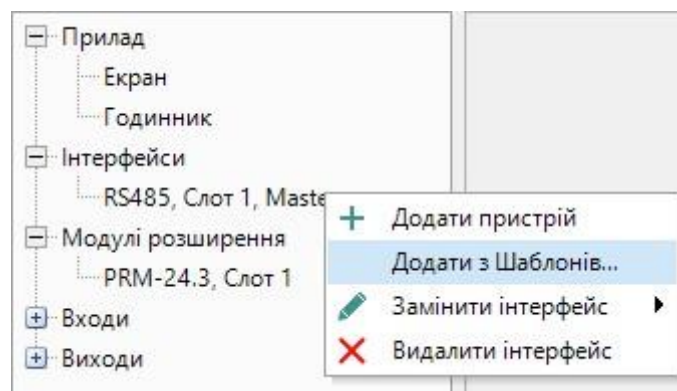


Рисунок 4.28

Для низки пристроїв виробництва АКУТЕК є готові шаблони, які можна завантажити з онлайн-бази за допомогою [Менеджера компонентів 3.10](#).

Конфігурування плати ПР-ИП485

Після конфігурування всіх параметрів в AQLogic слід перевірити положення джамперів на мережевій платі ПР-ИП485. Для роботи в режимі Master за інтерфейсом RS-485 джампери слід встановити в положення MST.

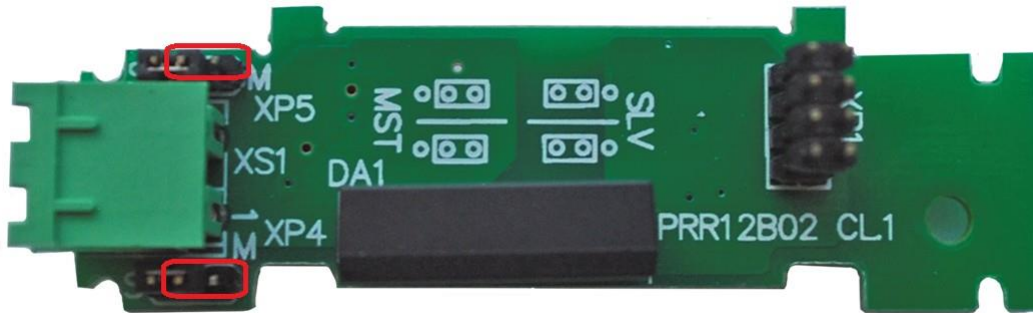


Рисунок 4.29

4.3.2.3 Режим Slave

У режимі Slave пристрій надає дані для зчитування іншим пристроям у мережі, самостійне опитування не веде.

Щоб налаштувати параметри передачі даних, слід натиснути на ім'я пристрою (за умовчанням — **Сам, 16**) під назвою інтерфейсу у дереві налаштувань.

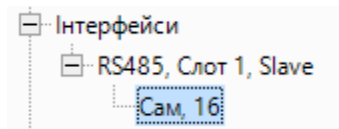


Рисунок 4.30

Параметри пристрою

Ім'я:	<input type="text" value="Сам"/>	Адреса:	<input type="text" value="16"/>
	<input type="checkbox"/> Старшим регістром уперед	<input checked="" type="checkbox"/> Старшим байтом уперед	
Float:	<input type="text" value="2"/> <input type="text" value="1"/>	<input type="text" value="4"/> <input type="text" value="3"/>	
Коментар:	<input type="text"/>		

Рисунок 4.31

У правій частині вікна налаштувань розташовані загальні для всіх змінних параметри пристрою:

- **Ім'я** — ім'я пристрою для відображення дерева налаштувань;
- **Адреса** — мережева адреса пристрою;
- **Старшим регістром уперед** — визначає черговість послідовності посилки регістрів під час роботи з двореєстровими змінними;
- **Старшим байтом уперед** — визначає черговість слідування байтів у посилці;
- **Коментар** — текстовий опис пристрою.

Для пристрою в режимі Slave допускається опитування до 64 регістрів при груповому опитуванні.


Змінні пристрою

У нижній частині вікна знаходиться таблиця змінних пристрою, значення яких будуть зчитуватися іншим пристроєм.

Щоб додати змінну, слід натиснути кнопку . Змінну буде поміщено в таблицю, як на рисунку нижче.

Ім'я змінної	Тип	Адреса регістру	Коментар
Var1	Цілочисленне	512	
Var2	Цілочисленне	513	

Рисунок 4.32

Щоб створити декілька змінних зі схожими налаштуваннями, слід вибрати змінну в таблиці і натиснути кнопку  **Тиражувати**.

Тиражування змінної ✕

Параметри

Ім'я:

Початковий номер: ▲▼

Кількість: ▲▼

Крок адреси: ▲▼

Рисунок 4.33


Налаштування тиражування змінної:

- **Ім'я** — ім'я змінної, яку буде тиражовано;
- **Початковий номер** — початкове число, яке буде додане до імені змінної, яку тиражують;
- **Кількість** — кількість змінних, яку потрібно додати;
- **Крок адреси** — крок, з яким буде збільшуватися значення параметра **Регістр**.

Змінні будуть додані до таблиці.

Ім'я змінної	Тип	Адреса регістру	Коментар
Var1	Цілочисленне	512	
Var2	Цілочисленне	513	
Var14	Цілочисленне	514	
Var15	Цілочисленне	516	
Var16	Цілочисленне	518	

Рисунок 4.34

Для видалення з таблиці слід виділити необхідну змінну і натиснути кнопку .

Властивості змінної пристрою

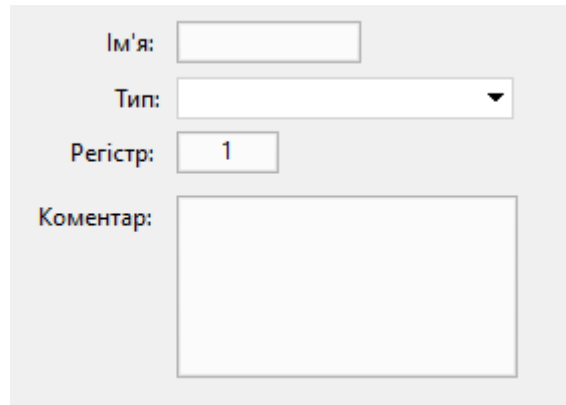


Рисунок 4.35

Властивості змінної пристрою налаштовуються праворуч від таблиці:

- **Ім'я** — ім'я змінної пристрою (задає користувач);
- **Тип** — [тип 5.1](#) змінної пристрою: цілочисельний або з рухомою комою;



ПРИМІТКА

Булевські змінні можна зчитати у цілочисельну змінну. Стан дискретного входу можна отримати за допомогою блоку [EXTRACT 6.1.5.1](#) або відповідного макросу з [Менеджера компонентів 3.10](#).

- **Регістр** — адреса регістру змінної пристрою (задає користувач). Діапазон доступних адрес вказаний у *настанові щодо експлуатування* пристрою;
- **Коментар** — текстовий опис значення змінної для відображення у [таблиці змінних 5.1](#).

4.4 Модулі розширення

У розділі **Модулі розширення** налаштовуються підключені до пристрою модулі розширення вводу/виводу.

Додати модуль розширення

Щоб додати модуль розширення, слід натиснути ПКМ на розділ **Модулі розширення** у дереві налаштувань і вибрати **Додати модуль розширення** → **< Модель модуля розширення >**.

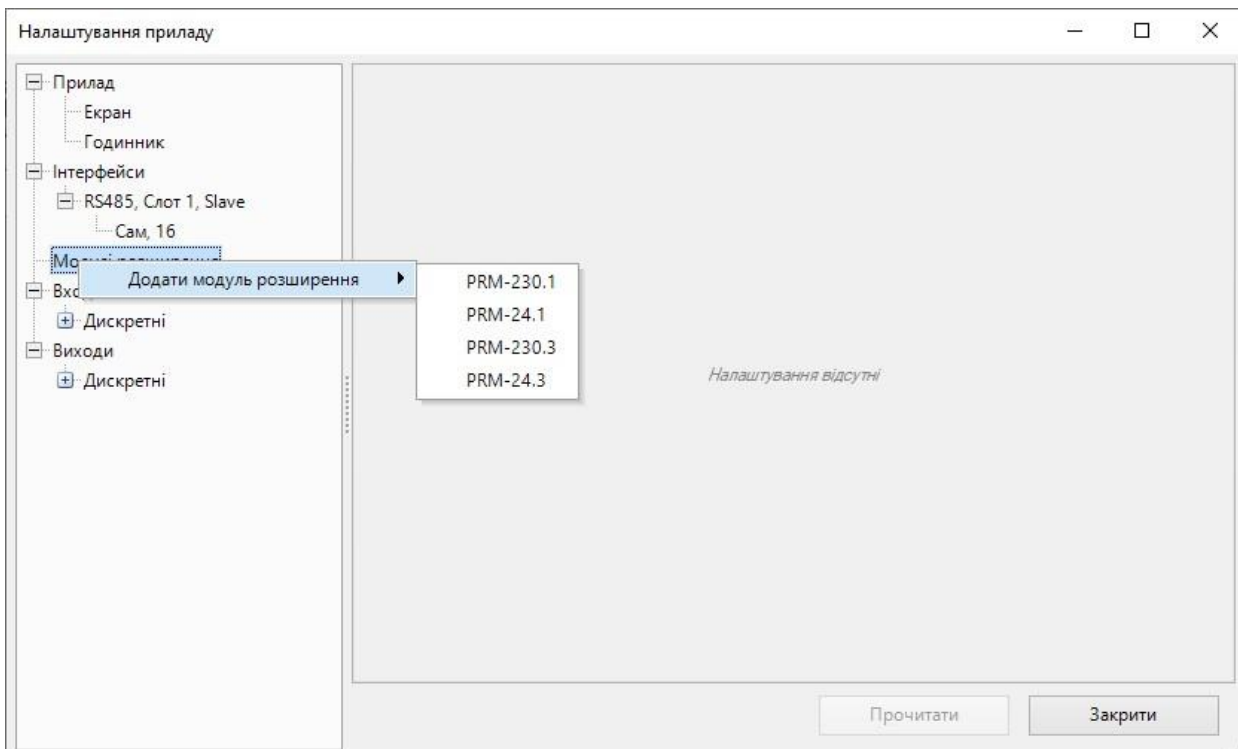


Рисунок 4.36

Модуль розширення відобразатиметься як відгалуження розділу в дереві налаштувань. Найменування модуля розширення складається з моделі модуля та номера зайнятого слота.

Модуль розширення можна видалити, натиснувши ПКМ на його найменування у дереві налаштувань та вибравши відповідну функцію.

Параметри модуля розширення

Параметри виділеного модуля розширення відображаються у правій частині вікна налаштувань.

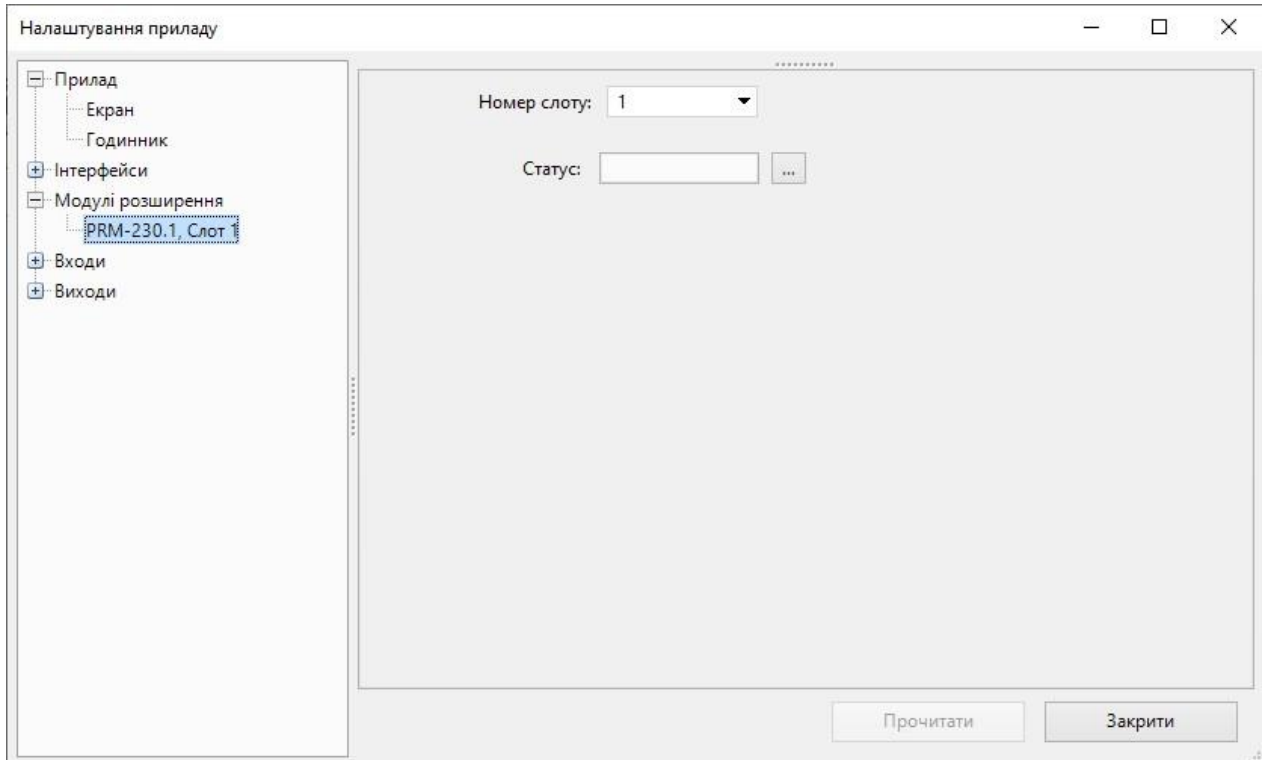


Рисунок 4.37

У випадковому меню **Номер слоту** вказується номер слота в пристрої, який займає модуль розширення. Кількість модулів розширення не може перевищувати кількості слотів у пристрої.

Статус підключення модуля розширення може зберігатися в змінну булевського типу. Якщо зв'язок з модулем встановлено, змінна зберігає значення **True**, якщо зв'язок не встановлено — **False**. Для вибору змінної, в якій зберігатиметься статус, слід натиснути кнопку «...».

Відкриється [таблиця змінних 5](#), де можна вибрати змінну з проекту або створити нову.

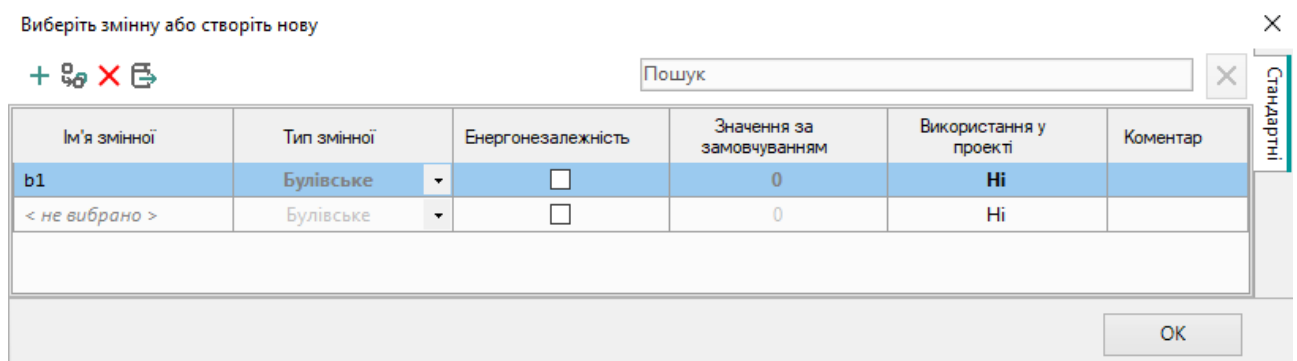


Рисунок 4.38

Вибрана змінна відобразатиметься у полі введення **Статус**.

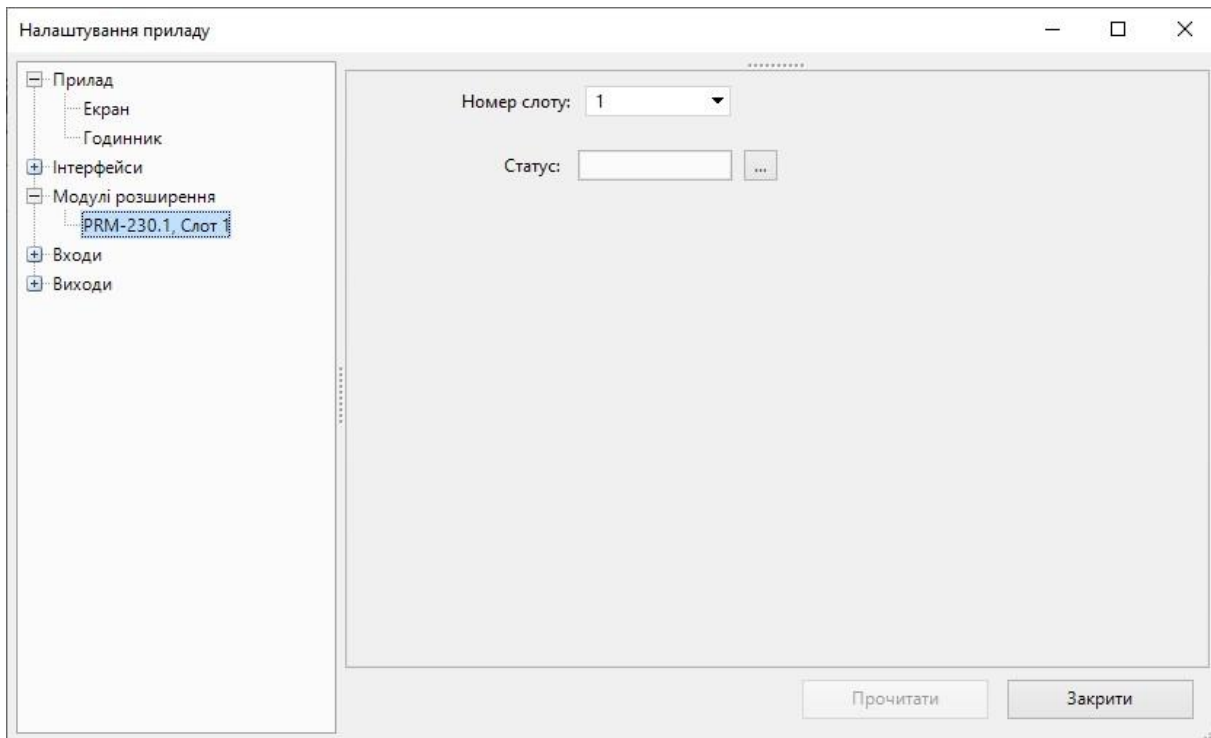


Рисунок 4.39

Для виходів модуля розширення може бути налаштовано [безпечний стан](#).

4.5 Входи та виходи

У розділах **Входи** та **Виходи** налаштовуються параметри і коментарі для входів/виходів пристрою та модулів розширення.

Параметри залежать від типу входу/виходу вибраного пристрою.

Входи/виходи модулів розширення відображаються у дереві налаштувань із номером слота, який займають, у дужках.

Для правильного налаштування входів/виходів пристрою слід звернутися до *настанови щодо експлуатування* пристрою.

Додавання коментарів

Коментар для входу/виходу можна написати в полі **Коментар** або на панелі властивостей.

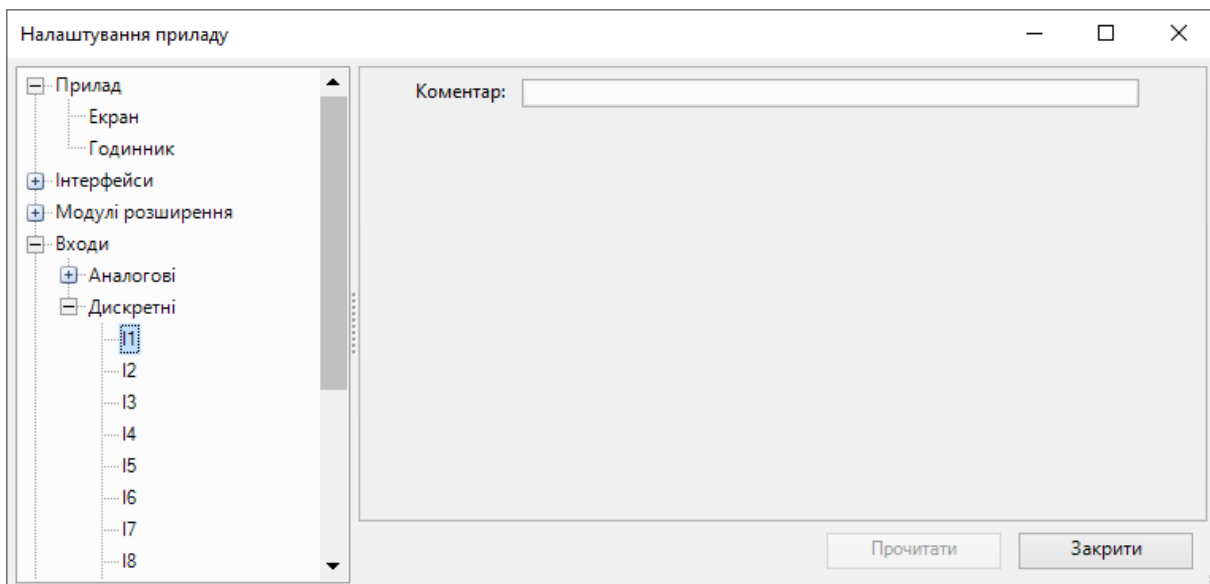


Рисунок 4.40

Текст введеного коментаря відобразиться у спливної підказці під час наведення курсору на вхід/вихід на схемі:

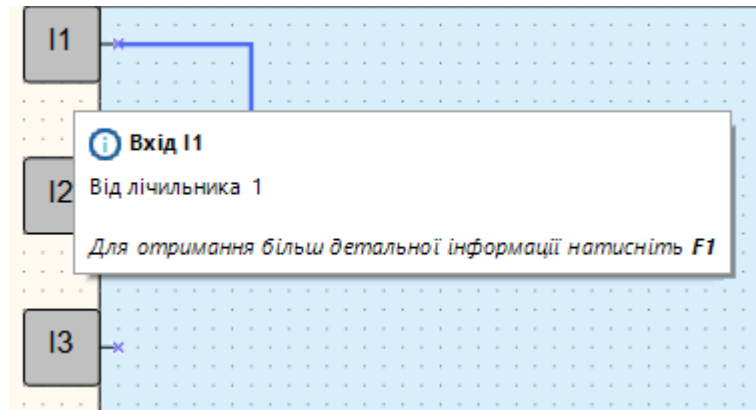


Рисунок 4.41

Безпечний стан виходів модулів розширення

Для виходів модулів розширення можна налаштувати безпечний стан. Вихід прийматиме безпечний стан у разі втрати зв'язку з пристроєм.

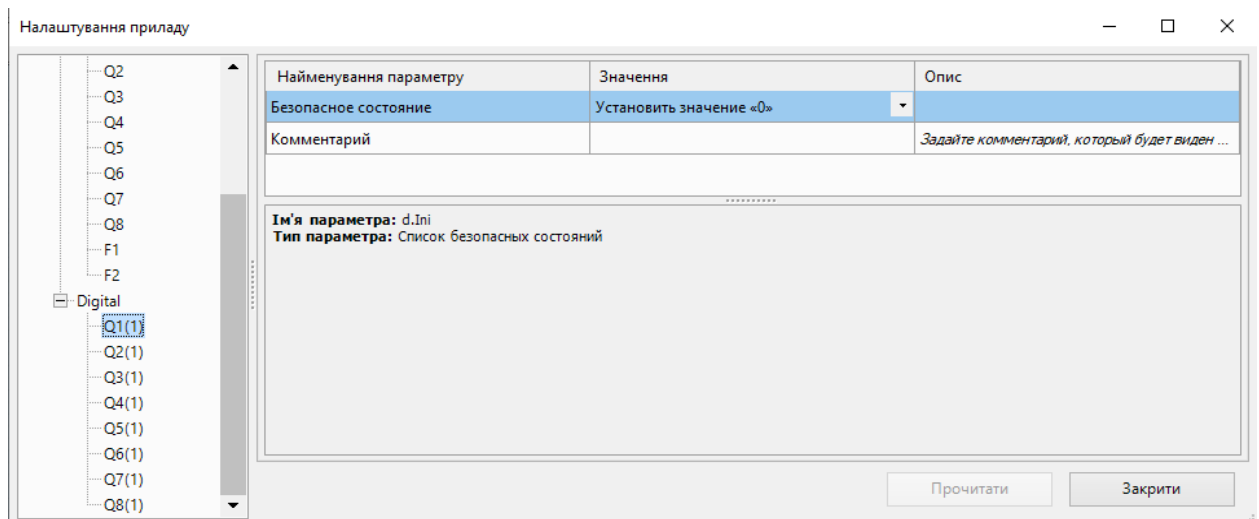



Рисунок 4.42

5 Змінні

Змінні слугують для запису і зчитування значень на схемі проекту та при програмуванні [екранів 3.4](#). Щоб використовувати змінну у проекті, спочатку її слід створити у таблиці змінних.

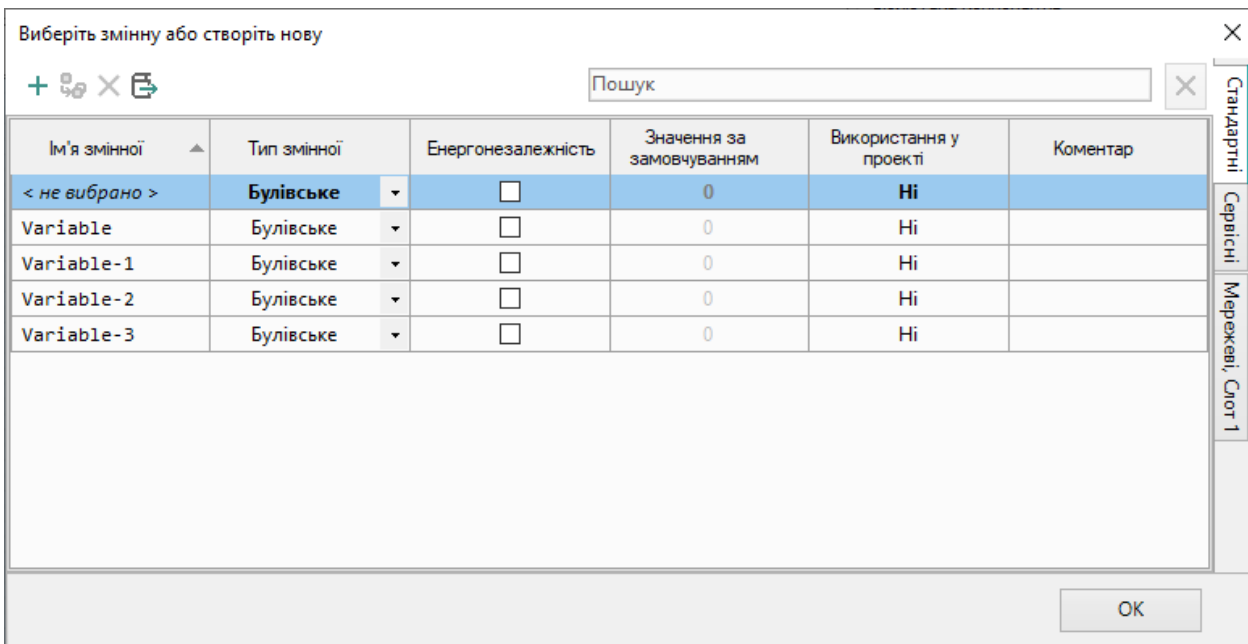
Щоб запустити Таблицю змінних, натисніть кнопку  на панелі інструментів або виберіть у головному меню **Прилад** → **Таблиця змінних**.

Таблиця змінних містить бічні вкладки:

- **Змінні**;
- [Сервісні змінні 5.2](#);
- [Змінні інтерфейсу зв'язку 5.3](#) (для кожного інтерфейсу зв'язку окрема вкладка).


Змінні

Змінні проекту відображаються у таблиці змінних.




Ім'я змінної	Тип змінної	Енергонезалежність	Значення за замовчуванням	Використання у проекті	Коментар
< не вибрано >	Булівське	<input type="checkbox"/>	0	Ні	
Variable	Булівське	<input type="checkbox"/>	0	Ні	
Variable-1	Булівське	<input type="checkbox"/>	0	Ні	
Variable-2	Булівське	<input type="checkbox"/>	0	Ні	
Variable-3	Булівське	<input type="checkbox"/>	0	Ні	

Рисунок 5.1

Для створення змінної слід ввести ім'я та встановити [тип 5.1](#) або натиснути кнопку  в лівій верхній частині таблиці. Інші параметри заповнюються залежно від експлуатаційної потреби:

- **Ім'я змінної** – ім'я для відображення на панелі Змінні та на блоці змінної у проекті;
- **Тип змінної** – може бути призначений один з трьох типів: булевський, цілочисельний, з рухомою комою. Докладніше див. розділ [Типи змінних 5.1](#);
- **Енергонезалежність** – для збереження значення змінної в ПЗП пристрою у разі відключення електроживлення слід поставити галочку в полі налаштування. Час зберігання змінної у ПЗП залежить від типу пристрою, див. *настанову щодо експлуатування* пристрою;
- **Значення за замовчуванням** – значення, яке зберігатиме змінна, доки в неї не буде записано нове значення;
- **Використання у проекті** – індикація прив'язки до блоків у програмі. Якщо змінна прив'язана, то значення – **Так**;
- **Коментар** – текстовий опис для відображення у спливіній підказці на схемі проекту при наведенні курсору на блок змінної.

Щоб видалити змінну з таблиці, слід натиснути ПКМ на рядок змінної та у контекстному меню вибрати **Видалити змінну** або натиснути кнопку  в лівій верхній частині таблиці.

Для дублювання змінної слід натиснути ПКМ на рядок змінної та у контекстному меню вибрати

Дублювати змінну або натиснути кнопку  в лівій верхній частині таблиці.

Таблиця змінних підтримує пошук та фільтрацію за іменем. Символ «Пробіл» виконує роль логічного АБО та дозволяє фільтрувати змінні за кількома критеріями.

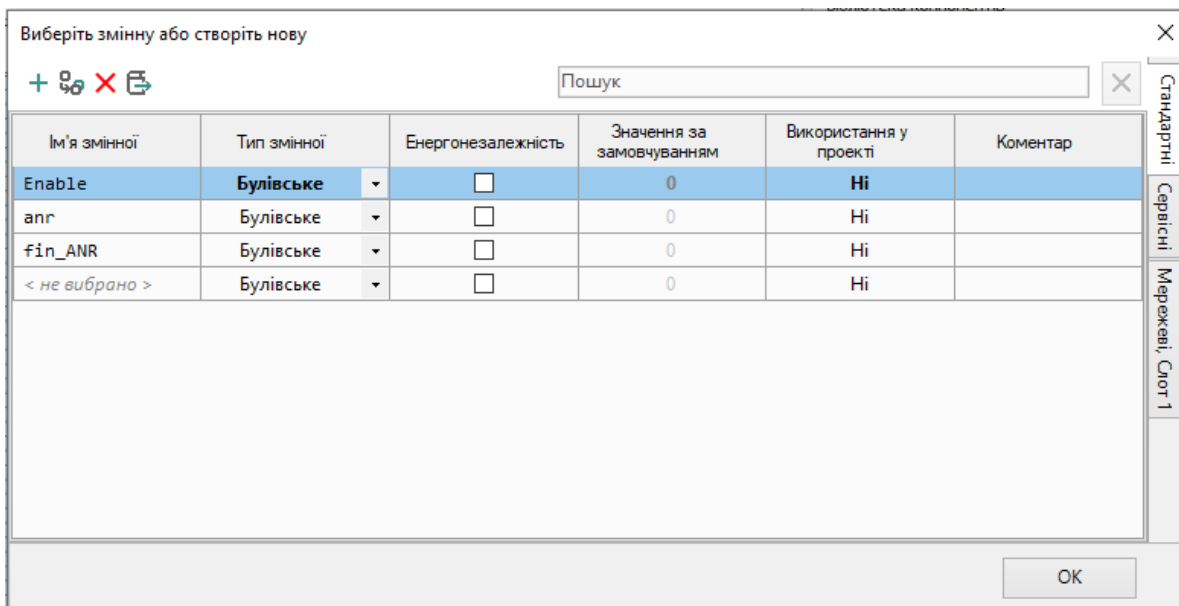



Рисунок 5.2

Якщо введеної у рядку пошуку змінної немає, її можна створити натисканням на кнопку  у лівій верхній частині таблиці. Змінній автоматично присвоюється введене у рядку пошуку ім'я.

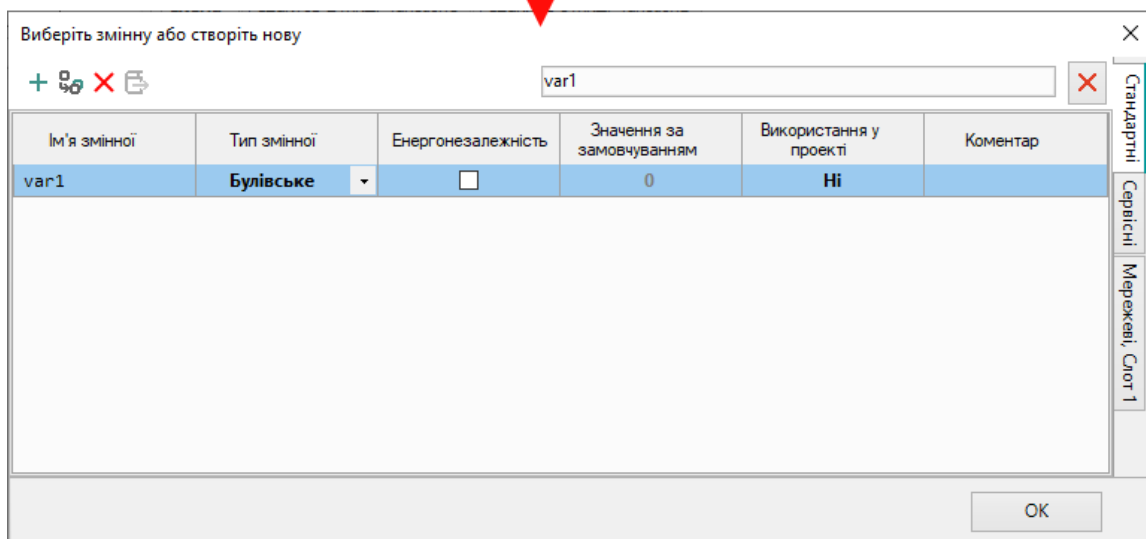
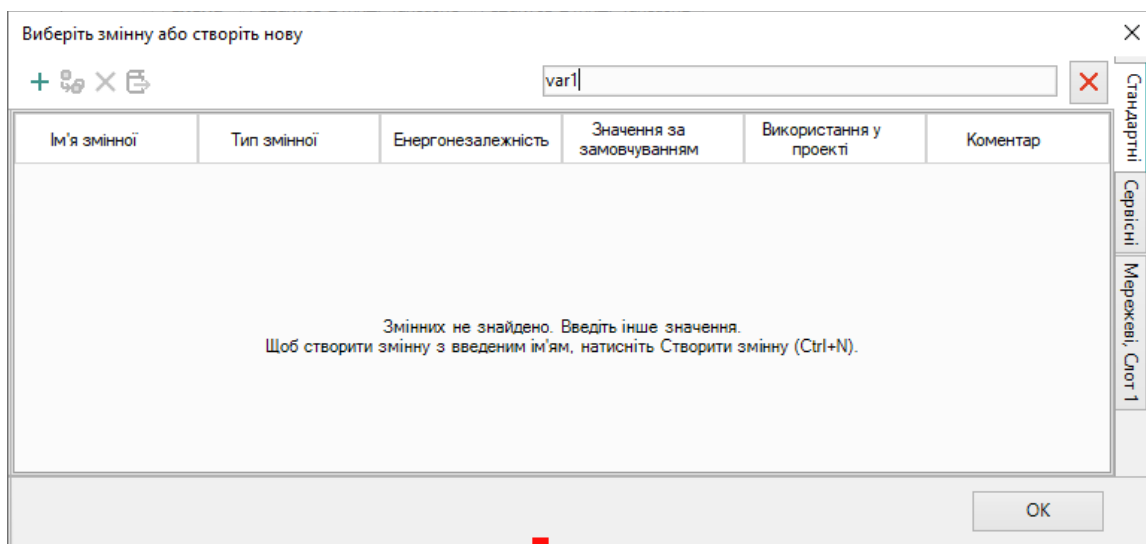


Рисунок 5.3

Таблиця змінних допускає сортування за значеннями. Для сортування значень слід натиснути назву колонки таблиці. Повторне натискання сортує колонку у зворотному порядку.

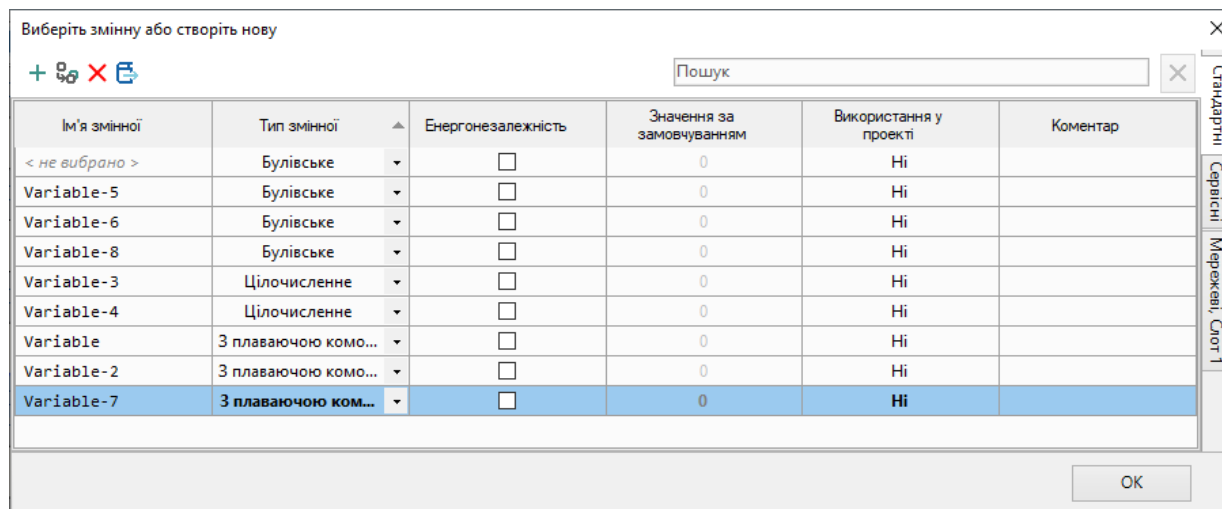


Рисунок 5.4

Щоб знайти місця використання змінної у проєкті, слід натиснути ПКМ на рядок змінної у таблиці та вибрати **Показати посилання**.

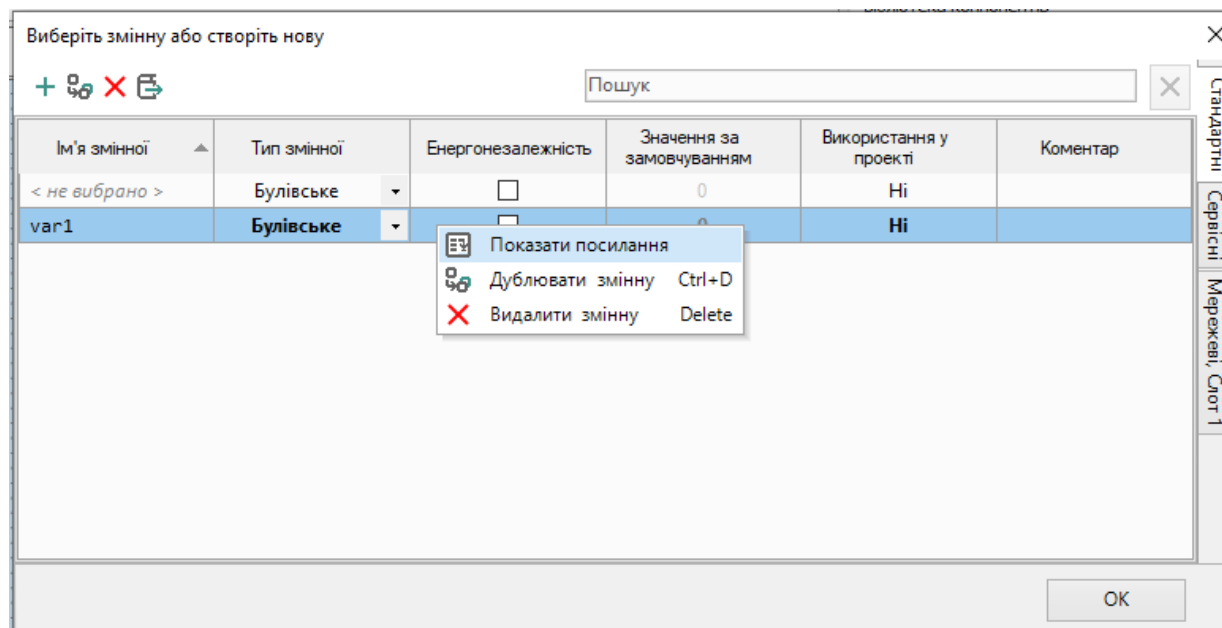


Рисунок 5.5

У вікні, що відкрілося, з'явиться інформація про обрану змінну.

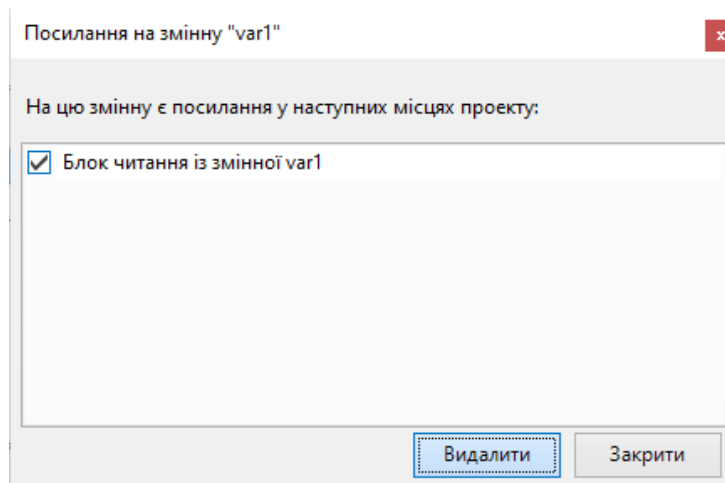


Рисунок 5.6

Щоб видалити прив'язку до блоку змінної у проекті, слід відзначити потрібні змінні галочками та натиснути кнопку **Видалити**. Змінна залишиться в таблиці, але не використовуватиметься у проекті.

Експорт змінних у файл

Можна експортувати вкладку змінних як таблицю у форматі **.csv**. Для цього слід:

1. Натиснути **Експортувати вкладку змінних у файл CSV** у лівій верхній частині таблиці.
2. У вікні, що відкрилося, вказати місце вивантаження файлу.
3. Натиснути кнопку **Зберегти**.



ПРИМІТКА

Ім'я файлу формується залежно від вкладки, що експортується, за схемою **Ім'яПроекту_Вкладка_Змінні**.

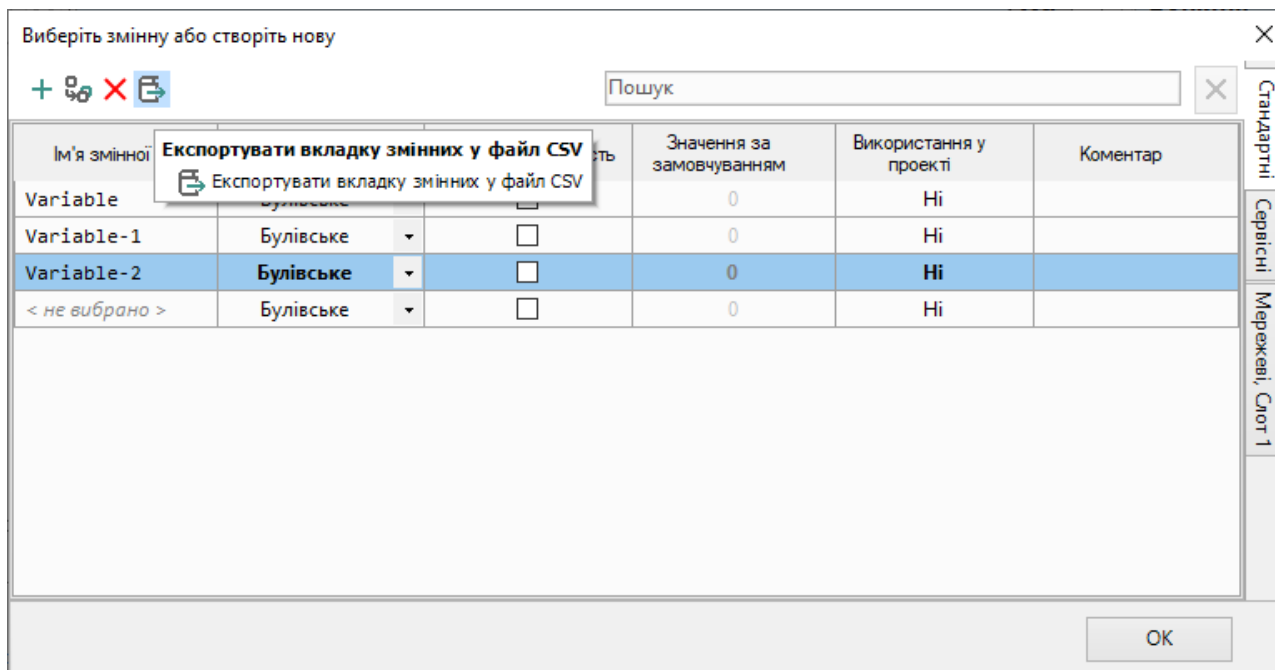


Рисунок 5.7

5.1 Типи змінних

В AQLogic використовуються три типи змінних:

- булевський (двійковий);
- цілочисельний;
- з рухомою комою (дійсний).

Значення від однієї змінної до іншої можуть передаватися тільки при збіганні типів змінних.

Булевський тип

Змінна булевського типу може набувати одне з двох значень: 1 (True) або 0 (False).

На схемі змінні булевського типу з'єднуються чорними лініями:

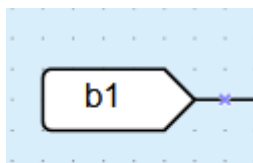


Рисунок 5.8

Цілочисельний тип

Змінна цілочисельного типу може приймати значення цілого числа в діапазоні від 0 до 4294967295.

На схемі змінні цілочисельного типу з'єднуються червоними лініями:

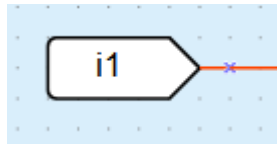


Рисунок 5.9

Тип з рухомою комою

Змінна типу з рухомою комою може набувати значення дійсного числа в діапазоні від $-3,402823E+38$ до $3,402823E+38$.

Числа одинарної точності з рухомою комою забезпечують відносну точність 7–8 десяткових цифр у діапазоні від 10^{-38} до приблизно 10^{38} .

На схемі змінні типу з рухомою комою з'єднуються фіолетовими лініями:

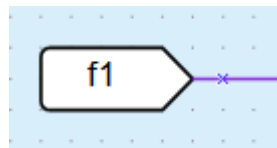


Рисунок 5.10

5.2 Сервісні змінні

Сервісні змінні пов'язані з налаштуванням пристрою і можуть лише зчитуватися. Запис до сервісних змінних недоступний. Набір сервісних змінних може відрізнятися залежно від пристрою.

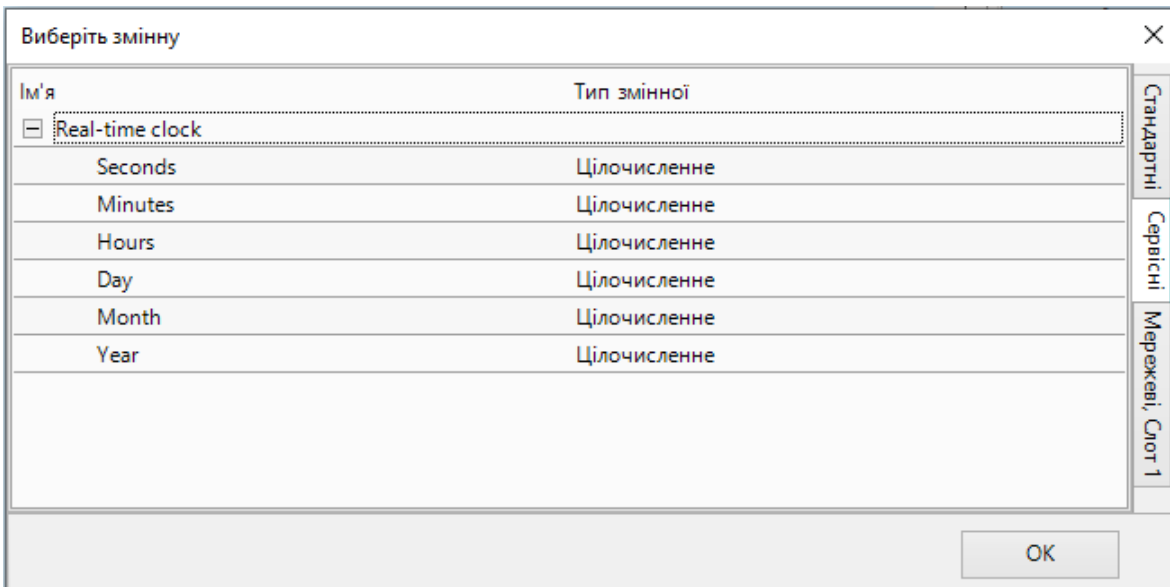


Рисунок 5.11

На схемі сервісні змінні позначаються сірим кольором:

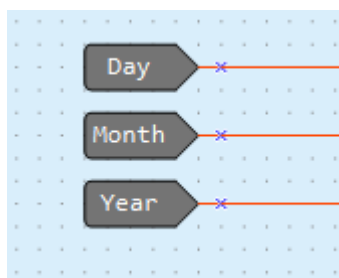


Рисунок 5.12

5.3 Змінні інтерфейсу зв'язку

У таблиці змінних для кожного інтерфейсу зв'язку створюється окрема вкладка з описом мережевих змінних. У заголовку вкладки вказується тип інтерфейсу та номер слота, який займається.

Режим Master

Для інтерфейсу зв'язку в режимі Master таблиця містить вкладки з мережевими змінними для кожного опитуваного пристрою.

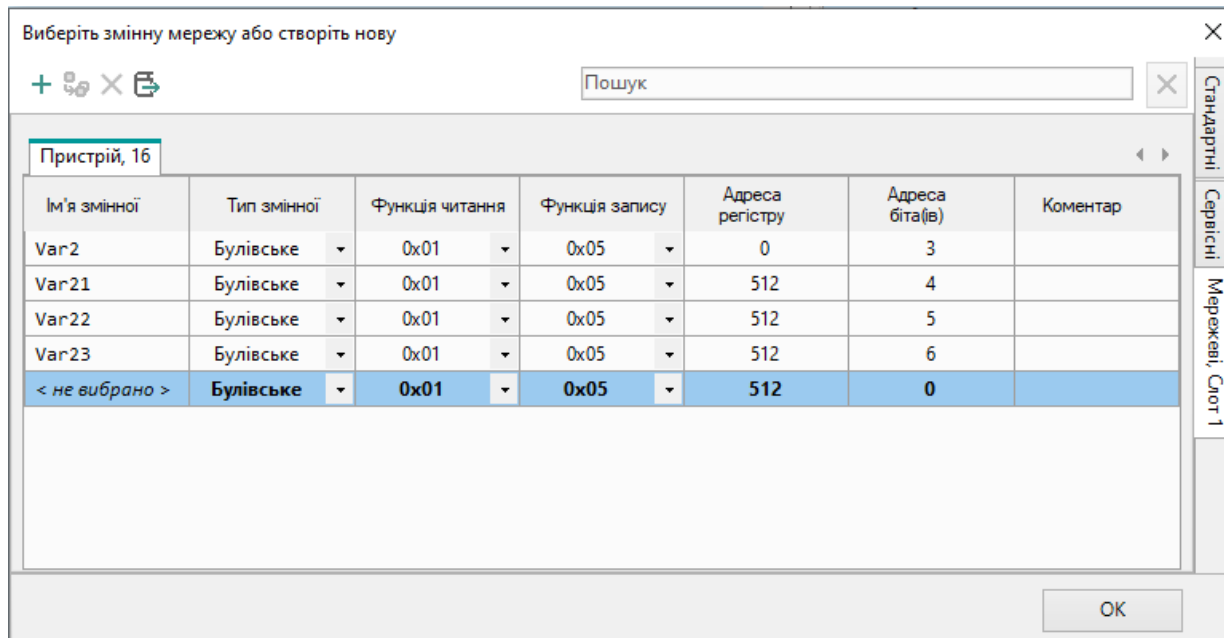


Рисунок 5.13

Додаткові відомості про налаштування змінних див. у розділі [Режим Master 4.3.2.2](#).

Режим Slave

Для інтерфейсу зв'язку в режимі Slave таблиця відображає зчитувані мережеві змінні.

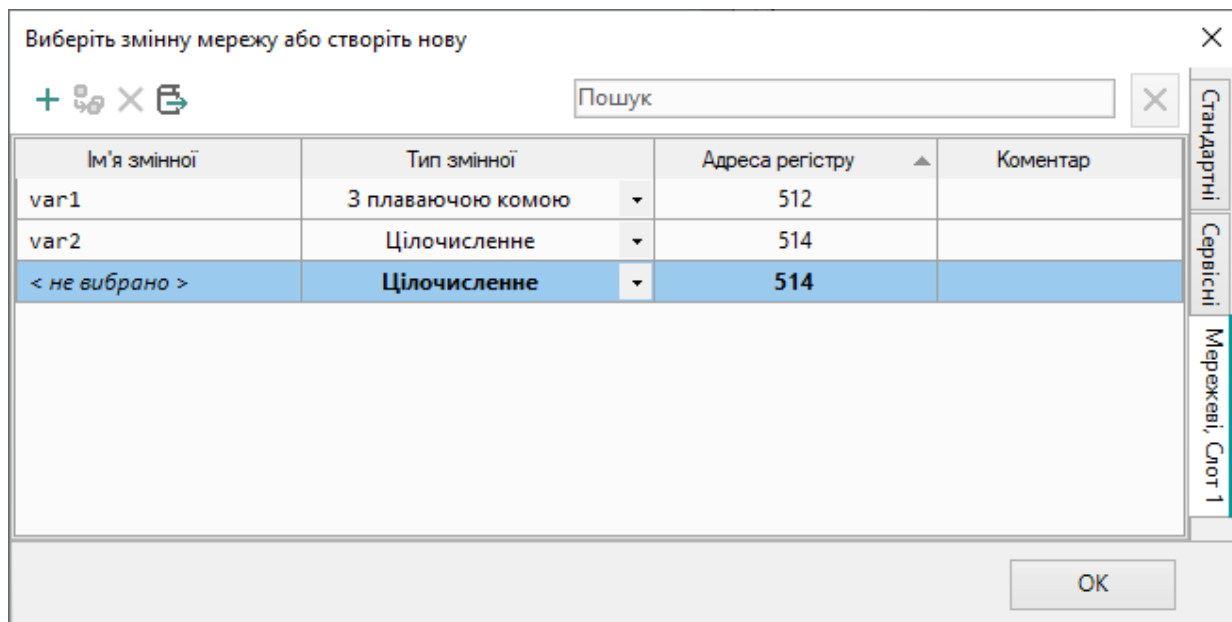




Рисунок 5.14

Додаткові відомості про налаштування змінних див. у розділі [Режим Slave 4.3.2.3](#).

5.4 Копіювання змінних

Змінні можна копіювати з проекту до проекту для повторного використання та скорочення часу на розробку. Для копіювання змінної слід виділити блок зі змінною у вихідному проекті та натиснути кнопку  на панелі інструментів або вибрати команду **Копіювати** у контекстному меню блоку.

Змінна вставляється в інший проект натисканням кнопки  на панелі інструментів або за допомогою вибору команди **Вставити** в контекстному меню полотна.

Змінні

Під час вставки скопійованого блоку змінної у проект перевіряються всі властивості змінних. Якщо в проекті є змінна з властивостями, що повністю збігаються, то вона дублюється в блок. Змінні з властивостями, що відрізняються, автоматично створюються в проекті. Якщо імена змінних збігаються, але інші властивості відрізняються, то створюється нова змінна, але конфлікт імен відображено таблиці змінних, і потрібно змінити ім'я однієї зі змінних.



ПРИМІТКА

У проект для пристрою, який не підтримує тип даних з рухомою комою, неможливо вставити змінні типу з рухомою комою.



ПРИМІТКА

Енергонезалежні змінні неможливо вставити в проект для пристрою, який їх не підтримує.

Сервісні змінні

У проекті для пристрою, який не підтримує годинник реального часу, неможливо скопіювати сервісні змінні.

Змінні інтерфейсу зв'язку

В інший проект можна копіювати лише змінні інтерфейсу зв'язку в режимі Slave для ідентичного номера слота. Змінні інтерфейси зв'язку в режимі Master слід створювати вручну. Під час вставки скопійованого блоку мережної змінної у проект перевіряються всі властивості змінних. Якщо в проекті є змінна з властивостями, що повністю збігаються, то вона дублюється в блок. Змінні з властивостями, що відрізняються, автоматично створюються в проекті. Якщо ім'я чи реєстри змінних збігаються, але інші властивості відрізняються, створюється нова змінна, але виникає конфлікт реєстрів чи імен. Конфлікт можна усунути у таблиці змінних, задавши інше ім'я чи реєстри.

6 Бібліотека



УВАГА

Для різних моделей пристроїв список доступних елементів бібліотеки компонентів може візнитися.

У відкритому проєкті на панелі [Бібліотека компонентів 2.3](#) є розділи з компонентами для [розробки програми 3.3](#).

Розділи бібліотеки компонентів:

- [Функції 6.1](#) слугують для виконання логічних, арифметичних операцій тощо. Функції не зберігають стану своїх внутрішніх змінних. Число функцій, що використовуються, обмежено тільки обсягом пам'яті пристрою;
- [Функціональні блоки \(ФБ\) 6.2](#) слугують для виконання операцій, що потребують збереження стану своїх внутрішніх змінних. До ФБ належать тригери, лічильники, таймери тощо. Кожна модель пристрою має власне обмеження на кількість ФБ, що використовуються (див. [настанову щодо експлуатування пристрою](#));
- розділ [Макроси проєкту 6.3](#) містить створені користувачем або завантажені за допомогою [менеджера компонентів 3.10](#) макроси;
- [Елементи керування 6.5](#) слугують для [програмування дисплея пристрою 3.4](#) і відображаються на панелі під час роботи з пристроєм з монохромним текстовим РКІ .

6.1 Функції

- [Логічні функції 6.1.1](#);
- [Арифметичні функції 6.1.2](#);
- [Функції порівняння 6.1.3](#);
- [Зсувні функції 6.1.4](#);
- [Бітові функції 6.1.5](#).

6.1.1 Логічні функції

- [І \(AND\) 6.1.1.1](#)
- [АБО \(OR\) 6.1.1.2](#)
- [НЕ \(NOT\) 6.1.1.3](#)
- [Виключне АБО \(XOR\) 6.1.1.4](#)

Особливістю роботи блоків логічних функцій є самоналаштування на тип даних. Якщо до входу блоку була приєднана цілочисельна змінна, то блок автоматично перебудовується на роботу з цілочисельними значеннями.

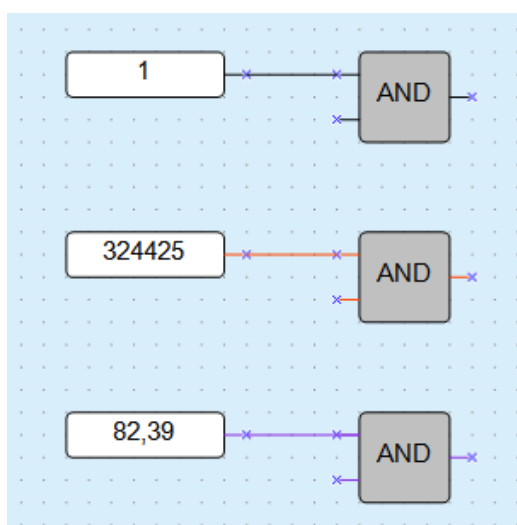


Рисунок 6.1

Для функцій **I** та **АБО** слід враховувати, що непідключені входи блоків будуть мати такі стани:

- для функції **I** – логічна «1»;
- для функції **АБО** – логічний «0».

У цьому випадку блоки виконують функцію повторювача сигналу. Для збільшення числа входів у логічних функціях використовується їхнє каскадне включення:

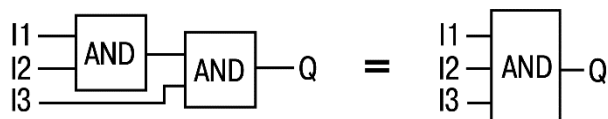


Рисунок 6.2

6.1.1.1 I (AND)

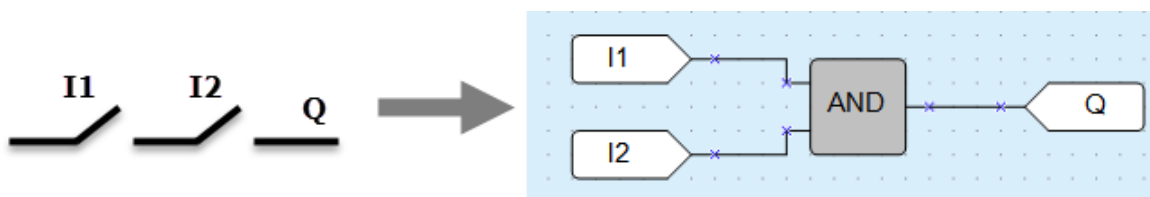


Рисунок 6.3

Якщо на обидва входи блоку функції надходять сигнали логічної «1» (всі входи включені – контакти замкнуті), то на виході з'являється сигнал логічної «1» (вихід включений).

Роботі функції відповідає таблиця станів:

I1	I2	Q
0	0	0
0	1	0
1	0	0
1	1	1

Якщо на входи блоку функції надходять цілочисельні значення, то операція буде проведена над кожним бітом значення окремо.

ТА	0011
	0101
	0001

6.1.1.2 АБО (OR)

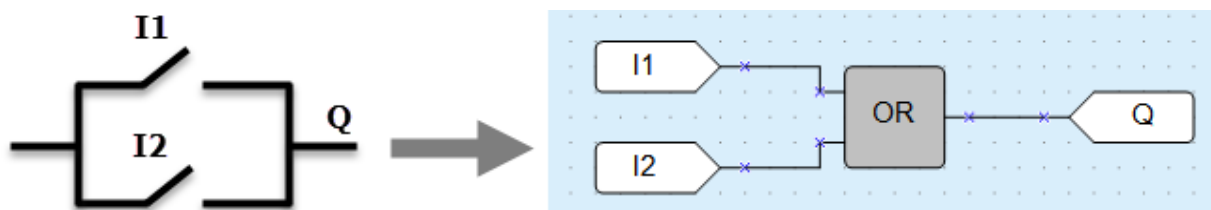


Рисунок 6.4

Якщо на один із входів блоку функції надходить сигнал логічної «1» (контакти замкнуті), то на виході елемента з'являється логічна «1» (вихід включений).

Роботі функції відповідає таблиця станів:

I1	I2	Q
0	0	0
0	1	1

I1	I2	Q
1	0	1
1	1	1

Якщо на входи блоку функції надходять цілочисельні значення, то операція буде проведена над кожним бітом значення окремо.

АБО	0011
	0101
	0111

6.1.1.3 НЕ (NOT)

Функція використовується для інвертування сигналу.

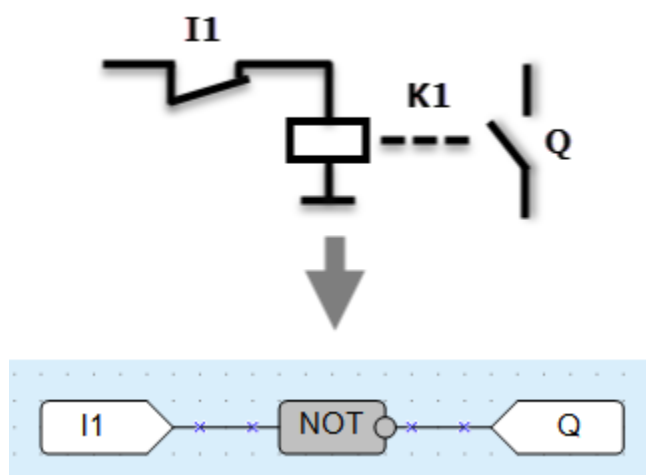


Рисунок 6.5

Якщо на вхід блоку функції надходить сигнал логічного «0» (контакти розімкнуті), то на виході елемента з'являється логічна «1» (вихід включено), і навпаки: сигнал інвертується.

Роботі функції відповідає таблиця станів:

I1	Q
0	1
1	0

Якщо на вхід блоку функції надходить цілочисельне значення, то операція буде проведена над кожним бітом значення окремо.

НЕ	01
	10

6.1.1.4 Виключне АБО (XOR)

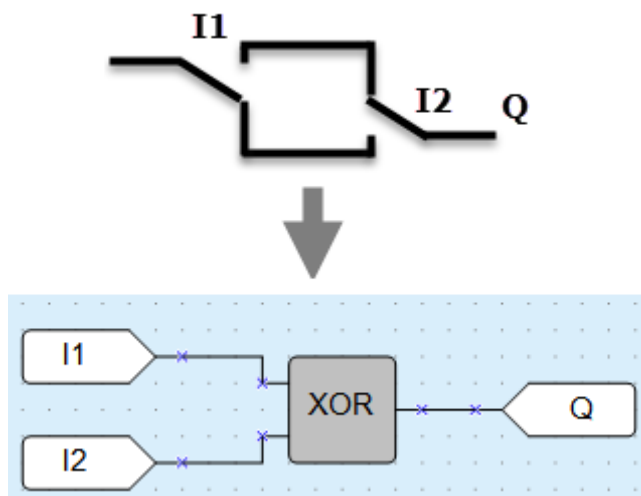


Рисунок 6.6

Якщо тільки на один із входів блоку функції надходить логічна «1», то на виході елемента з'являється логічна «1» (вихід включено).

Роботі функції відповідає таблиця станів:

I1	I2	Q
0	0	0
0	1	1
1	0	1
1	1	0

Якщо на входи функції надходять цілочисельні значення, то операція буде проведена над кожним бітом значення окремо.

XOR	0011
	0101
	0110

6.1.2 Арифметичні функції

**УВАГА**

Арифметичні функції рухомої коми не доступні для пристроїв лінійки ПР110.

Операції **арифметичні функції** можуть проводитися лише над значеннями цілочисельного типу і типу з рухомою комою.

Назва	Цілочисельний тип	Тип з рухомою комою
Додавання	ADD 6.1.2.1	fADD 6.1.2.1
Віднімання	SUB 6.1.2.2	fSUB 6.1.2.2
Множення	MUL 6.1.2.3	fMUL 6.1.2.3
Ділення	DIV 6.1.2.4	fDIV 6.1.2.4
Ділення із залишком	MOD 6.1.2.5	–

Назва	Цілочисельний тип	Тип з рухомою комою
Зведення до ступеня	–	fPOW 6.1.2.6
Модуль від числа	–	fABS 6.1.2.7

6.1.2.1 Додавання (ADD, fADD)

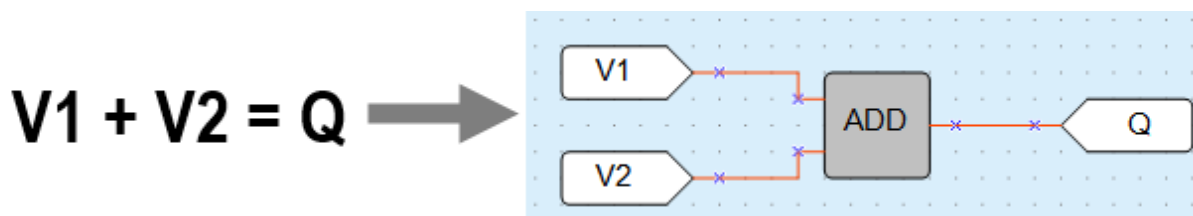


Рисунок 6.7

Для операцій над цілочисельними значеннями використовується блок **ADD**, над значеннями з рухомою комою – **fADD**.

Результатом операції функції на виході є сума вхідних значень.

Якщо під час виконання операції значення числа виходить більше 4294967295 (32 біти), то біти, що виходять за розрядність 32 біти, відсікаються.

Приклад

Застосування блоків функції для додавання чисел $24 + 2 + 2 + 7 = Q = 35$.

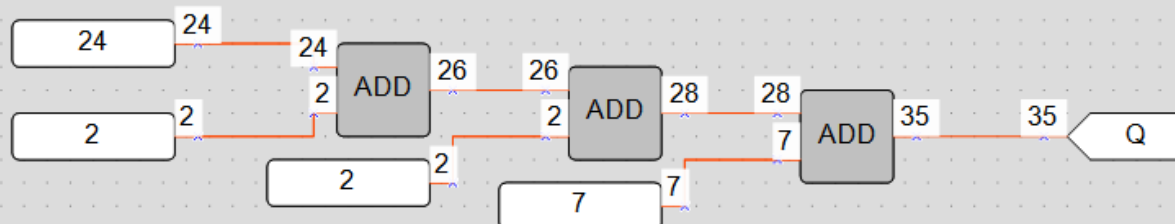


Рисунок 6.8

6.1.2.2 Віднімання (SUB, fSUB)

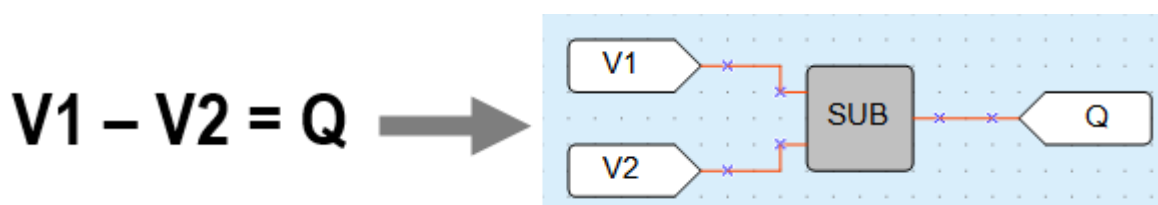


Рисунок 6.9

Для операцій над цілочисельними значеннями використовується блок **SUB**, над значеннями з рухомою комою – **fSUB**.

Результатом операції функції на виході є різниця вхідних значень.

Якщо під час виконання операції функції на вході **V1** значення числа менше значення на вході **V2**, то результатом буде число, отримане додаванням молодшого числа плюс $0x100000000$ (4294967296) мінус значення числа більшого: $(V1 + 0x100000000) - V2 = Q$.

Приклад

Застосування блоку функції для віднімання чисел $24 - 2 = 22$:

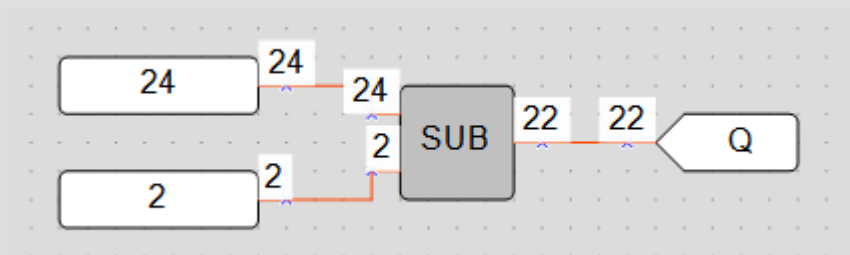


Рисунок 6.10

Застосування блоку функції для віднімання чисел, де $V1$ менше $V2$: $2 - 24 = Q = 4294967274$.

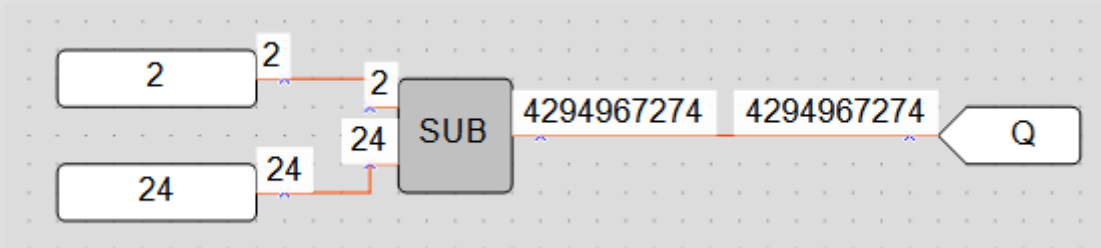


Рисунок 6.11

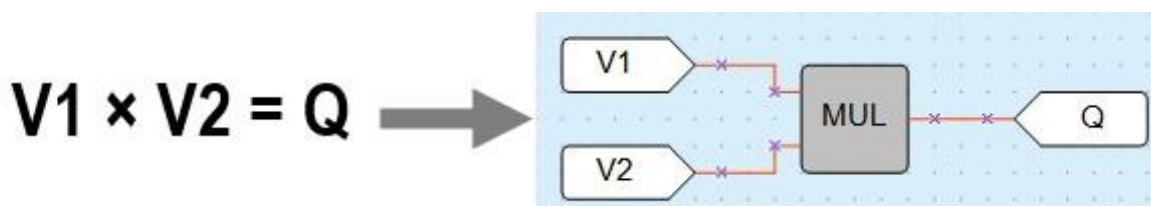
6.1.2.3 Множення (MUL, fMUL)

Рисунок 6.12

Для операцій над цілочисельними значеннями використовується блок **MUL**, над значеннями з рухомою комою – **fMUL**.

Результатом операції функції на виході є добуток вхідних значень.

Якщо під час виконання операції функції значення числа виходить більше 4294967295 (32 біти), то біти, що виходять за розрядність 32 біти, відсікаються.

Приклад

Застосування блоків функції для перемноження чисел $24 \cdot 2 \cdot 2 \cdot 7 = Q = 672$.

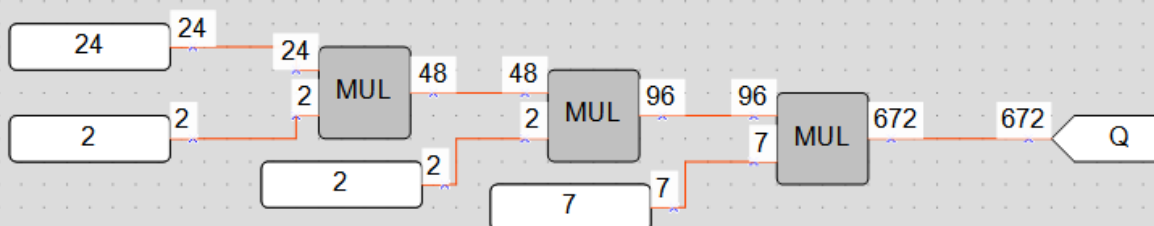


Рисунок 6.13

6.1.2.4 Ділення (DIV, fDIV)

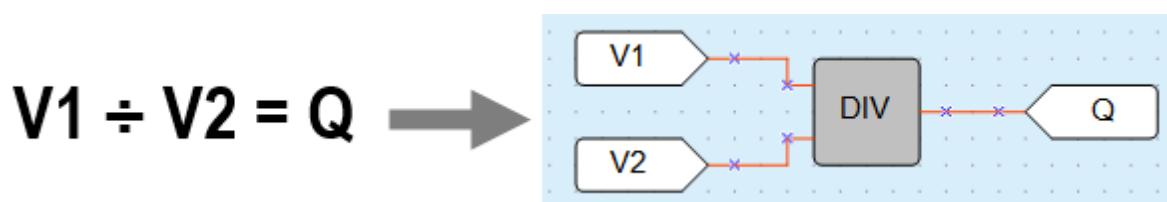


Рисунок 6.14

Для операцій над цілочисельними значеннями використовується блок **DIV**, над значеннями з рухомою комою – **fDIV**.

Результатом операції функції на виході є частка від ділення вхідних значень.

Якщо під час використання блоку DIV в результаті ділення виходять частки цілого числа, то на виході проводиться округлення значення до цілого числа в менший бік.

У разі ділення на 0 на виході елемента буде значення 0.

Приклад

Застосування блоку функції для ділення чисел $24 \div 2 = Q = 12$

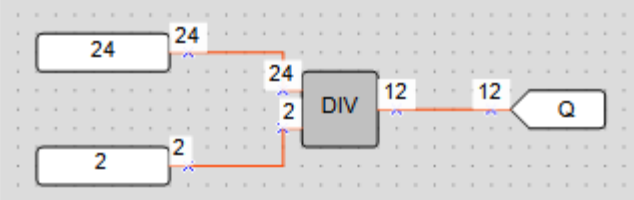


Рисунок 6.15

6.1.2.5 Ділення із залишком (MOD)

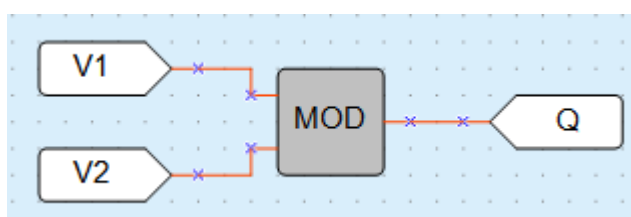


Рисунок 6.16

Результатом операції функції на виході є залишок від ділення вхідних цілочислових значень.

У разі ділення на 0 на виході блоку функції буде значення 0.

Приклад

Застосування блоку функції для виділення цілого залишку від ділення числа 22 на 3.

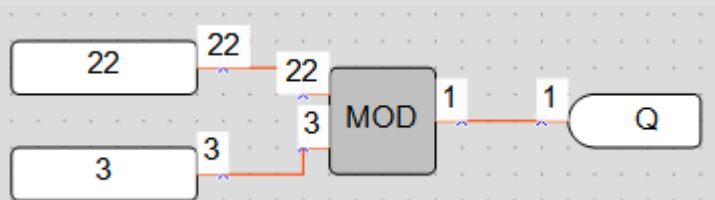


Рисунок 6.17

6.1.2.6 Зведення числа до ступеня (fPOW)

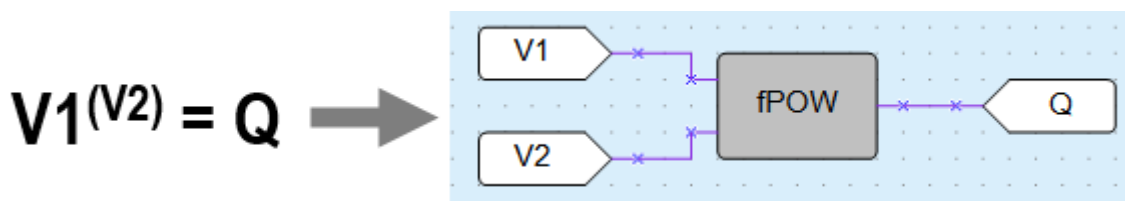


Рисунок 6.18

Результатом операції функції на виході є змінна типу float, що дорівнює числу V1, зведеному до ступеня V2.

Блок працює лише зі значеннями з рухомою комою.

Приклад

Застосування блоку функції для зведення числа 3 у ступінь числа 4.

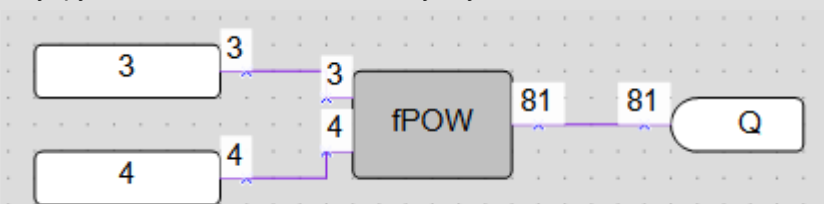


Рисунок 6.19

6.1.2.7 Взяття модуля від числа (fABS)

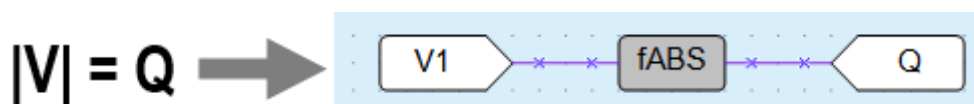


Рисунок 6.20

Блок працює лише зі значеннями з рухомою комою.

Результатом операції функції на виході є модуль значення, що подається на вхід.

Приклад

Застосування блоків функції визначення модулів чисел: $|-3,4| = Q = |-3,4|$, $|6,7| = Q = 6,7$

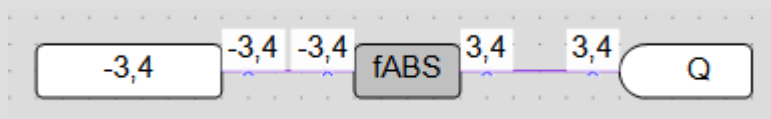


Рисунок 6.21

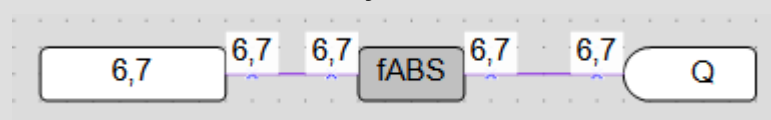


Рисунок 6.22

6.1.3 Функції порівняння

- Дорівнює (EQ) 6.1.3.1;
- Більше (GT, fGT) 6.1.3.2;
- Вибір (SEL) 6.1.3.3.

6.1.3.1 Дорівнює (EQ)

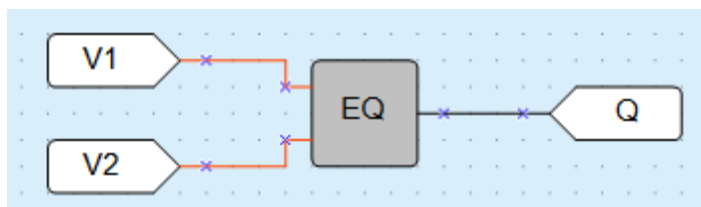


Рисунок 6.23

Результати виконання операції блоку на виході порівняння вхідних значень:

- $V1 = V2 \rightarrow Q = 1$;
- $V1 > V2 \rightarrow Q = 0$;
- $V1 < V2 \rightarrow Q = 0$.

Приклад

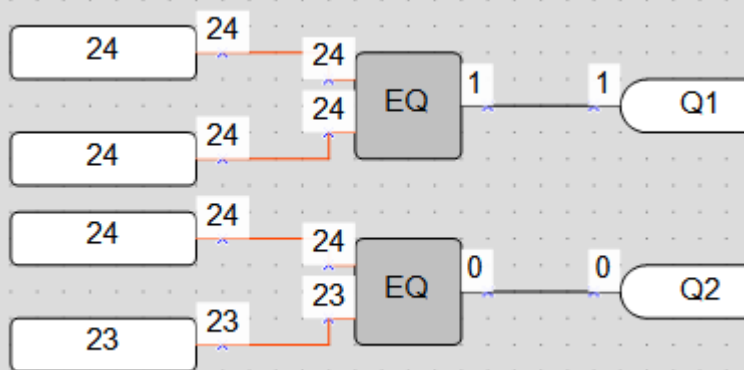


Рисунок 6.24

6.1.3.2 Більше (GT, fGT)

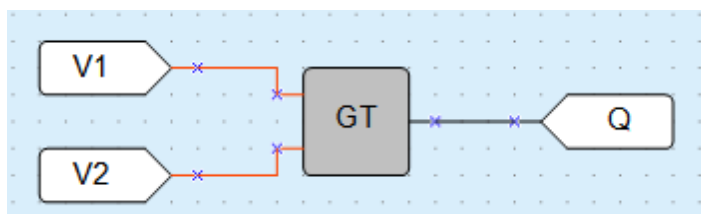


Рисунок 6.25

Результати виконання операції блоку на виході при порівнянні вхідних значень:

- $V1 = V2 \rightarrow Q = 0$;
- $V1 > V2 \rightarrow Q = 1$;
- $V1 < V2 \rightarrow Q = 0$.

Для порівняння цілочисельних значень використовується блок **GT**, для значень з рухомою комою – блок **fGT**.

Приклад

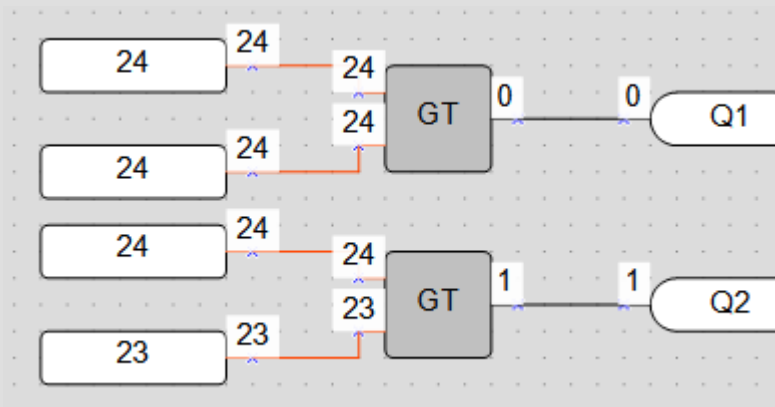


Рисунок 6.26

6.1.3.3 Вибір (SEL, fSEL)



ПРИМІТКА

Функція Вибір (SEL, fSEL) в AQLogic називається Тернарна умовна операція порівняння.

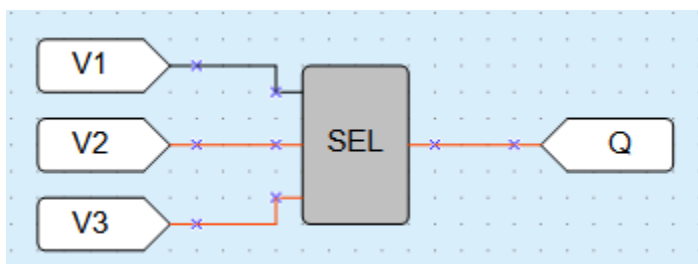


Рисунок 6.27

Якщо **V1** дорівнює логічному «0», то результатом виконання операції блоку на виході є вхідний сигнал **V2**. Якщо **V1** дорівнює логічній «1», результатом виконання операції блоку на виході є вхідний сигнал **V3**.

- $V1 = 0 \rightarrow Q = V2$;
- $V1 = 1 \rightarrow Q = V3$.

Для вибору цілочисельних значень використовується блок **SEL**, для значень з рухомою комою – блок **fSEL**.

Приклад

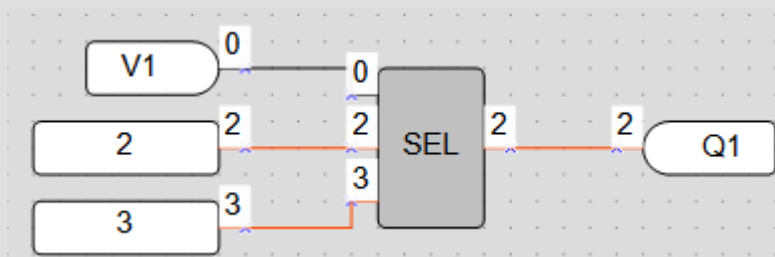


Рисунок 6.28

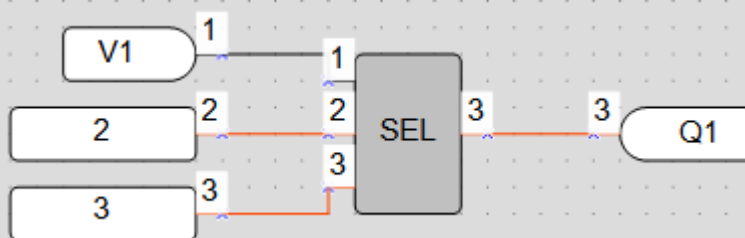


Рисунок 6.29

6.1.4 Зсувні функції

- Логічний зсув вліво (SHL) 6.1.4.1;
- Логічний зсув вправо (SHR) 6.1.4.2.

6.1.4.1 Побітовий логічний зсув вліво (SHL)

Побітовий логічний зсув вліво (SHL) використовується для виконання операції побітового логічного зсуву операнда X вліво на N біт з доповнення нулями праворуч.

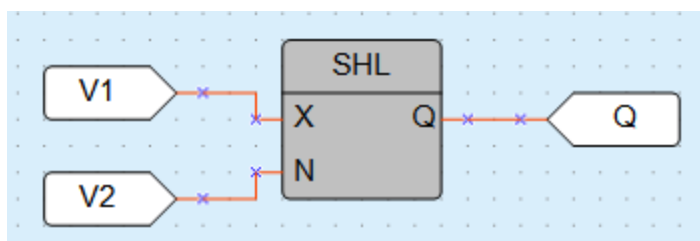


Рисунок 6.30

Приклад

Застосування для зсуву числа 38 (десятькове) = 00100110 (двійкове).

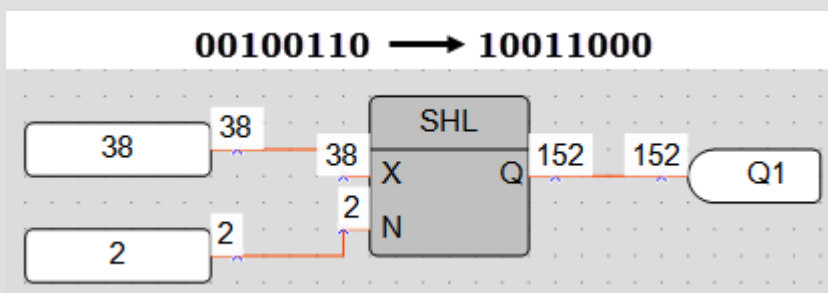


Рисунок 6.31

6.1.4.2 Побітовий логічний зсув вправо (SHR)

Побітовий логічний зсув праворуч (SHR) використовується для виконання операції побітового логічного зсуву операнда **X** вправо на **N** біт з доповненням нулями зліва.

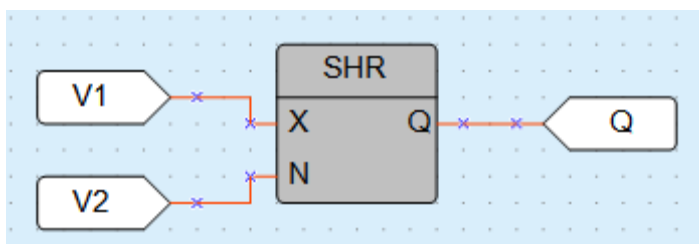


Рисунок 6.32

Приклад

Застосування для зсуву числа 152 (десятькове) = 10011000 (двійкове).

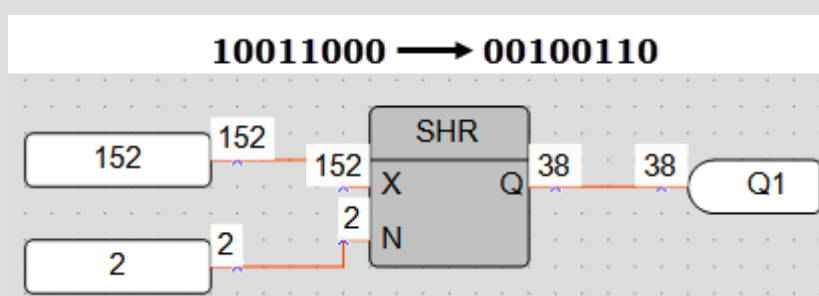


Рисунок 6.33

6.1.5 Бітові функції

- Читання біта (EXTRACT) 6.1.5.1;
- Запис біта (PUTBIT) 6.1.5.2;
- Дешифратор (DC32) 6.1.5.3;
- Шифратор (CD32) 6.1.5.4.

6.1.5.1 Читання біта (EXTRACT)

Читання біта (EXTRACT) використовується для виконання операції читання значення біта **N** у числі входу **X**. Біти нумеруються з кінця. Число на вході **X** визначається в десятковій системі. Вихідне значення **Q** завжди двійкове (0 або 1).

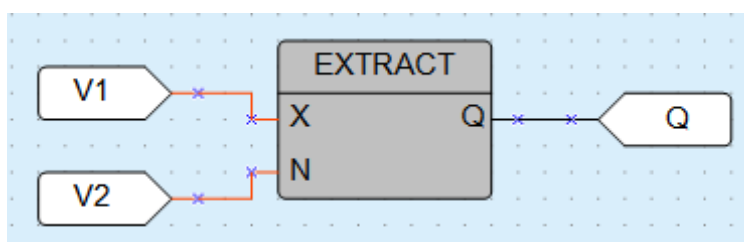


Рисунок 6.34

Приклад

Застосування блоку для читання п'ятого біта у складі 81 (десятькове) = 1010001 (двійкове).

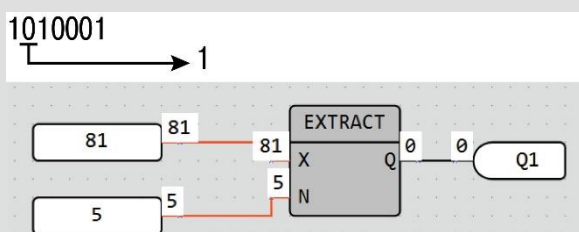


Рисунок 6.35

6.1.5.2 Запис біта (PUTBIT)

Запис біта (PUTBIT) використовується для виконання операції запису в числі X значення біта N у стан, зазначений на вході B (логічні «0» або «1»). Число на вході X визначається в десятковій системі. Вихідне значення Q – цілочисельне.

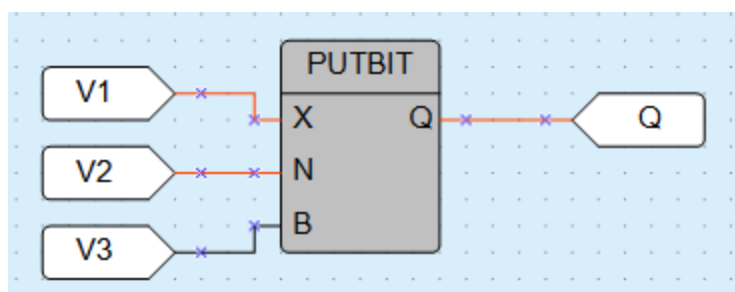


Рисунок 6.36

Приклад

Застосування запису четвертого біта у сигналі логічної «1» для числа 38 (десятькове) = 100110 (двійкове).

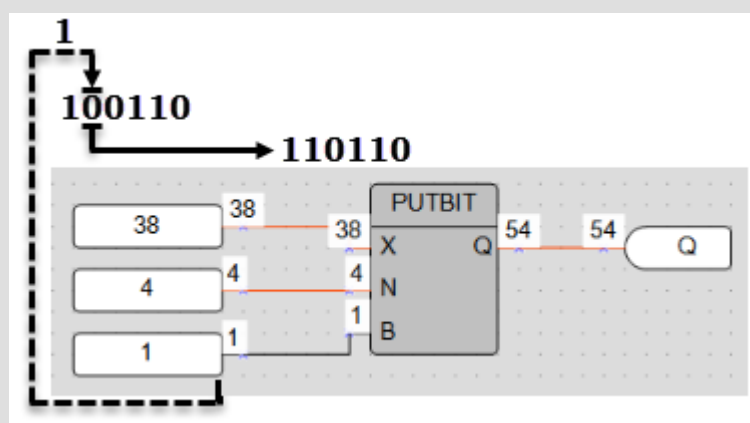


Рисунок 6.37

6.1.5.3 Дешифратор (DC32)

Дешифратор (DC32) використовується для виконання операції перетворення двійкового коду на вході в позиційний код на виході. Перед виконанням операції над значенням на вході попередньо виконується побітова логічна операція «І» з операндом 0x1F (11111b).



Рисунок 6.38

Роботі дешифратора відповідає наведена таблиця станів:

Двоичный код							Позиционный код					
5	4	3	2	1	32	31						
							6	5	4	3	2	1
0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	1	1	0	0	0	0	1	0	0	0
		1	0	0	0	0		1				
...							...					
1	1	1	0	1	0	0	0	0	0	0	0	0
1	1	1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0

Рисунок 6.39

Приклад

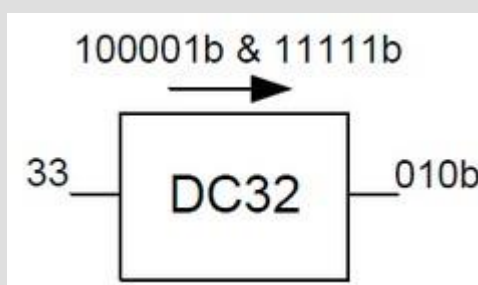


Рисунок 6.40

6.1.5.4 Шифратор (CD32)

Шифратор (CD32) використовується для виконання операції перетворення позиційного коду на вході в двійковий код на виході.

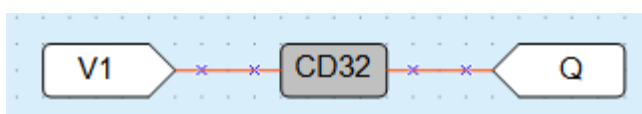


Рисунок 6.41

Якщо вхідне двійкове значення має більше однієї логічної «1» у розрядах, то робота шифратора ведеться лише зі старшим одиничним розрядом.

Роботі шифратора відповідає таблиця станів.

Двоичный код					Позиционный код								
5	4	3	2	1	32	31	...	6	5	4	3	2	1
0	0	0	0	0	0	0		0	0	0	0	0	1
0	0	0	0	1	0	0		0	0	0	0	1	0
0	0	0	1	0	0	0		0	0	0	1	0	0
0	0	0	1	1	0	0		0	0	1	0	0	0
		1	0	0	0	0	...		1				
...									
1	1	1	0	1	0	0		0	0	0	0	0	0
1	1	1	1	0	0	1		0	0	0	0	0	0
1	1	1	1	1	1	0		0	0	0	0	0	0

Рисунок 6.42

6.2 Функціональні блоки

- Тригери 6.2.1;
- Таймери 6.2.2;
- Генератори 6.2.3;
- Лічильники 6.2.4;
- Регулятори 6.2.5.

6.2.1 Тригери

- RS-тригер з пріоритетом відключення (RS) 6.2.1.1;
- SR-тригер з пріоритетом включення (SR) 6.2.1.2;
- Детектор переднього фронту імпульсу (RTRIG) 6.2.1.3;
- Детектор заднього фронту імпульсу (FTRIG) 6.2.1.4;
- D-тригер (DTRIG) 6.2.1.5.

6.2.1.1 RS-тригер з пріоритетом відключення

RS-тригер із пріоритетом відключення використовується для переключення з фіксацією стану під час надходження коротких імпульсів на відповідний вхід. На виході **Q** з'явиться логічна «1» по фронту сигналу на вході **S**.

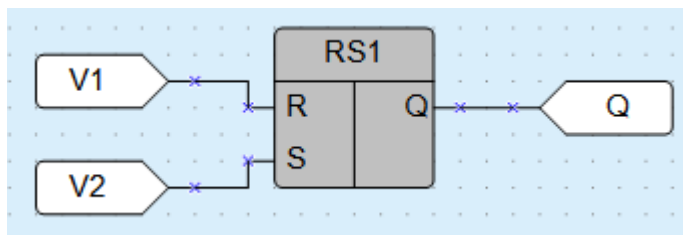


Рисунок 6.43

Роботу тригера пояснює наведена на рисунку нижче діаграма.

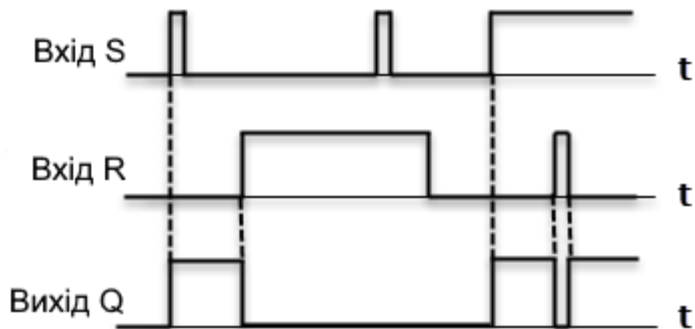


Рисунок 6.44

У разі одночасного надходження сигналів на обидва входи пріоритетним є сигнал входу R.

6.2.1.2 SR-тригер з пріоритетом включення

SR-тригер із пріоритетом включення використовується для переключення з фіксацією стану у разі надходження коротких імпульсів на відповідний вхід. На виході Q з'явиться логічна «1» по фронту сигналу на вході S.

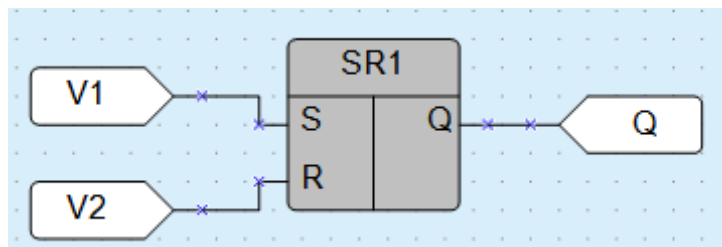


Рисунок 6.45

Роботу тригера пояснює наведена на рисунку нижче діаграма.

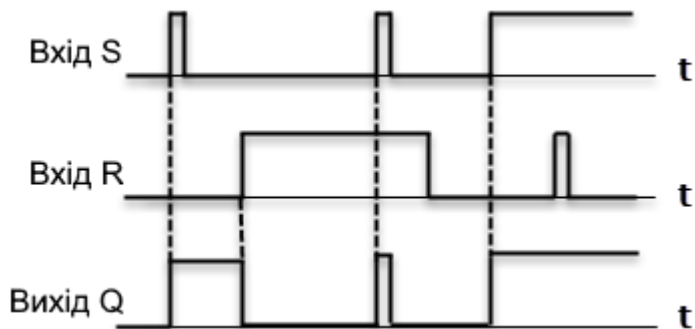


Рисунок 6.46

У разі одночасного надходження сигналів на обидва входи пріоритетним є сигнал входу S.

6.2.1.3 Детектор переднього фронту імпульсу (RTRIG)

Детектор переднього фронту імпульсу (RTRIG) використовується у разі потреби мати реакцію на зміну стану дискретного вхідного сигналу. На виході **Q** генерується одиничний імпульс переднього фронту входу **I**.

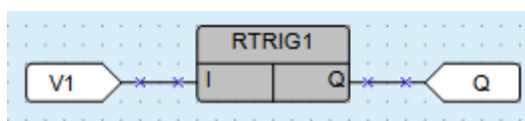


Рисунок 6.47

Роботу детектора пояснює наведена на рисунку нижче діаграма.

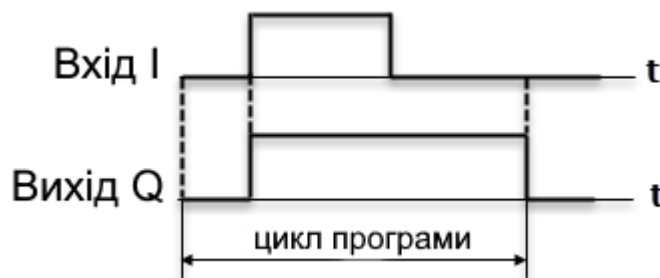


Рисунок 6.48

6.2.1.4 Детектор заднього фронту імпульсу (FTRIG)

Детектор заднього фронту імпульсу (FTRIG) використовується у разі потреби мати реакцію на зміну стану дискретного вхідного сигналу. На виході **Q** генерується одиничний імпульс заднього фронту входу **I**.

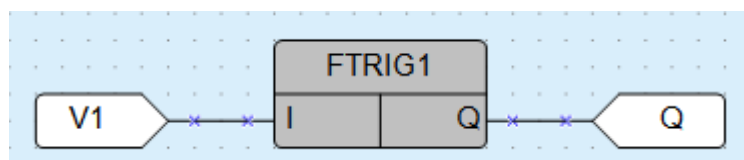


Рисунок 6.49

Роботу детектора пояснює наведена на рисунку нижче діаграма.

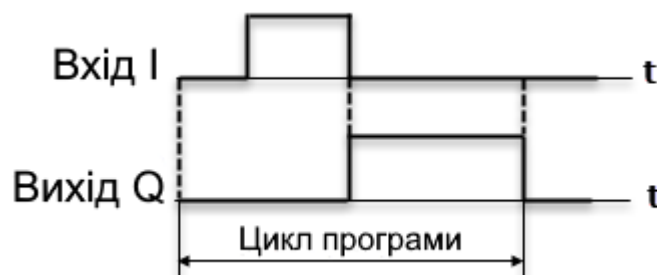


Рисунок 6.50

6.2.1.5 D-тригер (DTRIG)

D-тригер (DTRIG) використовується для формування імпульсу включення виходу на інтервал часу імпульсу на вході **D**, вихідний інтервал буде синхронізовано з тактовою частотою на вході **C**.

На виході **Q** тригера з'явиться сигнал логічної «1» по фронту тактових імпульсів на вході **C** за наявності сигналу логічної «1» на вході **D**. Повернення виходу **Q** у сигнал логічного «0» відбудеться по фронту тактових імпульсів на вході **C** за наявності сигналу логічного «0» на вході **D**.

Вхід **S** примусово встановлює вихід **Q** у стан логічної «1».

Вхід **R** є пріоритетним і встановлює вихід **Q** у стан логічного «0».

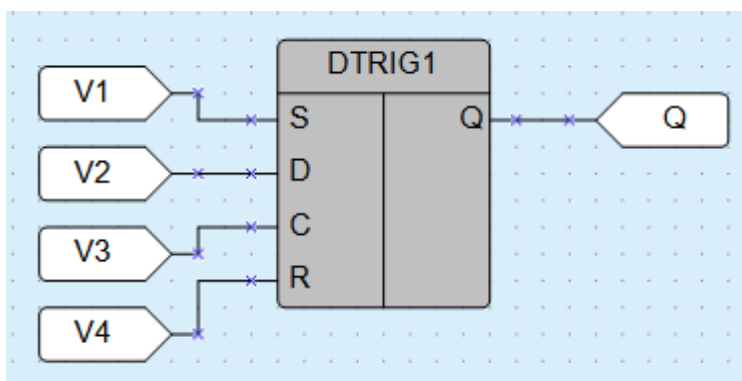


Рисунок 6.51

Роботу тригера пояснює наведена на рисунку нижче діаграма.

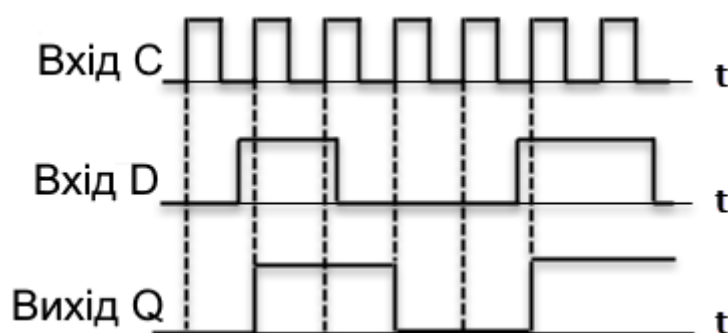


Рисунок 6.52

6.2.2 Таймери

- Імпульс включення заданої тривалості (TP) 6.2.2.1;
- Таймер із затримкою включення (TON) 6.2.2.2;
- Таймер із затримкою відключення (TOF) 6.2.2.3;
- Інтервальний таймер (CLOCK) 6.2.2.4;
- Інтервальний таймер із тижневим циклом (CLOCKWEEK) 6.2.2.5.

6.2.2.1 Імпульс включення заданої тривалості (TP)

Імпульс включення заданої тривалості (TP) використовується для формування імпульсу включення виходу на заданий інтервал часу. На виході **Q** блоку з'являється сигнал логічної «1» по фронту вхідного сигналу **I**. Після запуску вихід **Q** не реагує на зміну значення вхідного сигналу протягом інтервалу $T_{\text{імп}}$. Після закінчення інтервалу $T_{\text{імп}}$ вихідний сигнал скидається в логічний «0».

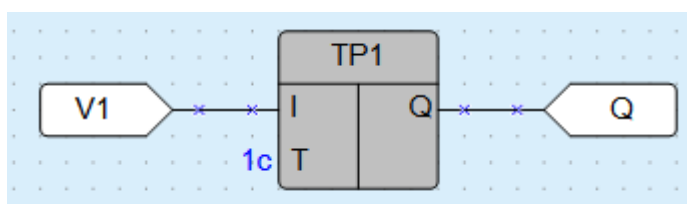


Рисунок 6.53

Роботу імпульсу пояснює наведена на рисунку нижче діаграма.

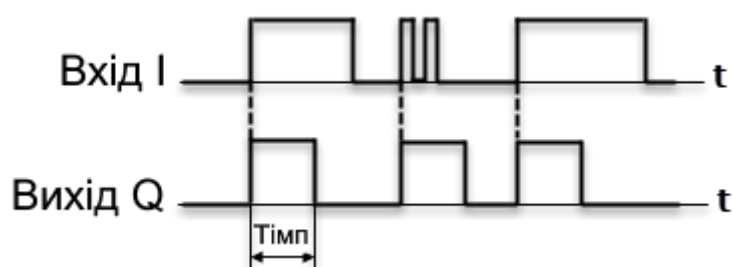


Рисунок 6.54

Допустимий діапазон значень $T_{\text{імп}}$ від 0 до 4147200000 мс або 48 днів.

Тривалість імпульсу та одиниці вимірювання часу налаштовуються на панелі властивостей ФБ.

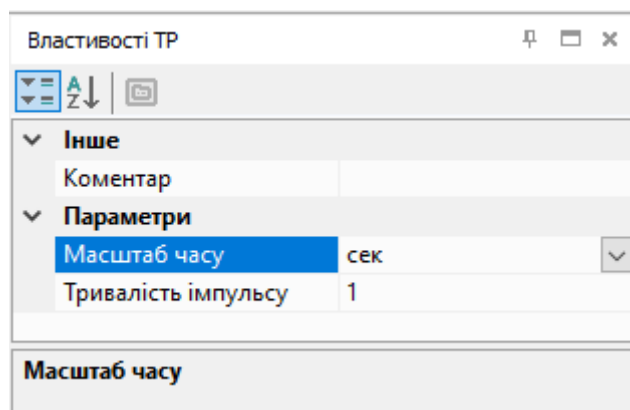


Рисунок 6.55

6.2.2.2 Таймер із затримкою включення (TON)

Таймер із затримкою включення (TON) використовується для операції затримки передання сигналу. На виході **Q** таймера з'явиться сигнал логічної «1» із затримкою щодо фронту вхідного сигналу **I** тривалістю не менше тривалості **Ton** і відключиться по спаду вхідного сигналу.

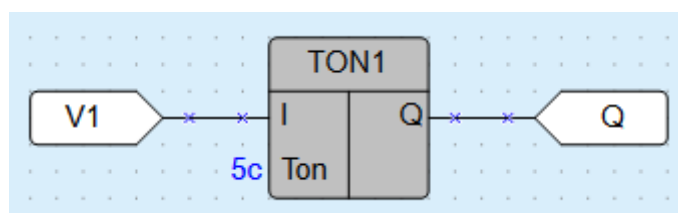


Рисунок 6.56

Роботу таймера пояснює наведено на рисунку нижче діаграма.

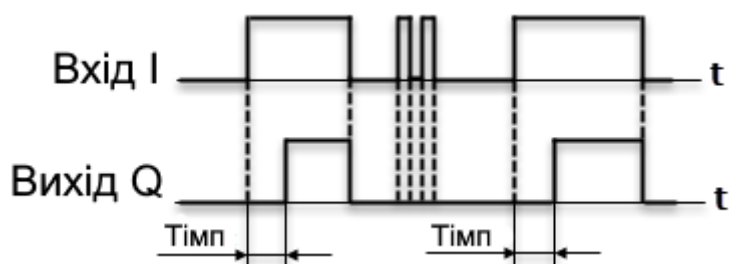


Рисунок 6.57

Допустимий діапазон значень **T** від 0 до 4147200000 мс або 48 днів.

Час затримки включення та одиниці виміру часу налаштовуються на панелі властивостей ФБ.

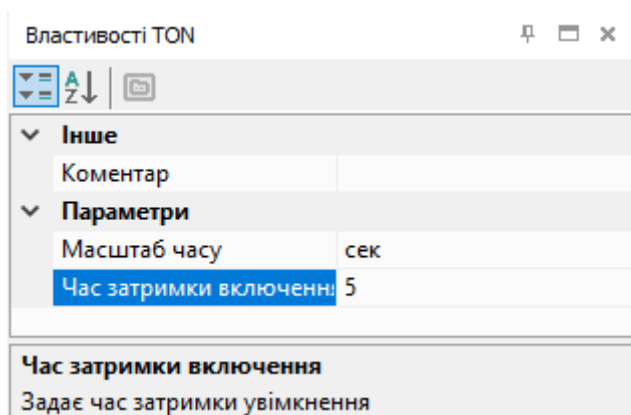


Рисунок 6.58

6.2.2.3 Таймер із затримкою відключення (TOF)

Таймер із затримкою відключення (TOF) використовується для затримки відключення виходу. На виході **Q** таймера з'явиться сигнал логічної «1» по фронту сигналу на вході **I**, відлік часу затримки відключення **Toff** почнеться по кожному спаду вхідного сигналу. Після відключення вхідного сигналу на виході з'явиться сигнал логічного «0» із затримкою **Toff**.

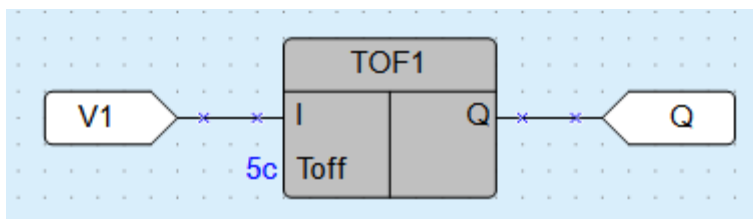


Рисунок 6.59

Роботу таймера пояснює наведена на рисунку нижче діаграма.

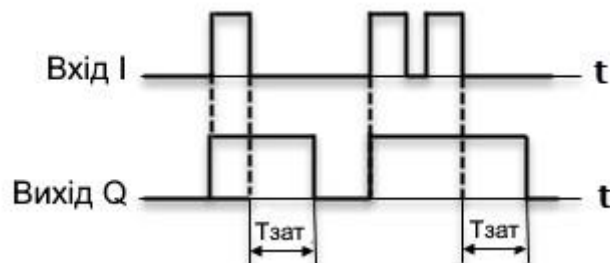


Рисунок 6.60

Допустимий діапазон значень $T_{\text{зат}}$ від 0 до 4147200000 мс або 48 днів.

Час затримки відключення та одиниці виміру часу налаштовуються на панелі властивостей ФБ.

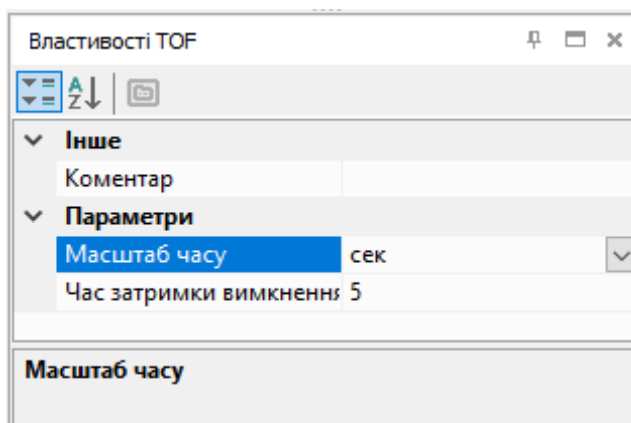


Рисунок 6.61

6.2.2.4 Інтервальний таймер (CLOCK)



ПРИМІТКА

Інтервальний таймер (CLOCK) доступний лише для пристроїв із годинником реального часу.

Інтервальний таймер (CLOCK) використовується для формування імпульсу включення виходу **Q** по годиннику реального часу. Час включення **Th** і відключення **Tl** виходу встановлюються як параметри таймера.

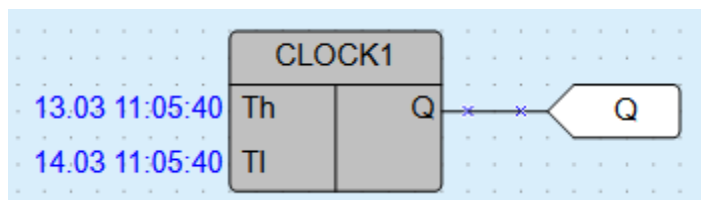


Рисунок 6.62

Роботу таймера пояснює наведена на рисунку нижче діаграма.

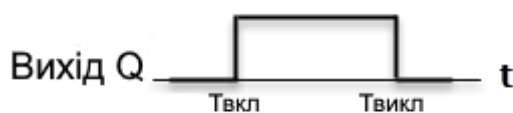


Рисунок 6.63

Якщо задане значення часу відключення менше часу включення, то діаграма переключень матиме вигляд:



Рисунок 6.64

Допустимий діапазон значень $T_{\text{вкл}}$ та $T_{\text{викл}}$ від 0,00 с до 24 год.

Дата та час включення та відключення налаштовуються на панелі властивостей ФБ.

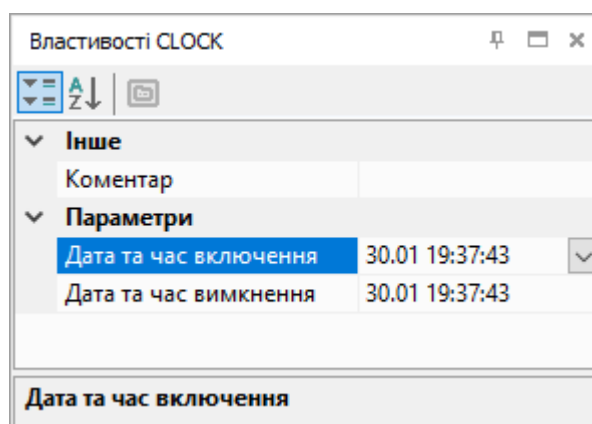


Рисунок 6.65

6.2.2.5 Інтервальний таймер із тижневим циклом (CLOCKWEEK)



ПРИМІТКА

Інтервальний таймер із тижневим циклом (CLOCKWEEK) доступний лише для пристроїв з годинником реального часу.

Інтервальний таймер із тижневим циклом (CLOCKWEEK) використовується для формування імпульсу включення виходу **Q** по годиннику реального часу з урахуванням днів тижня. Час включення **Th** і відключення **Tl** виходу **Q** і дні тижня роботи встановлюються як параметри таймера.

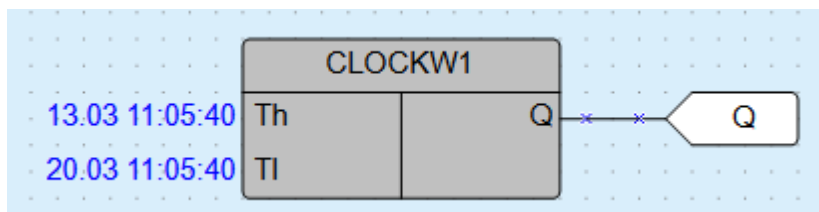


Рисунок 6.66

Роботу таймера пояснює наведена на рисунку нижче діаграма.



Рисунок 6.67

Внутрішня структура таймера має такий вигляд.

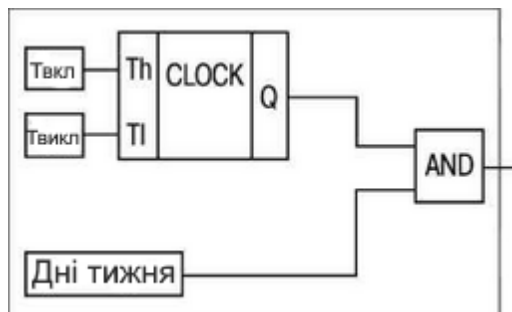


Рисунок 6.68

Допустимий діапазон значень $T_{\text{вкл}}$ і $T_{\text{викл}}$ від 0,00 с до 24 год.

Налаштування часу включення та відключення і зазначення днів тижня проводяться на панелі властивостей ФБ.

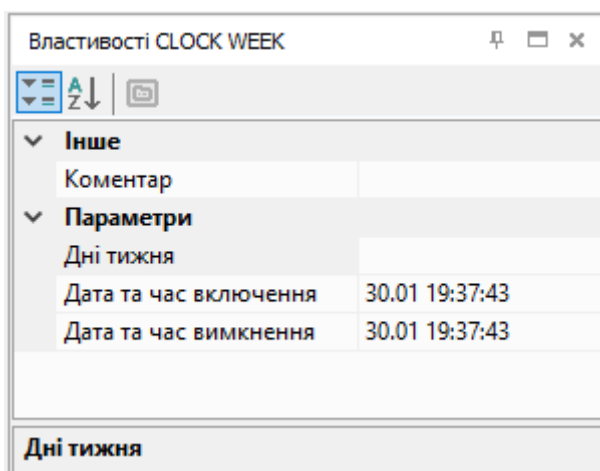


Рисунок 6.69

6.2.3 Генератори

- [Генератор прямокутних імпульсів \(BLINK\) 6.2.3.1.](#)

6.2.3.1 Генератор прямокутних імпульсів (BLINK)

Генератор прямокутних імпульсів (BLINK) використовується для формування прямокутних імпульсів пульсації. На виході **Q** генератора формуються імпульси із заданими параметрами тривалості включеного ($T_{\text{вкл}}$ – сигнал логічної «1») і відключеного ($T_{\text{викл}}$ – сигнал логічного «0») стану на час дії керуючого сигналу на вході **I** (сигнал логічної «1»).

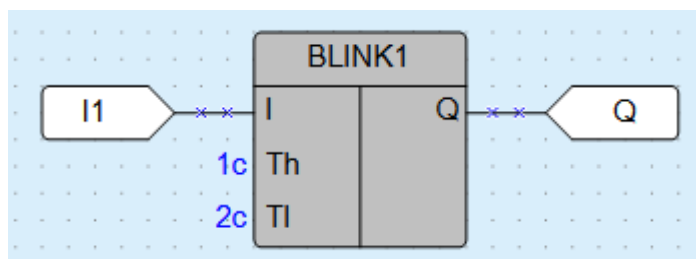


Рисунок 6.70

Роботу генератора пояснює наведена на рисунку нижче діаграма.

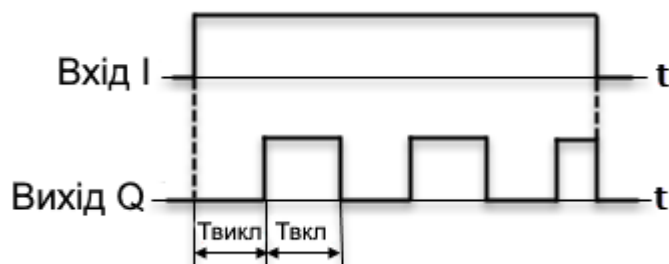


Рисунок 6.71

Допустимий діапазон значень $T_{\text{вкл}}$ і $T_{\text{викл}}$: від 0 до 4233600000 мс або 49 днів.

Тривалість включеного та відключеного станів налаштовується на панелі властивостей ФБ.

Параметр T_h відповідає $T_{\text{вкл}}$.

Параметр T_l відповідає $T_{\text{викл}}$.

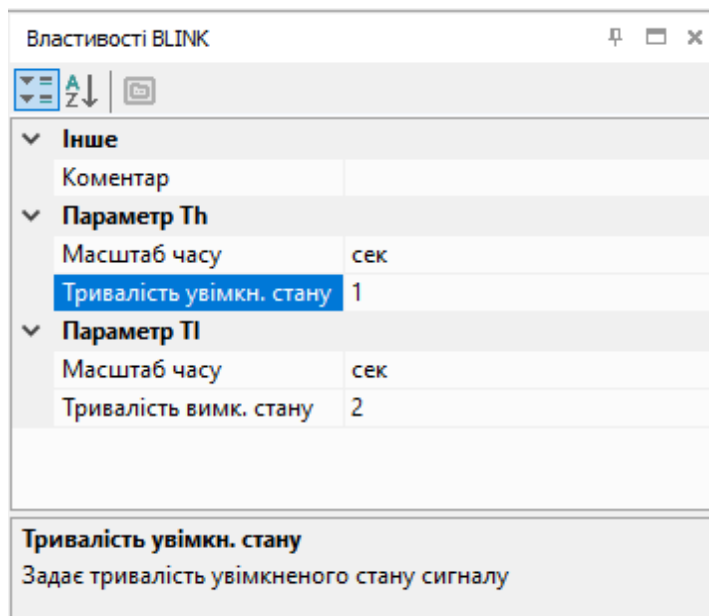


Рисунок 6.72

6.2.4 Лічильники

- Інкрементний лічильник з автоскиданням (СТ) 6.2.4.1;
- Універсальний лічильник (СТN) 6.2.4.2;
- Інкрементний лічильник (СТU) 6.2.4.3.

6.2.4.1 Інкрементний лічильник з автоскиданням (СТ)

Інкрементний лічильник з автоскиданням (СТ) використовується для підрахунку заданого числа імпульсів N (вхід N – уставка числа імпульсів). На виході Q лічильника з'явиться імпульс сигналу логічної «1» з тривалістю робочого циклу пристрою ($T_{\text{цикл}}$), якщо число імпульсів, що приходять на вхід C , досягне встановленого значення N .

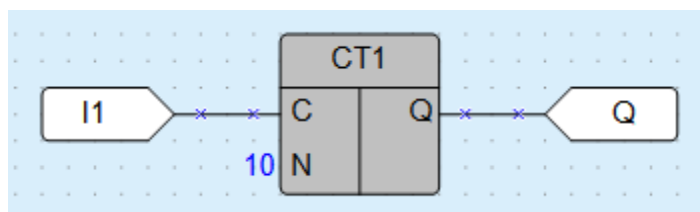


Рисунок 6.73

Роботу лічильника пояснює наведена на рисунку нижче діаграма.

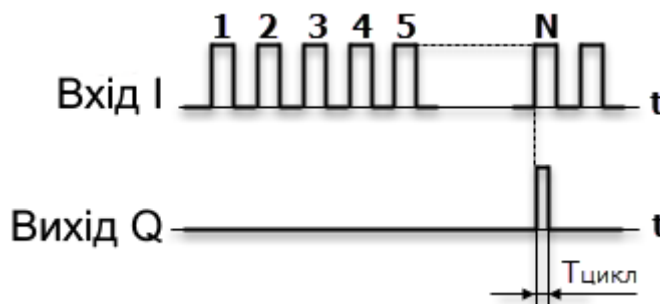


Рисунок 6.74

Допустимий діапазон значень числа імпульсів N : від 0 до 65535.

Число імпульсів N і параметр збереження стану у ПЗП задаються на панелі властивостей ФБ.

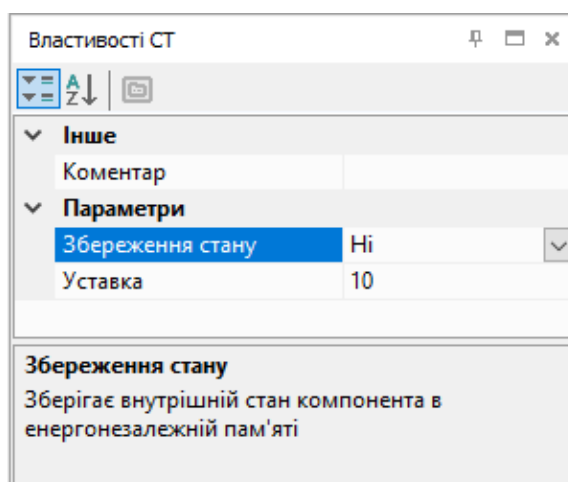


Рисунок 6.75

6.2.4.2 Універсальний лічильник (СТН)

Універсальний лічильник (СТН) використовується для прямого та зворотного рахунку. Операція «прямий рахунок» виконується по передньому фронту імпульсу на вході прямого рахунку **U**, що збільшує значення вихідного сигналу **Q**. Імпульси, що приходять на вхід **D** («зворотний рахунок»), зменшують значення виходу **Q**. У разі надходження на вхід **R** сигналу логічної «1», вихід лічильника **Q** встановлюється значення входу **N**.

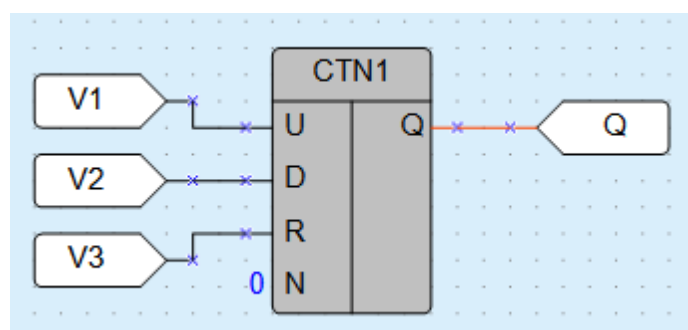


Рисунок 6.76

Роботу лічильника пояснює наведена на рисунку нижче діаграма.

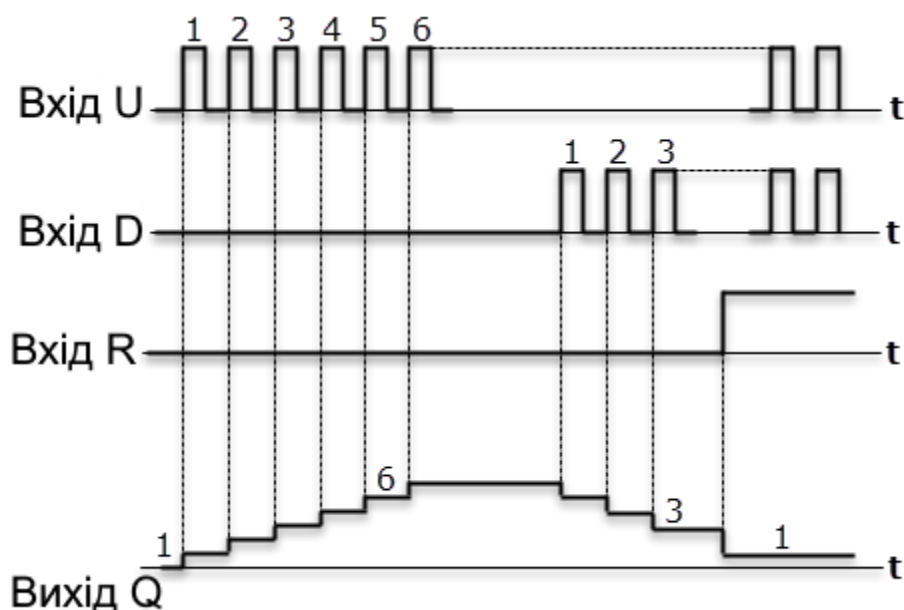


Рисунок 6.77

У разі одночасного надходження сигналів на входи **U** та **D** пріоритетним є сигнал входу **U**.

Допустимий діапазон значень числа імпульсів **N**: від 0 до 65535.

Значення **N** та параметр збереження стану в ПЗП задаються на панелі властивостей ФБ.

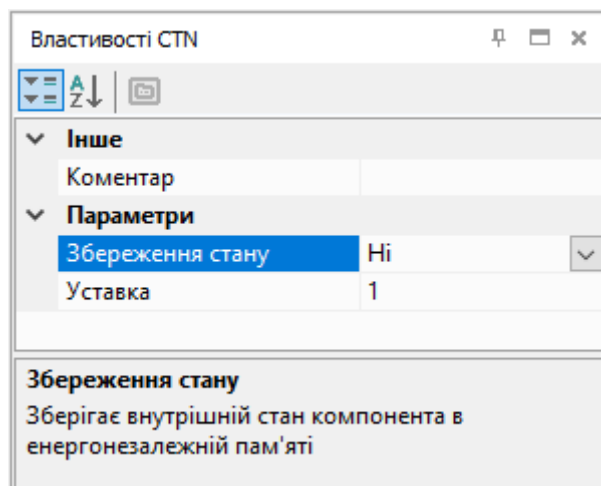


Рисунок 6.78

6.2.4.3 Інкрементний лічильник (СТУ)

Інкрементний лічильник (СТУ) використовується для підрахунку числа імпульсів, що приходять на вхід **C**. На виході **Q** лічильника з'явиться імпульс сигналу логічної «1», якщо число імпульсів, що приходять на вхід, досягне встановленого значення на вході **N** (**N** – уставка).

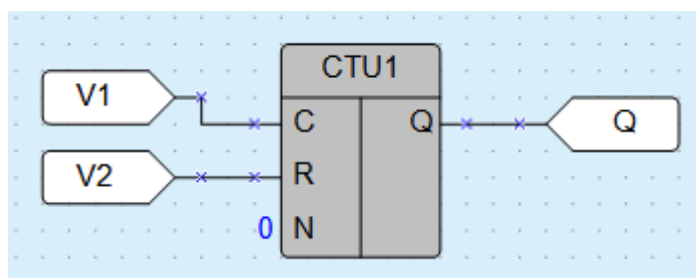


Рисунок 6.79

Роботу лічильника пояснює наведена на рисунку нижче діаграма.

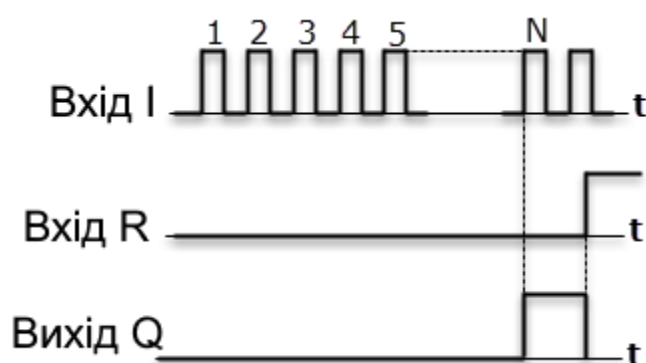


Рисунок 6.80



ПРИМІТКА

Блок **Запис у ФБ** для СТУ не працює. Замість СТУ рекомендується використовувати універсальний лічильник [СТН 6.2.4.2](#).

Допустимий діапазон значень числа імпульсів **N**: від 0 до 65535.

Лічильник скидається в 0 за переднім фронтом імпульсу на вході **R**. У разі одночасного надходження сигналів на входи пріоритетним є сигнал входу **R**.

Число імпульсів **N** задається на панелі властивостей ФБ.

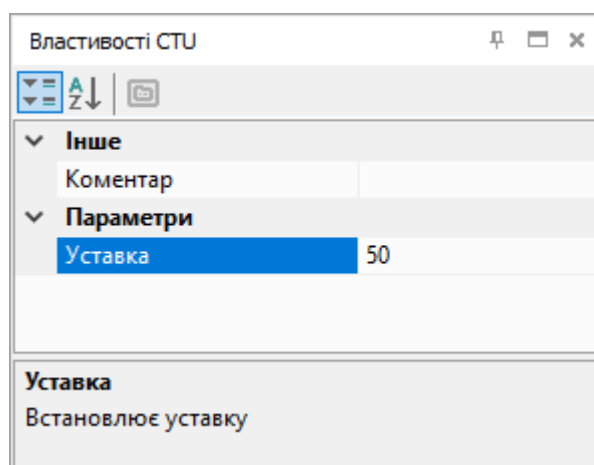


Рисунок 6.81

6.2.5 Регулятори

- [ПІД-регулятор \(PID\) 6.2.5.1](#)

6.2.5.1 ПІД-регулятор (PID)

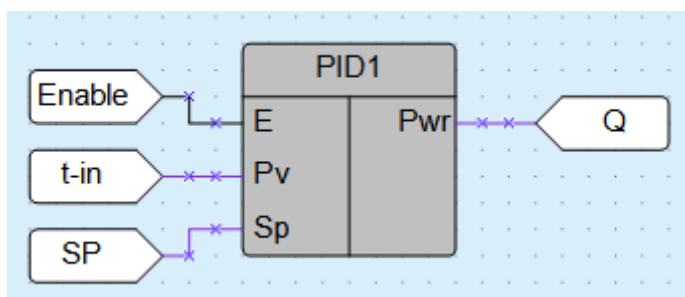


Рисунок 6.82

ПІД-регулятор використовується для реалізації пропорційно-інтегрально-диференціального (ПІД) закону регулювання.

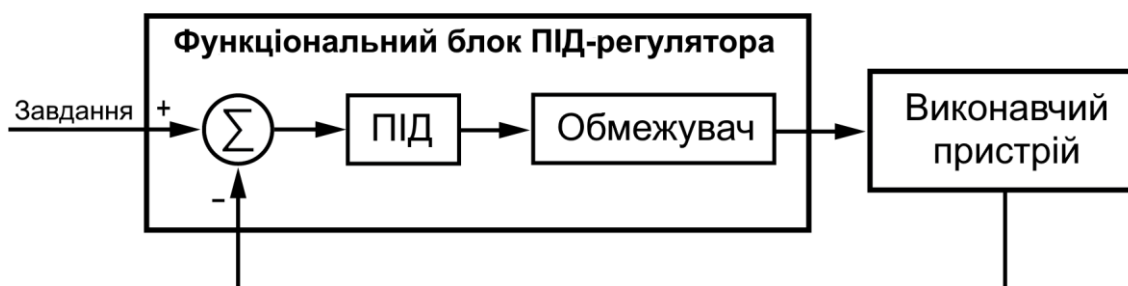


Рисунок 6.83

Входи блоку

Таблиця 6.1 – Входи блоку

Назва	Тип	Опис	Значення
E	Булевський	Дозвіл на роботу блоку. У відключеному стані на вихід блоку подається значення, задане у властивості як «вихідна потужність»	0 – Відключено; 1 – Включено
Pv	З рухомою комою	Поточне значення регульованої величини	
Sp	З рухомою комою	Задане значення регульованої величини	

Виходи блоку

Таблиця 6.2 – Виходи блоку

Назва	Тип	Опис	Значення
Pwr	З рухомою комою	Вихідна потужність, %	0...100

Властивості**Таблиця 6.3 – Опис властивостей**

Назва	Тип	Опис	Значення
Режим роботи регулятора	Булевський	Режими роботи регулятора: <ul style="list-style-type: none"> режим «нагрівач», використовується для керування виконавчими механізмами, вплив яких приводить до зростання значення регульованої величини; режим «холодильник», використовується для керування виконавчими механізмами, вплив яких приводить до зниження значення регульованої величини 	0 – Нагрівач; 1 – Холодильник
Вихідна потужність	З рухомою комою	Вихідна потужність у відключеному стані, %	0...100
Диференціальний коефіцієнт	З рухомою комою	Диференціальний коефіцієнт, з яким працює ПІД	-3,402823E+38...3,402823E +38
Інтегральний коефіцієнт	З рухомою комою	Інтегральний коефіцієнт, з яким працює ПІД	-3,402823E+38...3,402823E +38
Пропорційний коефіцієнт	З рухомою комою	Пропорційний коефіцієнт, з яким працює ПІД	0...100
Мінімальна потужність (значення за умовчанням)	З рухомою комою	Нижнє обмеження потужності, що видається, %	0...100 (20)
Максимальна потужність (значення за умовчанням)	З рухомою комою	Верхнє обмеження потужності, що видається, %	0...100 (80)
Запуск АНР	Булевський	Змінна, яка генерує запуск автоналаштування регулятора у разі подання значення «1». Значення цієї змінної визначається за допомогою блоку «Запис в ФБ»	0 – Стоп; 1 – Запуск

Автоналаштування

Автоналаштування ПІД-регулятора проводиться за допомогою блоків [читання та запису ФБ 3.3.6](#) (

 
 ).

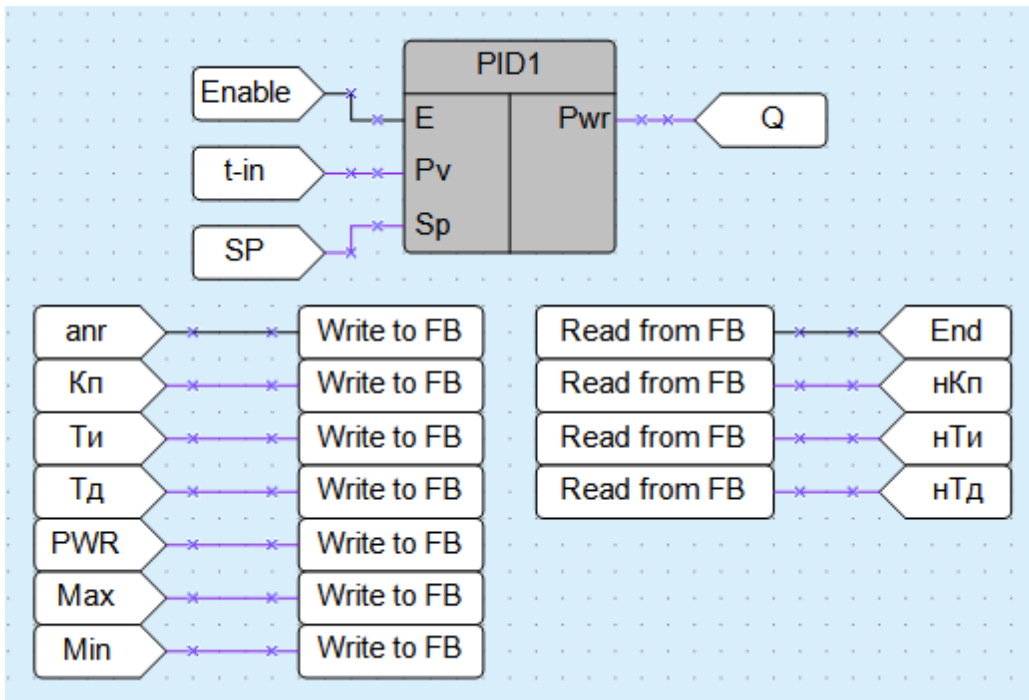


Рисунок 6.84

Для запуску автоналаштування слід додати блок **Запис у ФБ** та прив'язати його до змінної **Запуск АНР** ПІД-регулятора.

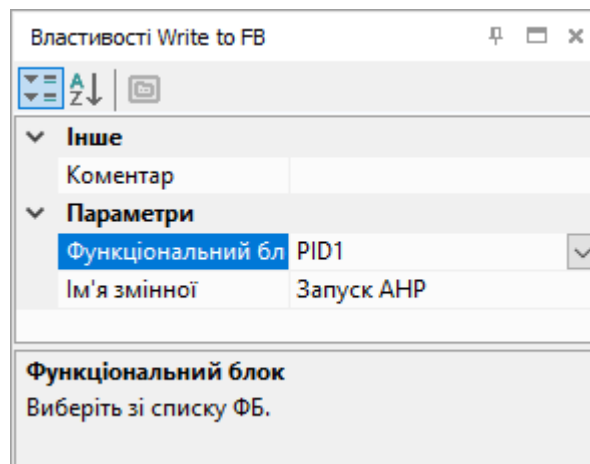


Рисунок 6.85

Значення інших параметрів ФБ можна задавати за допомогою блоку **Запис у ФБ**, як показано на рисунку вище, або налаштовувати на панелі властивостей.

За допомогою блоків **Читання з ФБ** зчитуються значення параметрів **Розрахований пропорційний коефіцієнт**, **Розрахований інтегральний коефіцієнт**, **Розрахований диференційний коефіцієнт** та **Прапор закінчення автоналаштування**.

Для старту автоналаштування слід подати сигнал логічного «1» на вхід **Е**.

Після завершення процесу налаштування для читання доступні нові значення коефіцієнтів: **Розрахований пропорційний коефіцієнт**, **Розрахований час інтегрування** та **Розрахований час диференціювання**. Параметр **Прапор закінчення автоналаштування** виставляється у значення логічної «1». Якщо після автоналаштування скинути вхід **Запуск АНР** у значення логічного «0», скинеться і прапор закінчення.

УВАГА

Прапор закінчення автоналаштування перебуває у стані логічної «1» один цикл.

Якщо скинути вхід **Запуск АНР** у значення логічного «0» до закінчення налаштування, то процес зупиняється, прапор закінчення не виставляється, нові значення коефіцієнтів не обчислюються.

Під час налаштування на вихід ПІД-регулятора подається тестова потужність, обмежена значеннями параметрів **Максимальна потужність** та **Мінімальна потужність**.

Послідовність автоналаштування для режиму «Нагрівач»:

1. Поточне значення менше уставки, на вихід блоку подається максимальна потужність (відповідно до налаштувань).
2. Як тільки поточне значення стане більше уставки, на вихід блоку подається мінімальна потужність.
3. Повторення кроків 1 та 2 ще один раз.
4. Розраховані параметри ПІД-регулятора подаються на відповідні виходи, і виставляється прапор закінчення.

Якщо в налаштуваннях вказано значення максимальної потужності, при якій неможливо досягти уставки, процес автоналаштування не закінчиться, доки він не буде скинутий вручну.

6.3 Макроси проекту

Розділ **Макроси проекту** містить створені користувачем або завантажені за допомогою [менеджера компонентів 3.10](#) макроси.

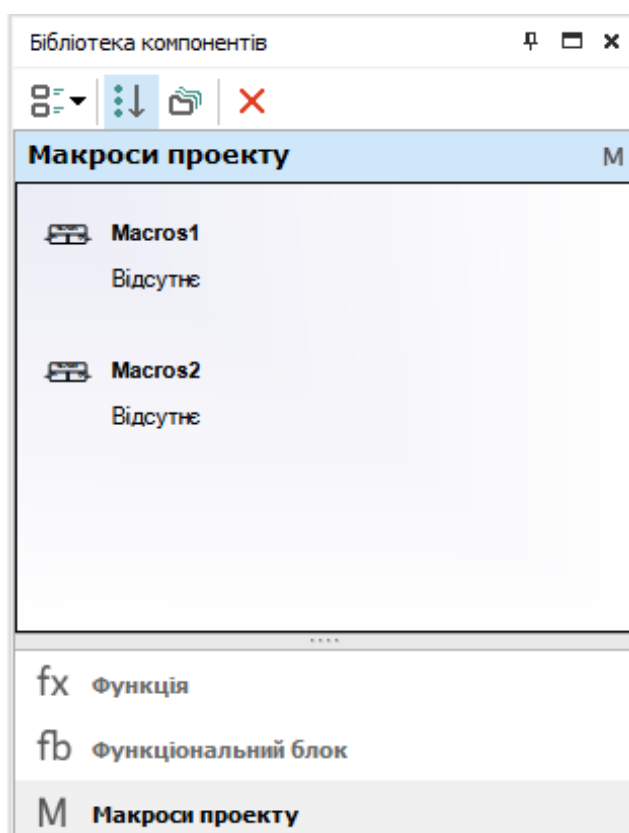


Рисунок 6.86

Для додавання макросу до проекту слід перетягнути макрос із панелі [Бібліотека компонентів 2.3](#) на полотно.

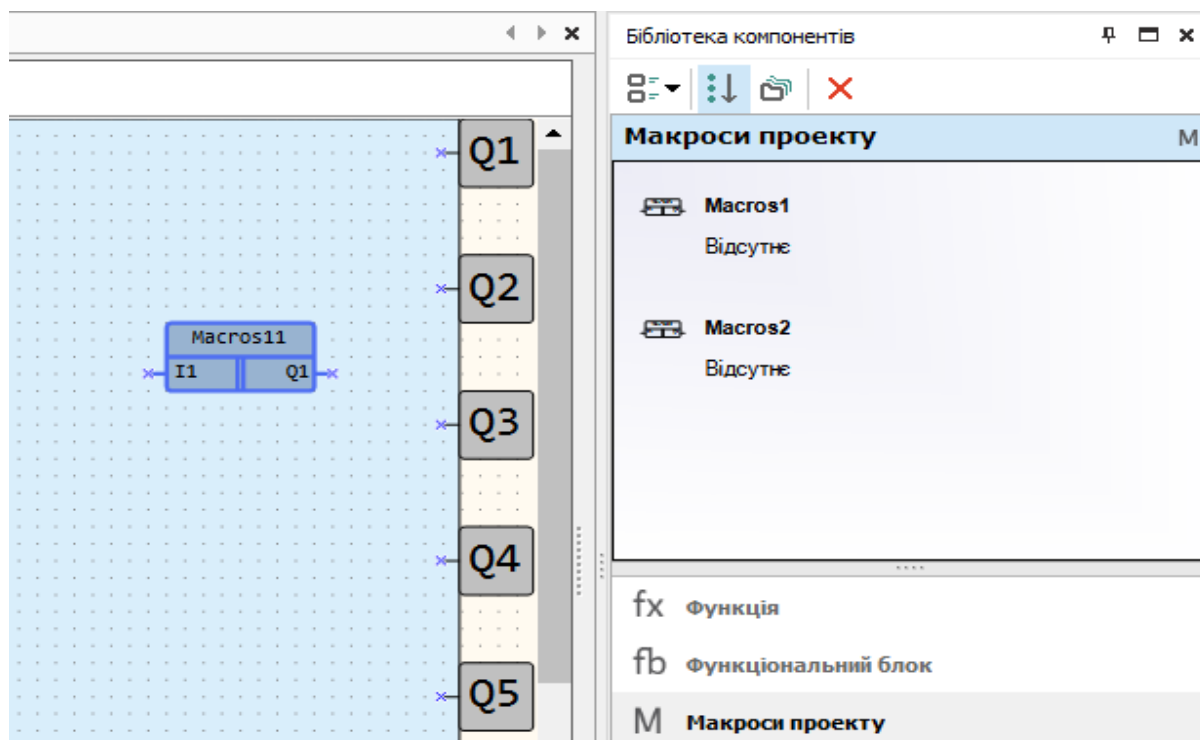


Рисунок 6.87

Щоб видалити макрос із панелі, слід виділити потрібний макрос і натиснути кнопку **X**.
Рекомендації щодо створення макросів наведено у розділі [Робота з макросами 3.11](#).

6.4 Функція на ST

Якщо у проекті [створені функції мовою ST 3.12](#), вони будуть доступні у бібліотеці компонентів.

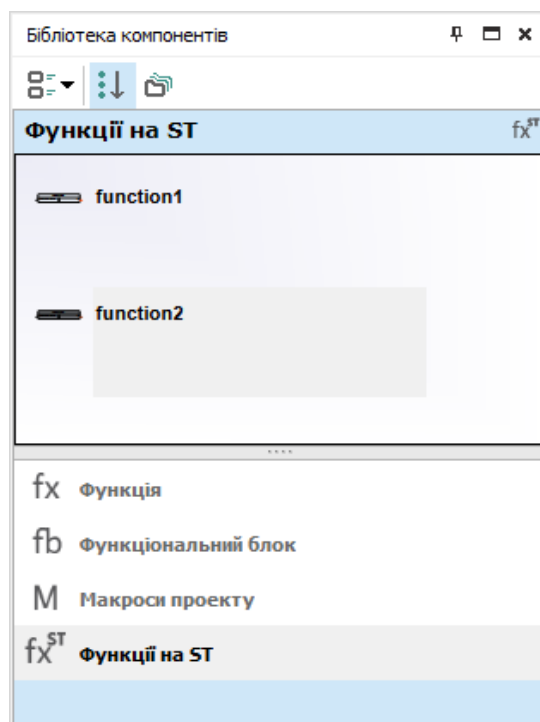



Рисунок 6.88

Місця використання функції

Для перегляду всіх місць використання функції слід:

1. Натиснути ПКМ на ім'я функції.
2. Вибрати пункт  Показати місця використання....

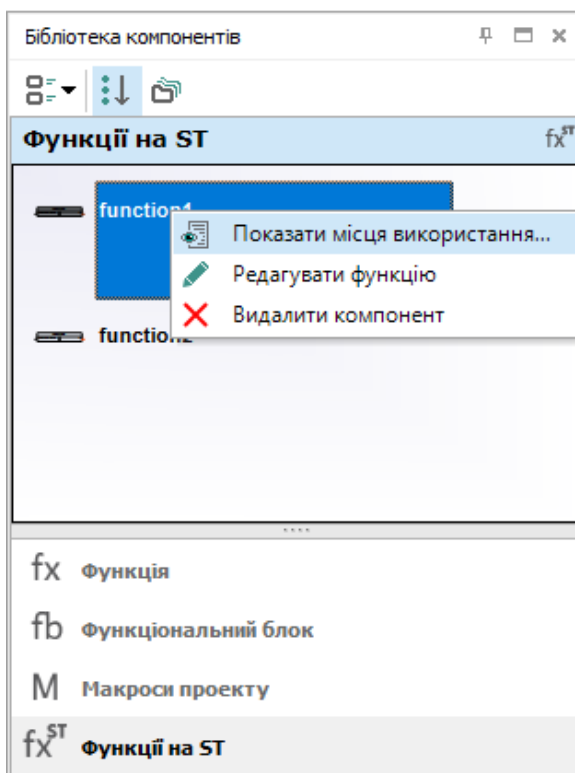


Рисунок 6.89

У нижній частині вікна відкриється панель **Місця використання функції**, на якій з'являться місця використання функції на схемі та в редакторі функцій.

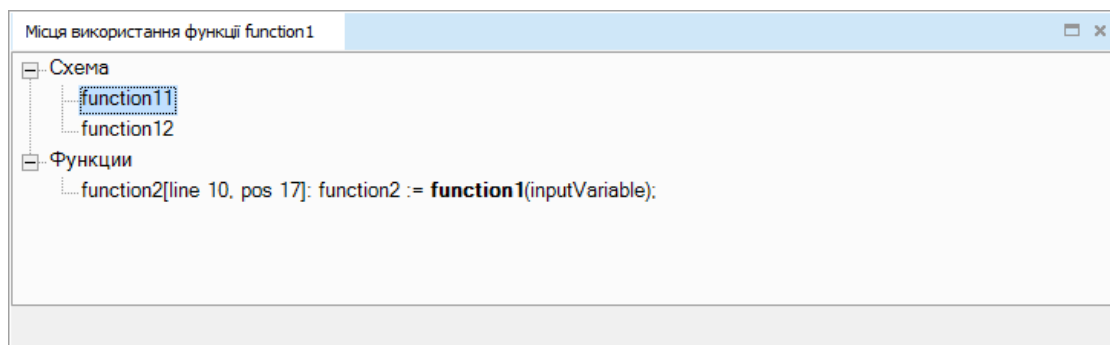


Рисунок 6.90

3. Натиснути ПКМ на рядок із місцем використання функції.

4. Вибрати пункт  **Перейти до місця використання.**

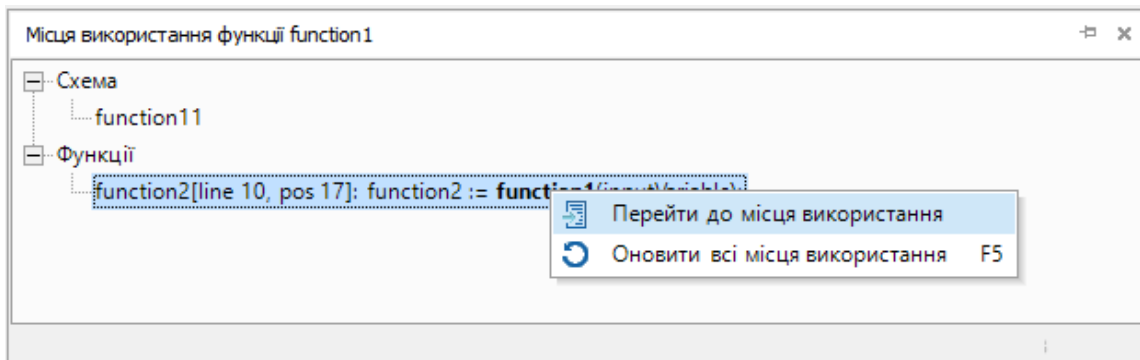



Рисунок 6.91

Фокус переміщується у місця використання функції на схемі або редакторі функції.

**ПРИМІТКА**

Подвійний клік ЛКМ приводить до такого ж результату.

Якщо під час роботи з програмою місця використання функції змінилися, слід оновити панель **Місця використання функції**:

1. Натиснути ПКМ на будь-який рядок панелі.
2. Вибрати пункт  **Оновити всі місця використання.**

Перехід до редактора функцій

Для переходу до [Редактора функцій 3.12](#) слід:

1. Натиснути ПКМ на ім'я функції.
2. Вибрати пункт **Редагувати функцію.**

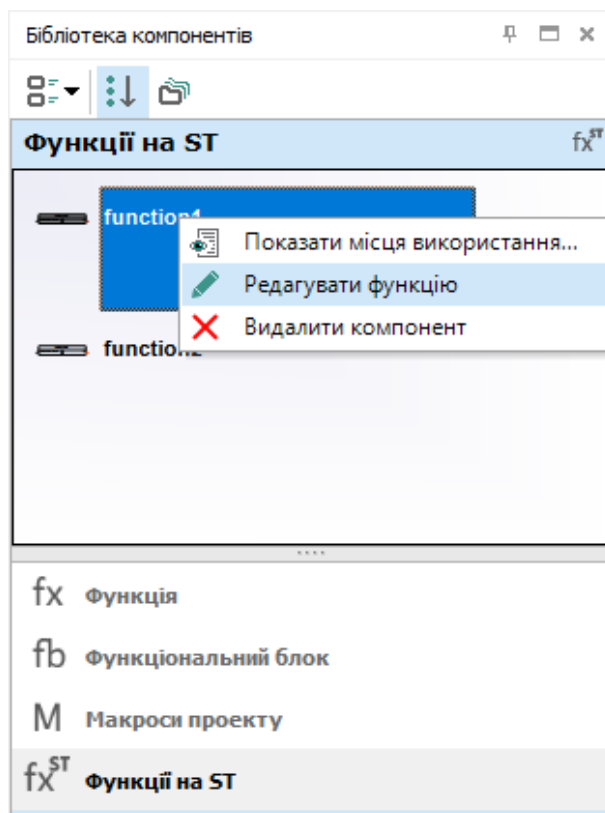


Рисунок 6.92

Відкриється [редактор функції 3.12](#).

Видалення функції

Для видалення функції з проекту слід:

1. Натиснути ПКМ на ім'я функції.
2. Вибрати пункт **Видалити компонент**.

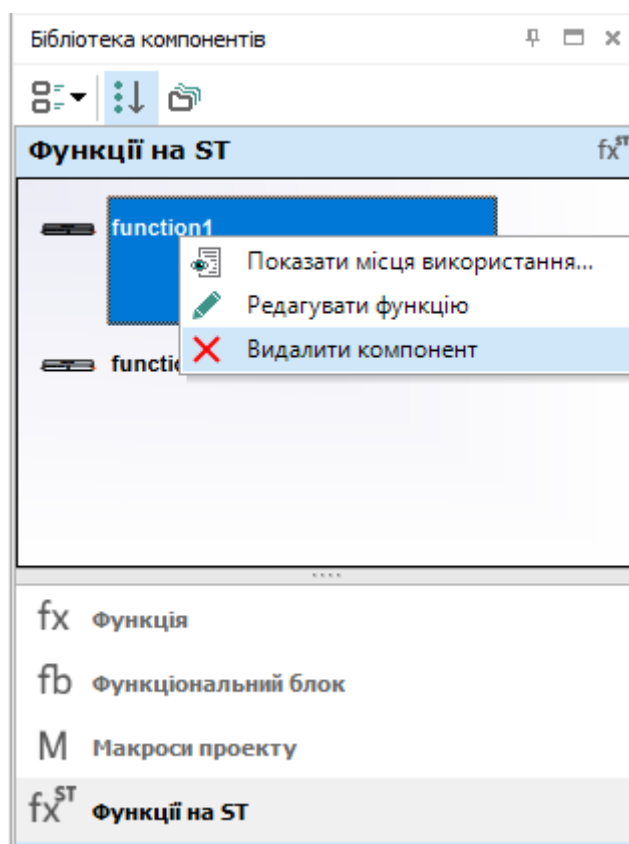


Рисунок 6.93



ПОПЕРЕДЖЕННЯ

Якщо функція використовується на схемі та/або в інших функціях, видалення може призвести до помилок компіляції.

6.5 Елементи керування

- [Мітка 6.5.1](#);
- [Ввід/вивід \(int/float\) 6.5.2](#);
- [Ввід/вивід \(Boolean\) 6.5.3](#);
- [Динамічний текст 6.5.4](#);
- [Комбінований список \(ComboBox\) 6.5.5](#).

6.5.1 Мітка

Елемент **Мітка** призначений для розміщення текстового блоку у рядку екрана.

Налаштування властивостей

Для коректного відображення на екрані пристрою слід вказати координати розташування і ввести текст на панелі властивостей.

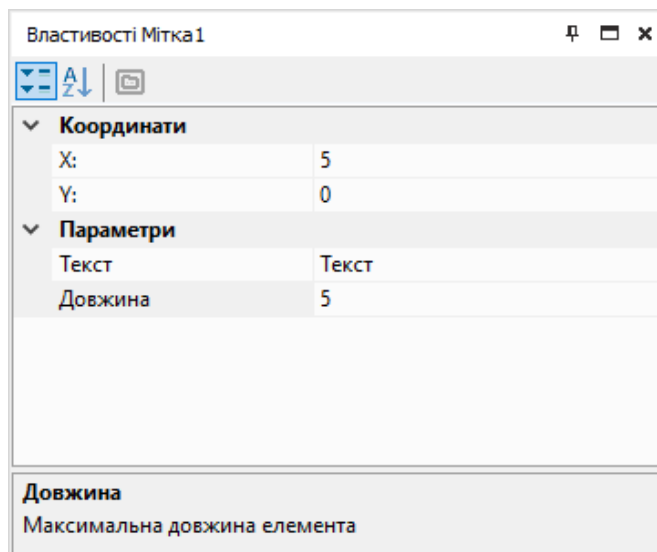


Рисунок 6.94

Координати

Для розташування першого символу елемента по осях X та Y слід вказати координати. Відлік координат по обох осях починається з 0:

- по осі X – зліва направо від 0 до 15 (залежно від властивості **Довжина**);
- по осі Y — зверху вниз (залежно від кількості рядків екрана, встановленої у редакторі екранів).

Значення координат (ціле число) можна задати у меню, що випадає, в рядку координати (X і Y налаштовуються однаково):

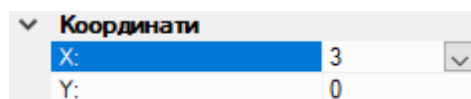


Рисунок 6.95

У меню, що випадає, можна вибрати спосіб задання координат: константа або змінна. Для вибору постійного розташування елемента слід вказати значення координати у полі введення.

Для створення біжучого рядка слід вибрати варіант **Змінна** і натиснути кнопку **Вибір**. Відкриється [таблиця змінних 5](#), у ній слід вибрати змінну цілочисельного типу, яка визначатиме значення координати, і натиснути кнопку **ОК**. Вибрана змінна відобразиться у полі введення:

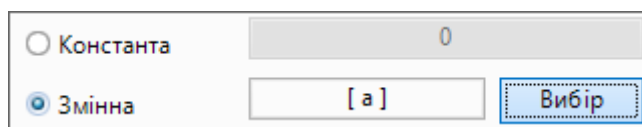


Рисунок 6.96

Параметри

- **Текст** — у поле вводиться текст, який відобразиться на екрані пристрою. Довжина тексту не повинна перевищувати значення, вказаного для параметра **Довжина**.
- **Довжина** — максимальна кількість символів блоку, що відображаються.

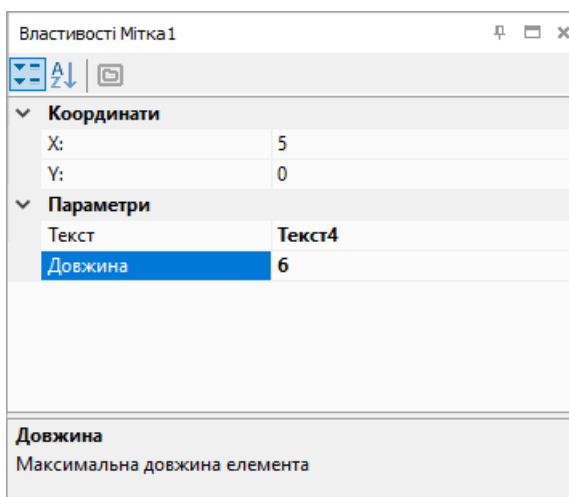


Рисунок 6.97

Відображення

У редакторі екранів відображається, як виглядатиме елемент **Мітка** на екрані пристрою.

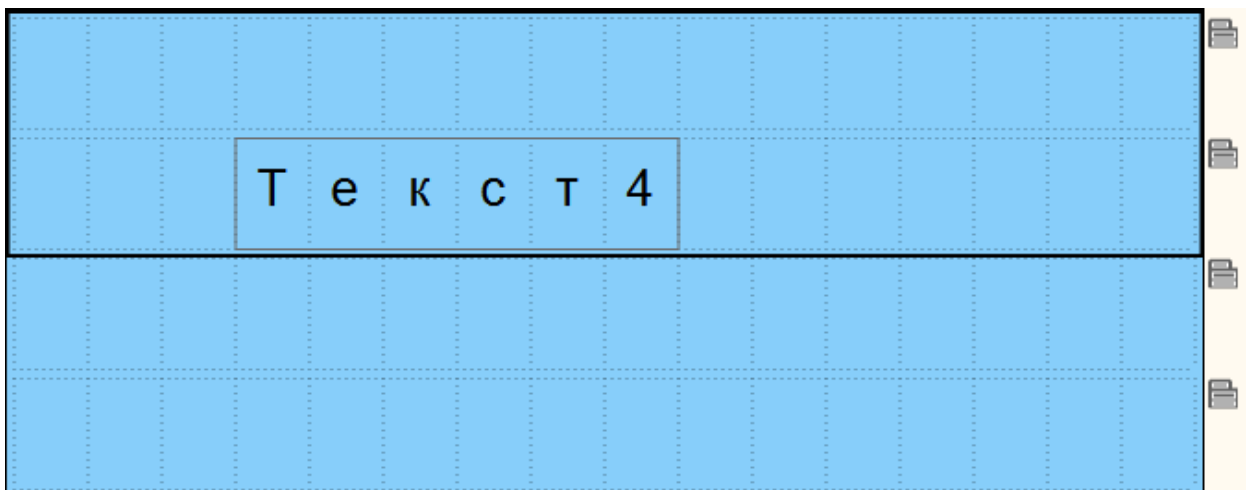


Рисунок 6.98

6.5.2 Ввід/вивід (int/float)

Елемент **Ввід/вивід (int/float)** призначений для виведення на екран пристрою або читання з екрана значення цілочисельної чи дійсної змінної.

Налаштування властивостей

Для коректної роботи елемента слід вказати координати розташування та налаштувати параметри введення/виведення на панелі властивостей.

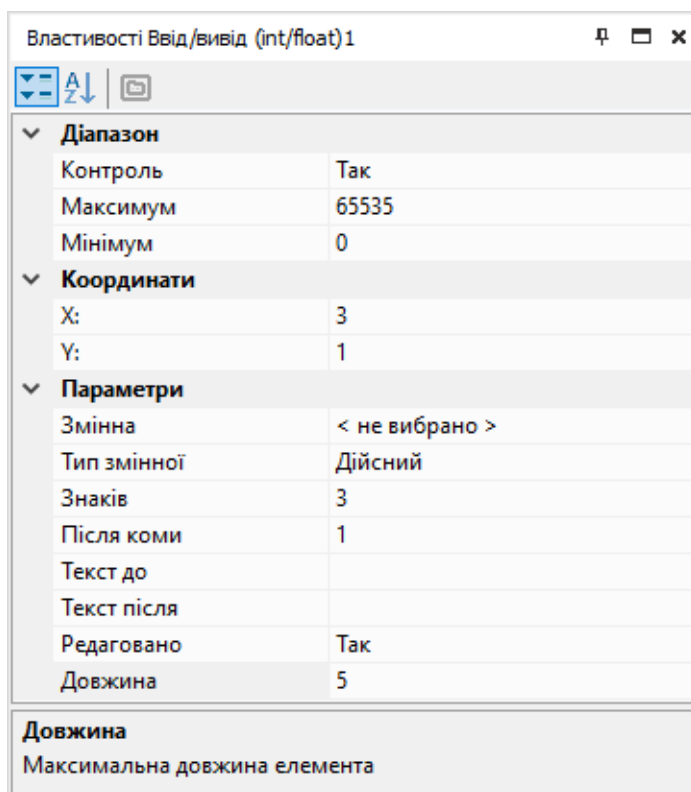


Рисунок 6.99

Діапазон

- **Контроль** — обмеження значення, яке користувач вводить з лицьової панелі пристрою. Обмеження не поширюється на значення, отримане під час роботи програми;
- **Максимум** — максимальне значення параметра для введення;
- **Мінімум** — мінімальне значення параметра для введення.

Координати

Для розташування першого символу елемента по осях X та Y слід вказати координати. Відлік координат по обох осях починається з 0:

- по осі X — зліва направо від 0 до 15 (залежно від властивості **Довжина**);
- по осі Y — зверху вниз (залежно від кількості рядків екрана, встановленої у редакторі екранів).

Значення координат (ціле число) можна задати у меню, що випадає, в рядку координати (X і Y налаштовуються однаково):

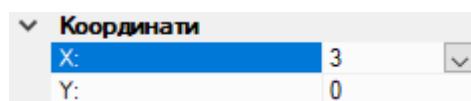


Рисунок 6.100

У меню, що випадає, можна вибрати спосіб задання координат: константа або змінна. Для вибору постійного розташування елемента слід вказати значення координати у полі введення.

Для створення біжучого рядка слід вибрати варіант **Змінна** і натиснути кнопку **Вибір**. Відкриється [таблиця змінних 5](#), у ній слід вибрати змінну цілочисельного типу, яка визначатиме значення координати, і натиснути кнопку **ОК**. Вибрана змінна відобразиться у полі введення:

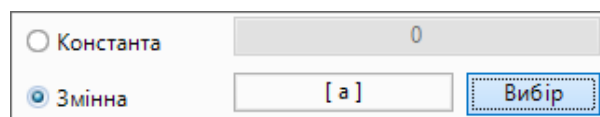


Рисунок 6.101

Параметри

- **Змінна** — прив'язка до змінної проекту (цілочисельної або з рухомою точкою). Для вибору змінної слід натиснути кнопку «...» і вибрати в [таблиці змінних 5](#);

- **Тип змінної** — вибір типу змінної: цілочисельний чи дійсний. Якщо змінна ще не прив'язана, необхідно задати тип;
- **Знаків** — сумарна кількість знаків, що відображаються до і після коми;
- **Після коми** — кількість знаків після коми: до 6 знаків або визначається автоматично (Авто). Докладнішу інформацію читайте в *настанові щодо експлуатування* пристрою;
- **Текст до** — текст зліва від числового значення;
- **Текст після** — текст справа від числового значення;
- **Редаговано** — якщо вибрано **Так**, значення прив'язаної змінної можна змінювати за допомогою кнопок на лицьовій панелі пристрою;
- **Довжина** — загальна максимальна довжина поля, що включає текст до та після значення змінної і власне значення змінної.

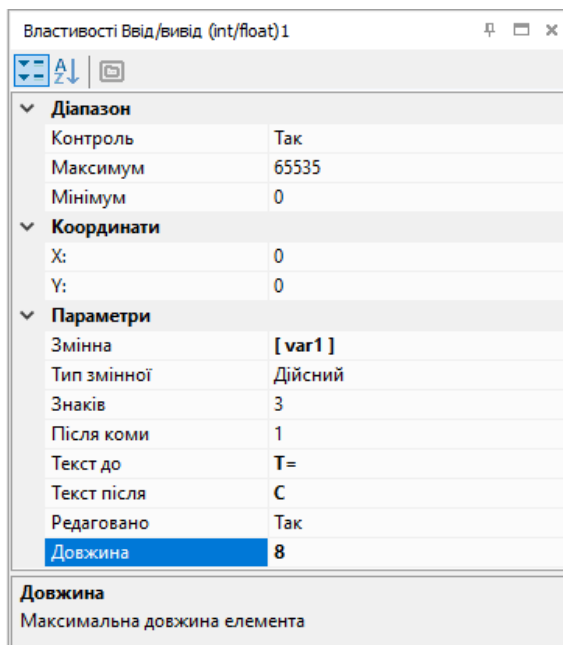


Рисунок 6.102

Відображення

У редакторі екранів відображається, як виглядатиме елемент **Ввід/вивід (int/float)** на екрані пристрою.

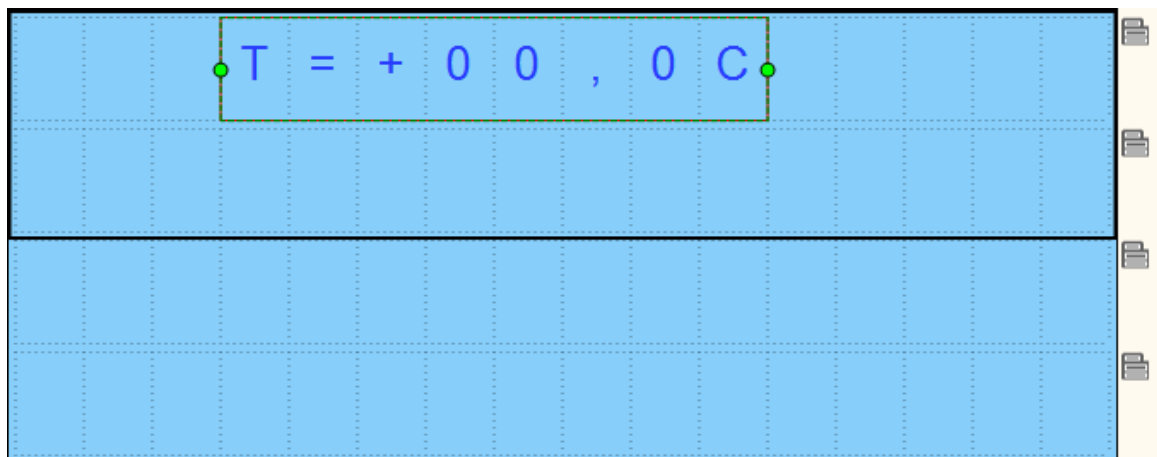


Рисунок 6.103

6.5.3 Ввід/вивід (boolean)

Елемент **Ввід/вивід (boolean)** призначений для виведення на екран або читання з екрана значення булевської змінної.

Налаштування властивостей

Для коректної роботи елемента слід вказати координати розташування та налаштувати параметри введення/виведення на панелі властивостей.

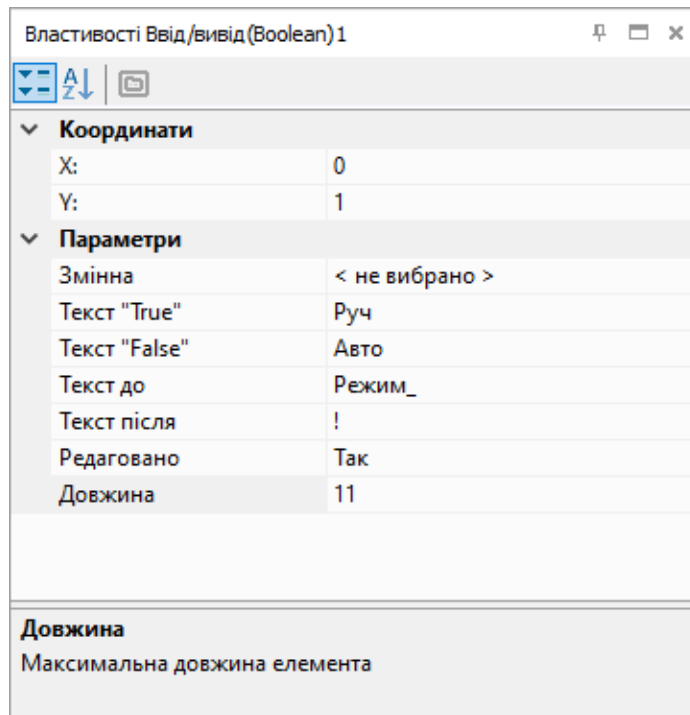


Рисунок 6.104

Координати

Для розташування першого символу елемента по осях X та Y слід вказати координати. Відлік координат по обох осях починається з 0:

- по осі X – зліва направо від 0 до 15 (залежно від властивості **Довжина**);
- по осі Y — зверху вниз (залежно від кількості рядків екрана, встановленої у редакторі екранів).

Значення координат (ціле число) можна задати у меню, що випадає, в рядку координати (X і Y налаштовуються однаково):

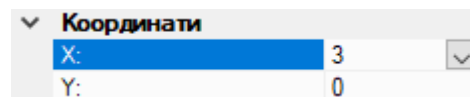


Рисунок 6.105

У меню, що випадає, можна вибрати спосіб задання координат: константа або змінна. Для вибору постійного розташування елемента слід вказати значення координати у полі введення.

Для створення біжучого рядка слід вибрати варіант **Змінна** і натиснути кнопку **Вибір**. Відкриється [таблиця змінних 5](#), у ній слід вибрати змінну цілочисельного типу, яка визначатиме значення координати, і натиснути кнопку **ОК**. Вибрана змінна відобразиться у полі введення:

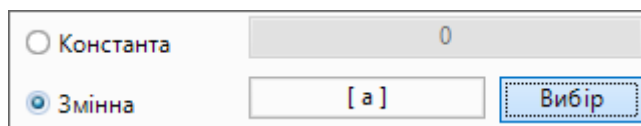


Рисунок 6.106

Параметри

- **Змінна** — прив'язка до булевської змінної проекту. Для вибору змінної слід натиснути кнопку «...» і вибрати в [таблиці змінних 5](#);
- **Текст «True»** — текст, що відповідає значенню змінної, яка дорівнює «1»;
- **Текст «False»** — текст, що відповідає значенню змінної, яка дорівнює «0»;
- **Текст до** — текст зліва від значення, що виводиться;
- **Текст після** — текст справа від значення, що виводиться;
- **Редаговано** — якщо вибрано значення **Так**, значення прив'язаної змінної можна змінювати за допомогою кнопок на лицьовій панелі пристрою;
- **Довжина** — загальна максимальна довжина поля, що включає текст до та після значення змінної і власне значення змінної.

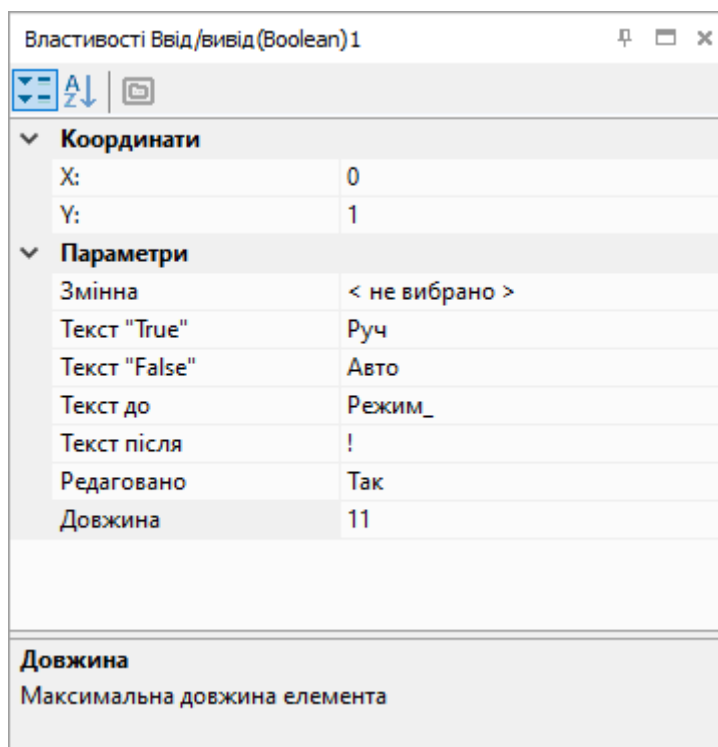


Рисунок 6.107

Відображення

У редакторі екранів відображається, як виглядатиме елемент **Ввід/вивід (boolean)** на екрані пристрою.

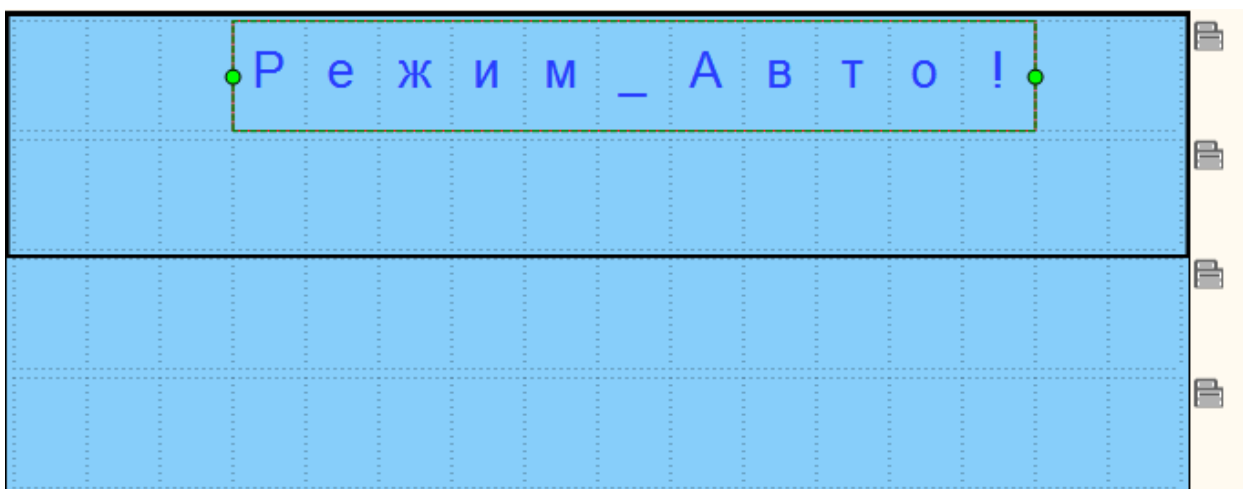


Рисунок 6.108

6.5.4 Динамічний текст

Елемент **Динамічний текст** призначений для виведення на екран пристрою одного з кількох текстових рядків залежно від значення прив'язаної змінної.

Налаштування властивостей

Для коректного відображення на екрані пристрою слід вказати координати розташування і ввести текст на панелі властивостей.

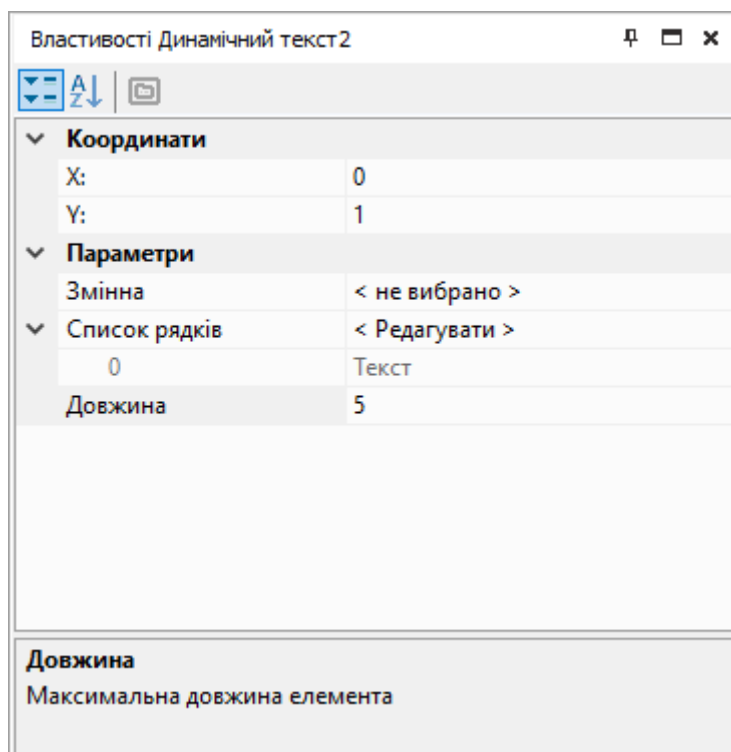


Рисунок 6.109

Координати

Для розташування першого символу елемента по осях X та Y слід вказати координати. Відлік координат по обох осях починається з 0:

- по осі X – зліва направо від 0 до 15 (залежно від властивості **Довжина**);
- по осі Y — зверху вниз (залежно від кількості рядків екрана, встановленої у редакторі екранів).

Значення координат (ціле число) можна задати у меню, що випадає, в рядку координати (X і Y налаштовуються однаково):

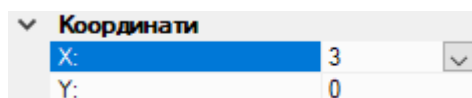


Рисунок 6.110

У меню, що випадає, можна вибрати спосіб задання координат: константа або змінна. Для вибору постійного розташування елемента слід вказати значення координати у полі введення.

Для створення біжучого рядка слід вибрати варіант **Змінна** і натиснути кнопку **Вибір**. Відкриється [таблиця змінних 5](#), у ній слід вибрати змінну цілочисельного типу, яка визначатиме значення координати, і натиснути кнопку **ОК**. Вибрана змінна відобразиться у полі введення:

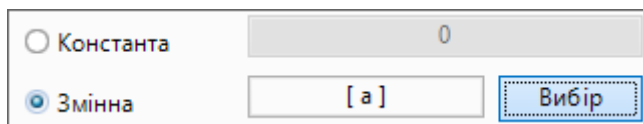


Рисунок 6.111

Параметри

- **Змінна**— прив'язка до цілочисельної змінної проекту. Для вибору змінної слід натиснути кнопку «...» і вибрати в [таблиці змінних 5](#);
- **Список рядків** — таблиця, **Значення** рядка якої виводиться на екран, коли номер (**№**) рядка відповідає значенню прив'язаної цілочисельної змінної (**Змінна**). У стовпці **Символів** вказується кількість символів у рядку, піктограма знака оклику виводиться у разі перевищення значення параметра **Довжина**;
- **Довжина** – максимальна довжина поля. У разі перевищення довжини, текст буде відображатися не повністю.

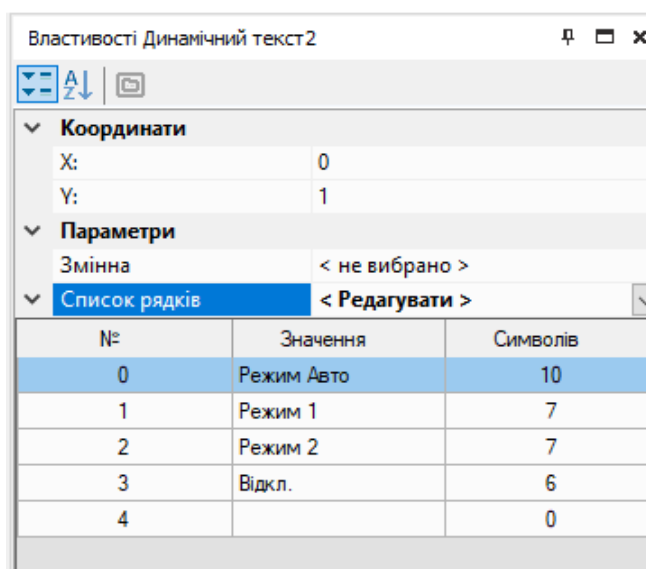


Рисунок 6.112

Відображення

У редакторі екранів відображається, як виглядатиме на екрані пристрою елемент **Динамічний текст**.

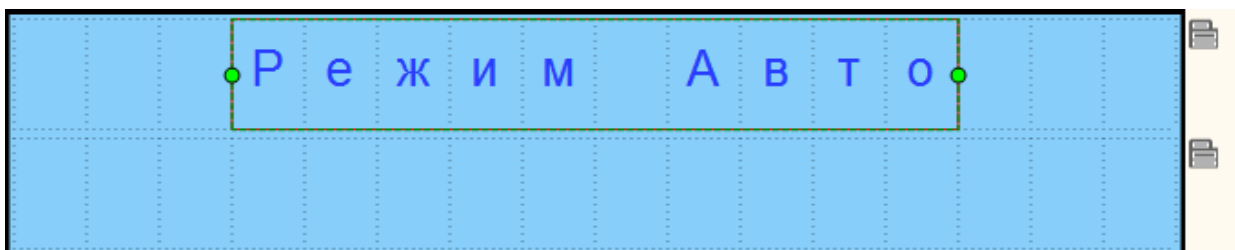


Рисунок 6.113

6.5.5 Комбінований список (ComboBox)

Елемент **Комбінований список (ComboBox)** використовується для вибору одного рядка з кількох текстових рядків кнопками пристрою та запису у прив'язану змінну значення, що відповідає вибору.

Налаштування властивостей

Для коректної роботи елемента слід вказати координати розташування та налаштувати параметри змінних і рядків на панелі властивостей.

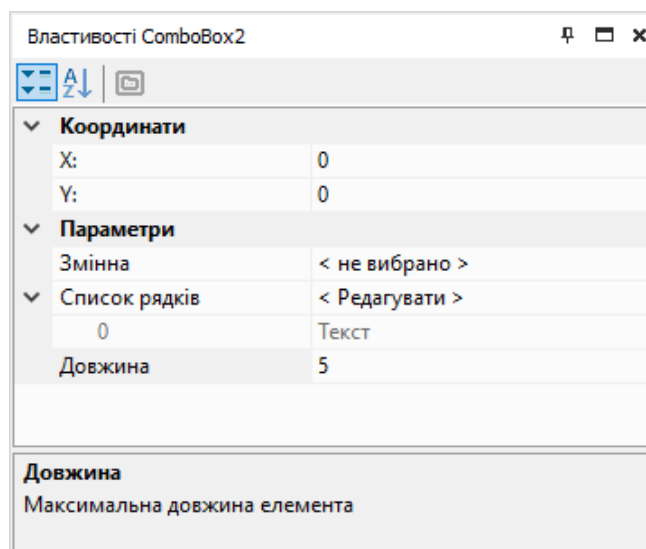


Рисунок 6.114

Координати

Для розташування першого символу елемента по осях X та Y слід вказати координати. Відлік координат по обох осях починається з 0:

- по осі X – зліва направо від 0 до 15 (залежно від властивості **Довжина**);
- по осі Y — зверху вниз (залежно від кількості рядків екрана, встановленої у редакторі екранів).

Значення координат (ціле число) можна задати у меню, що випадає, в рядку координати (X і Y налаштовуються однаково):

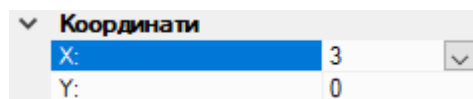


Рисунок 6.115

У меню, що випадає, можна вибрати спосіб задання координат: константа або змінна. Для вибору постійного розташування елемента слід вказати значення координати у полі введення.

Для створення біжучого рядка слід вибрати варіант **Змінна** і натиснути кнопку **Вибір**. Відкриється [таблиця змінних 5](#), у ній слід вибрати змінну цілочисельного типу, яка визначатиме значення координати, і натиснути кнопку **ОК**. Вибрана змінна відобразиться у полі введення:

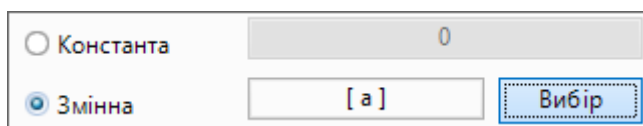


Рисунок 6.116

Параметри

- **Змінна** — прив'язка до цілочисельної змінної проекту. Щоб вибрати змінну, слід натиснути кнопку «...» і вибрати в [таблиці змінних 5](#).
- **Список рядків** — таблиця, номер (**№**) кожного рядка якої слугує для запису в прив'язану цілочисельну змінну при виборі **Значення** на дисплеї пристрою. У стовпці **Символів** вказується кількість символів у рядку, знак оклику виводиться при перевищенні значення параметра **Довжина**.
- **Довжина** – максимальна довжина поля. У разі перевищення довжини, текст буде відображатися не повністю.

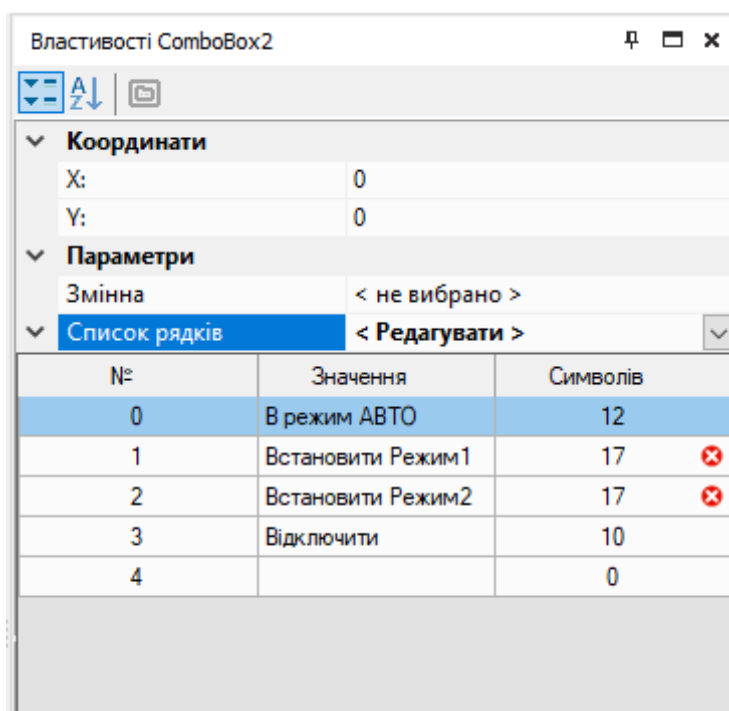


Рисунок 6.117

Відображення

У редакторі екранів відображається, як виглядатиме елемент **Комбінований список (ComboBox)** на екрані пристрою.

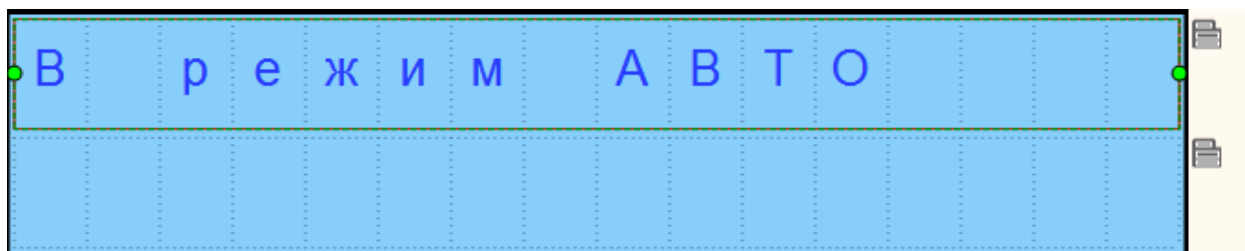


Рисунок 6.118

7 Робота з пристроєм

У цьому розділі описуються функції роботи та конфігурації пристрою:

- [Інформація про прилад 7.1](#);
- [Час циклу 7.2](#);
- [Оновлення вбудованого ПЗ 7.3](#);
- [Юстування 7.4](#).

7.1 Інформація про пристрій

Щоб отримати інформацію про підключений пристрій, натисніть кнопку  на панелі інструментів або виберіть у головному меню **Прилад** → **Інформація...**

Відкриється вікно з інформацією про підключений пристрій:

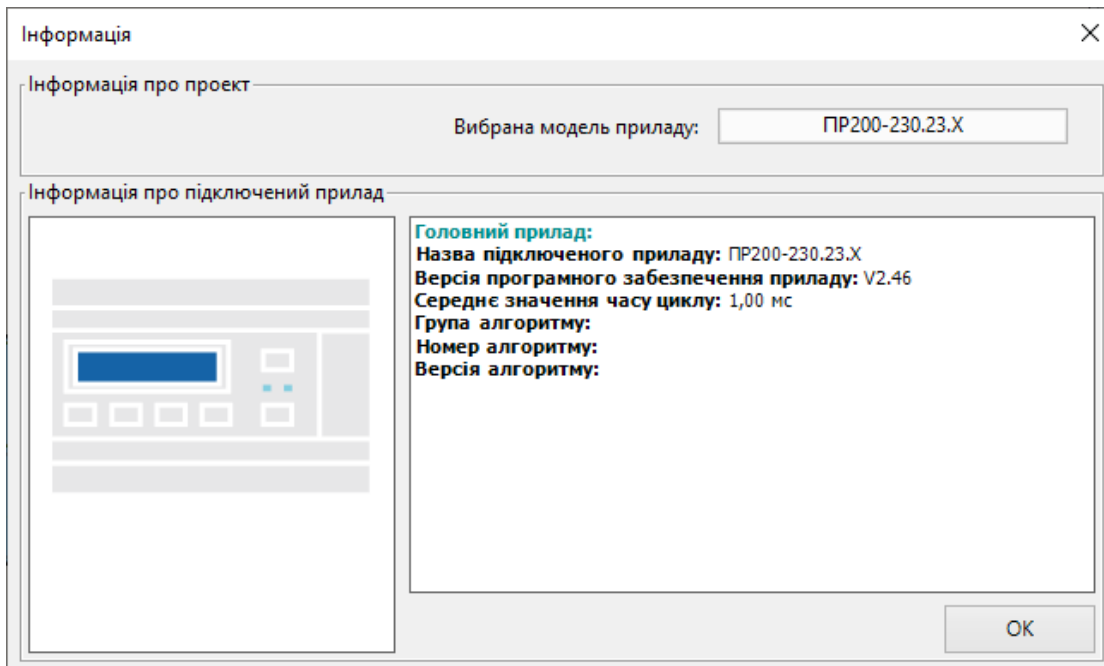


Рисунок 7.1

Інформація про проект:

- **Вибрана модель приладу** – модель та модифікація пристрою, вибрані під час створення проекту.

Інформація про підключений пристрій:

- **Назва підключеного приладу** – модель та модифікація підключеного пристрою;
- **Версія ПЗ приладу** – версія вбудованого ПЗ підключеного пристрою;
- **Середнє значення часу циклу** – середній часовий проміжок виконання операції в програмі, залежить від складності завантаженої програми (див. розділ [Час циклу 7.2](#));
- **Тип заказного елемента №** – тип вихідного елемента підключеного пристрою (дискретний, аналоговий);
- **Група, номер, версія алгоритма** – значення, задані користувачем для контролю версій завантаженої програми у вікні [Відомості про проект 3.9](#).

Кнопка **Експортувати у проект** дозволяє експортувати у створений проект налаштування входів/виходів із підключеного пристрою.

7.2 Час циклу

Час циклу – це час виконання робочого циклу пристрою, а саме:

- опитування стану фізичних входів пристрою та копіювання їх значень у комірки пам'яті;
- обробка програми;

- читання/запис мережевих змінних програми;
- запис результатів роботи програми у фізичні виходи пристрою.

За умовчанням час циклу дорівнює **1 мс**. Пристрій підлаштовує час циклу залежно від складності програми.

Умови збільшення часу циклу:

- зростає складність алгоритму (задіяна велика кількість ФБ та макросів);
- у програмі використовується велика кількість мережевих змінних;
- використовується велика кількість елементів керування даними за допомогою дисплея пристрою.

Користувач не може встановити час циклу. Якщо пристрій оснащений дисплеєм, то поточний час циклу можна переглянути у системному меню пристрою. Якщо пристрій підключено до ПК, час циклу можна переглянути у вікні [Інформація про прилад 7.1](#).

7.3 Оновлення вбудованого ПЗ

Оновлення вбудованого ПЗ у пристрої

Якщо для підключеного пристрою випущено нову версію вбудованого ПЗ, то під час завантаження програми в пристрій буде запропоновано оновити вбудоване ПЗ:

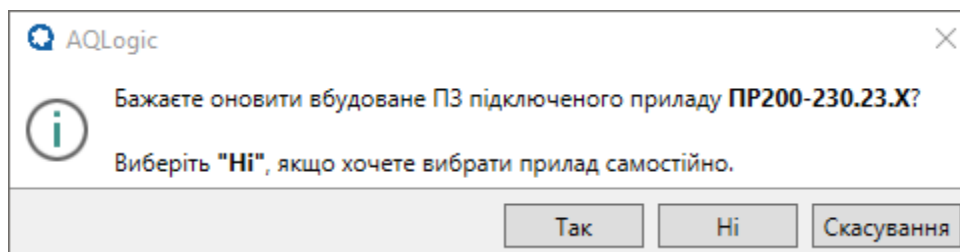


Рисунок 7.2

Для запуску оновлення вбудованого ПЗ слід натиснути кнопку **Так**.

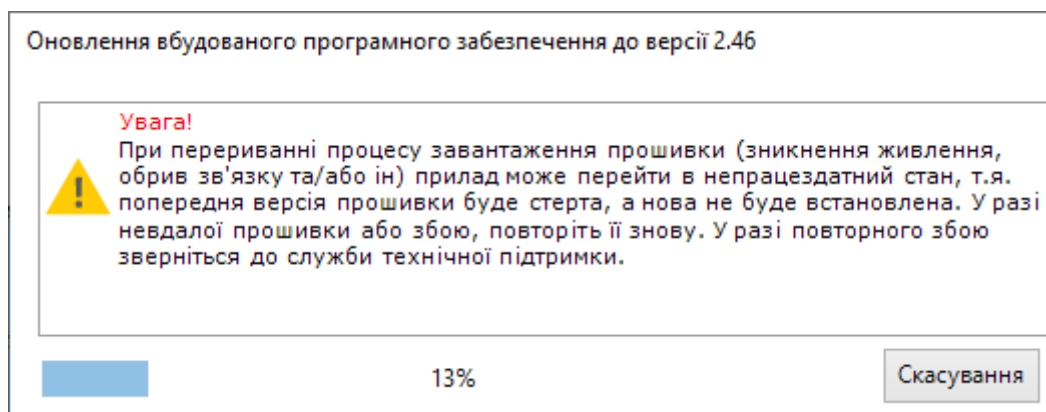


Рисунок 7.3

Без оновлення вбудованого ПЗ програма не буде завантажена в пристрій.

Розрив зв'язку між ПК та пристроєм під час оновлення призведе до пошкодження вбудованого ПЗ та нероботоздатності пристрою. Для відновлення роботоздатності пристрою слід повторити операцію.

Відновлення пошкодженого вбудованого ПЗ пристрою

Для відновлення пошкодженого вбудованого ПЗ слід:

1. Якщо пристрій справний, то перевести його в режим завантажувача (див. *настанову щодо експлуатування пристрою*).

- У головному меню вибрати **Прилад** → **Оновлення вбудованого ПЗ**.

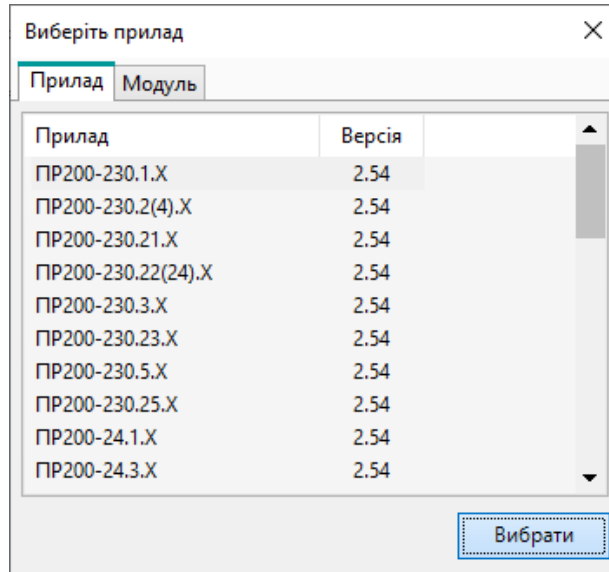


Рисунок 7.4

- У вікні, що відкрилося, на вкладці **Прилад** вибрати модель і модифікацію.
- Натиснути кнопку **Вибрати**. Почнеться процес завантаження, після закінчення система виведе повідомлення про результати завантаження.

Якщо вищезазначений спосіб не спрацює, необхідно звернутися до служби технічної підтримки компанії АКУТЕК.

Оновлення вбудованого ПЗ модуля розширення

Якщо до пристрою підключено модуль розширення і для нього випущено нову версію вбудованого ПЗ, то під час завантаження програми в пристрій буде запропоновано оновити вбудоване ПЗ модуля розширення.

Вбудоване ПЗ для модуля розширення оновлюється за аналогією з пристроєм.

Для самостійного оновлення вбудованого ПЗ модуля розширення слід:

- У головному меню вибрати **Прилад** → **Оновлення вбудованого ПЗ**.
- У вікні, що відкрилося, на вкладці **Модуль** слід вибрати модифікацію і номер слота.

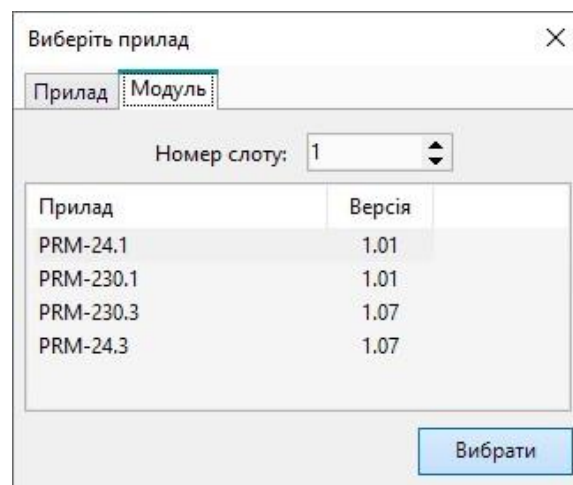


Рисунок 7.5

- Натиснути кнопку **Вибрати**. Почнеться процес завантаження, після закінчення система виведе повідомлення про результати завантаження.

Під час оновлення не переривайте подання живлення на пристрій та модулі розширення.

Якщо в пристрій завантажувється програма без підключеного модуля розширення, можлива ситуація, коли внутрішнє ПЗ пристрою та модуля розширення матимуть несумісні версії. Це призведе до ситуації, що на модулі розширення буде індикувана **Аварія**. Для усунення **Аварії** слід повторно оновити вбудоване ПЗ модуля розширення.

7.4 Юстування



ПОПЕРЕДЖЕННЯ

Необхідність проведення юстування визначається за результатами перевірки пристрою і повинна проводитися лише кваліфікованими спеціалістами метрологічних служб, які здійснюють цю перевірку.



УВАГА

Юстування пристрою слід виконувати лише відповідно до вимог *настанови щодо експлуатування* пристрою. Рекомендується не проводити юстирування самостійно без належних знань та кваліфікації і звернутися до сервісного центру АКУТЕК. Неправильне юстирування призведе до некоректної роботи пристрою.

Для запуску юстування слід вибрати в головному меню **Прилад** → **Юстування входів/виходів....**

Перед початком юстування з'явиться меню вибору моделі та модифікації пристрою.

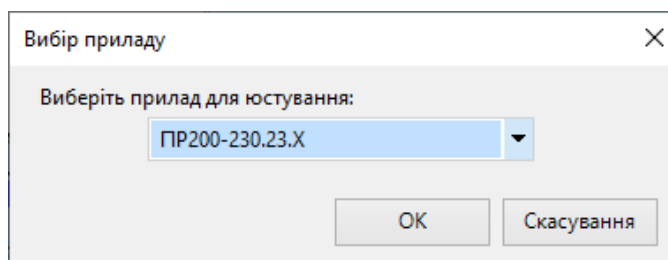


Рисунок 7.6

Далі у діалоговому вікні, що відкрилося, слід вибрати юстирування:

- [аналогових входів 7.4.1](#);
- [аналогових виходів 7.4.2](#).

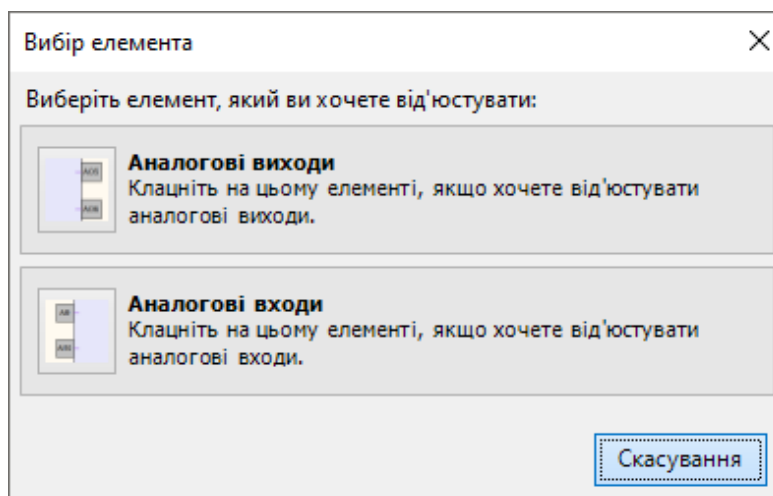


Рисунок 7.7

Якщо в пристрої запущено програму, то під час юстування вона зупиняється – на дисплеї пристрою з'являється напис **Програму LOGIC: Зупинено**.

7.4.1 Юстування входів

Юстування аналогових входів

Для юстування входів слід:

1. У діалоговому вікні юстирування вибрати **Аналогові входи**. Відкриється вікно налаштування.
2. У вікні налаштування вибрати **Тип датчика**.

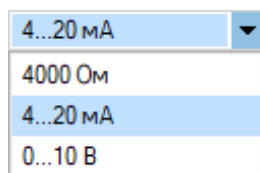


Рисунок 7.8

3. Ввести параметри підключених до входів датчиків.

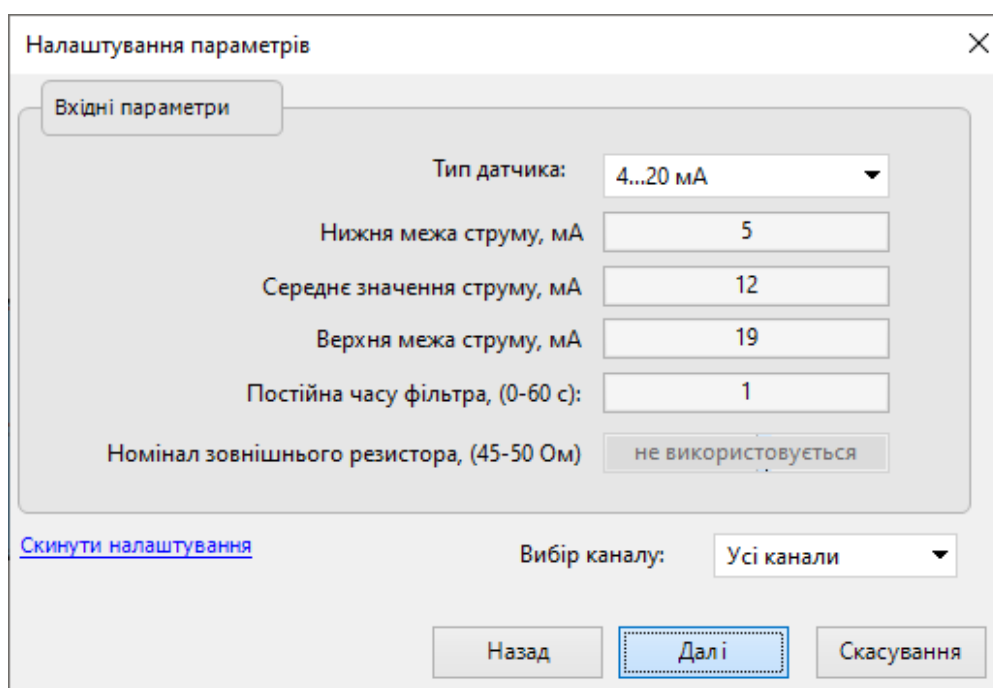


Рисунок 7.9

4. Якщо потрібно відновити значення за умовчанням у полях введення, слід натиснути **Скинути налаштування**.
5. У випадному меню **Вибір каналу** вибрати номер входу для налаштування відповідно до введених параметрів і натиснути **Далі**.

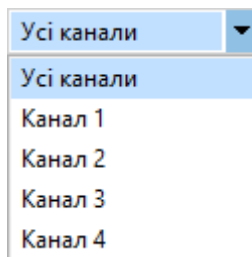


Рисунок 7.10

6. Виконати подальші вказівки щодо встановлення положення перемичок у пристрої.

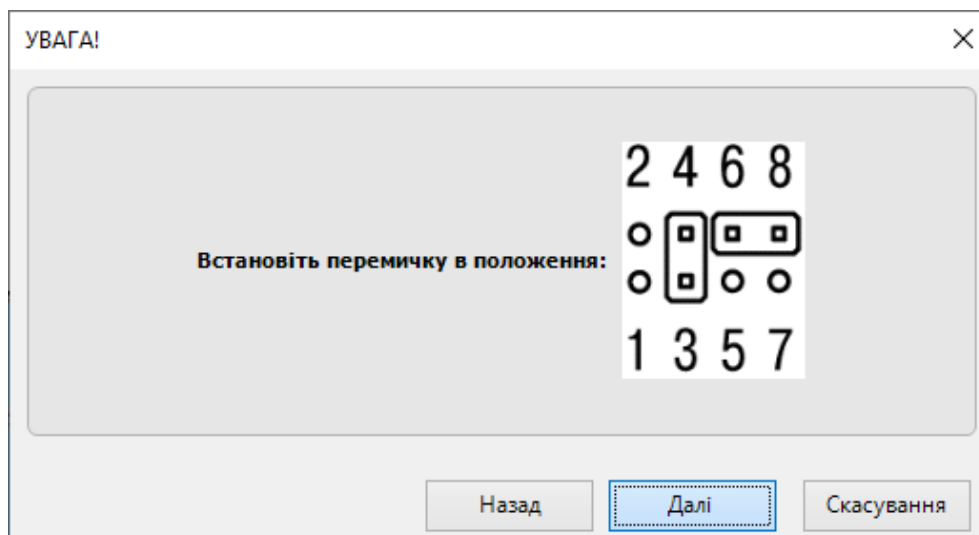


Рисунок 7.11

**ПРИМІТКА**

Установлення перемичок описується у *настанові щодо експлуатування* пристрою.

7. На входи, що юстуються, подати сигнали з необхідними характеристиками.

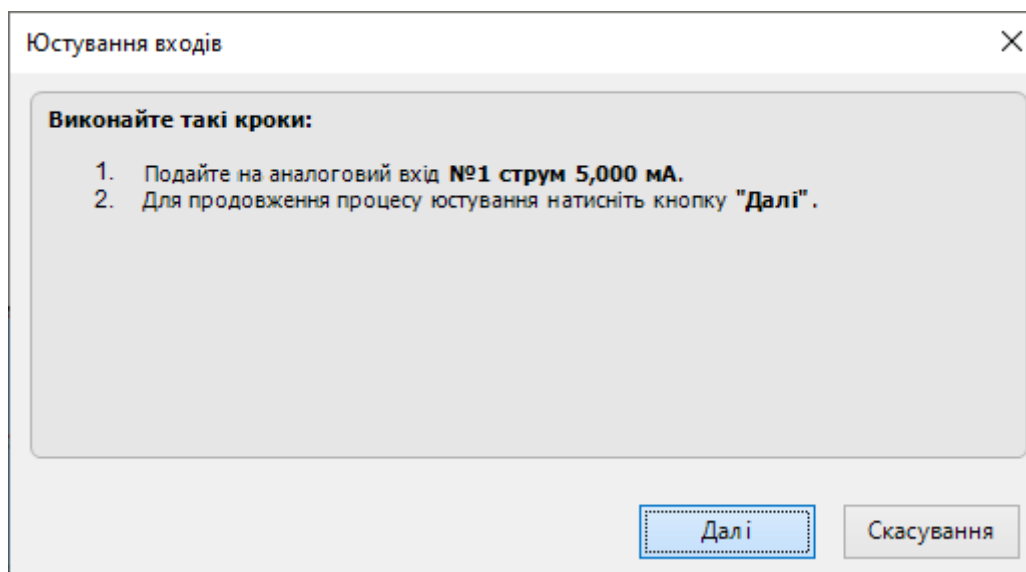


Рисунок 7.12

8. Після юстування з'явиться повідомлення про успішне завершення. Програму пристрою буде знову запущено.

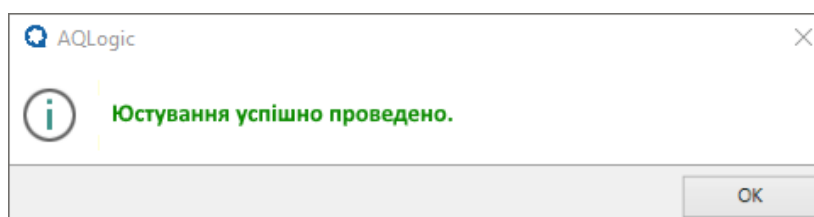


Рисунок 7.13

7.4.2 Юстування виходів

Для юстування виходів слід:

1. У діалоговому вікні юстування вибрати **Аналогові виходи**. Відкриється вікно налаштування.
2. У вікні налаштування вибрати тип вихідного сигналу, що юстується:
 - Дискретний;
 - Аналоговий, напруга;
 - Аналоговий, струм.

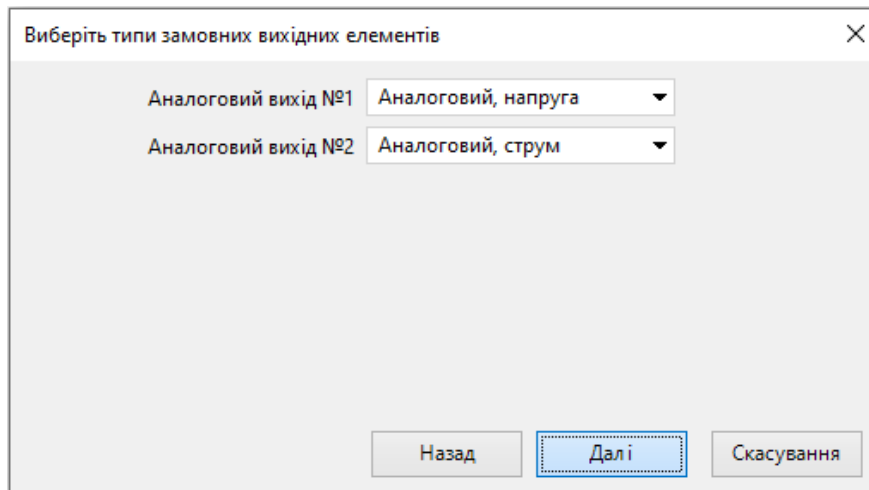


Рисунок 7.14

3. Виміряти величину поданого на вихід сигналу. Номер виходу вказаний у правому верхньому кутку вікна. Ввести значення у полі введення та натиснути **Далі**.

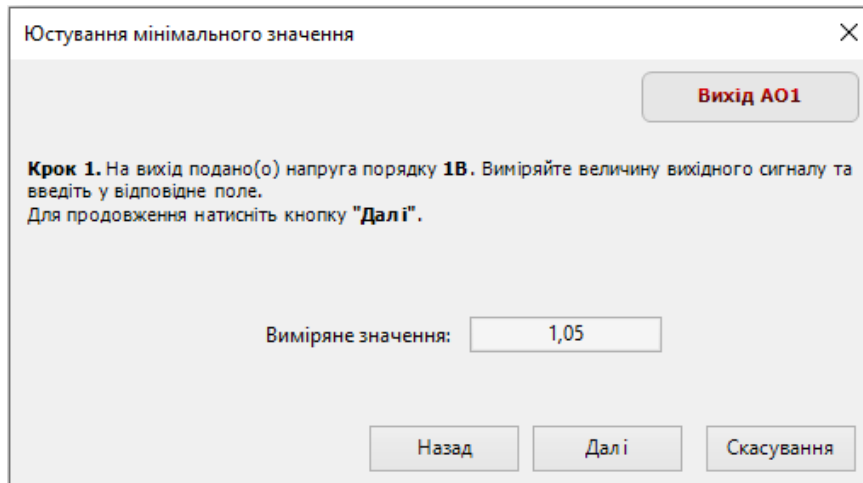


Рисунок 7.15

4. Аналогічно виміряти та ввести значення для всіх типів сигналів.
5. Після юстування з'явиться повідомлення про успішне завершення.

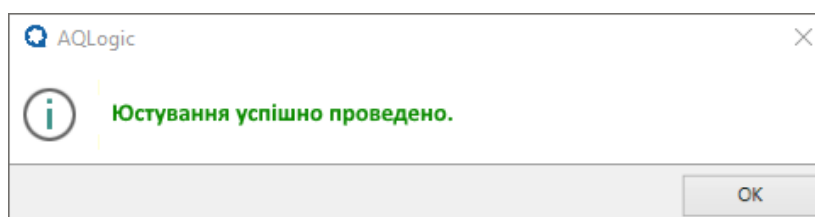


Рисунок 7.16

8 Зміна цільової платформи

Функція **Зміна цільової платформи** призначена для перенесення проекту в іншу модифікацію пристрою.

Для зміни цільової платформи слід:

1. Відкрити проект, який слід перенести в іншу модифікацію пристрою.
2. Вибрати в меню **Файл** → **Зміна цільової платформи**. AQLogic запропонує зберегти проект перед зміною платформи. Далі з'явиться меню зі списком доступних для зміни платформи модифікацій:

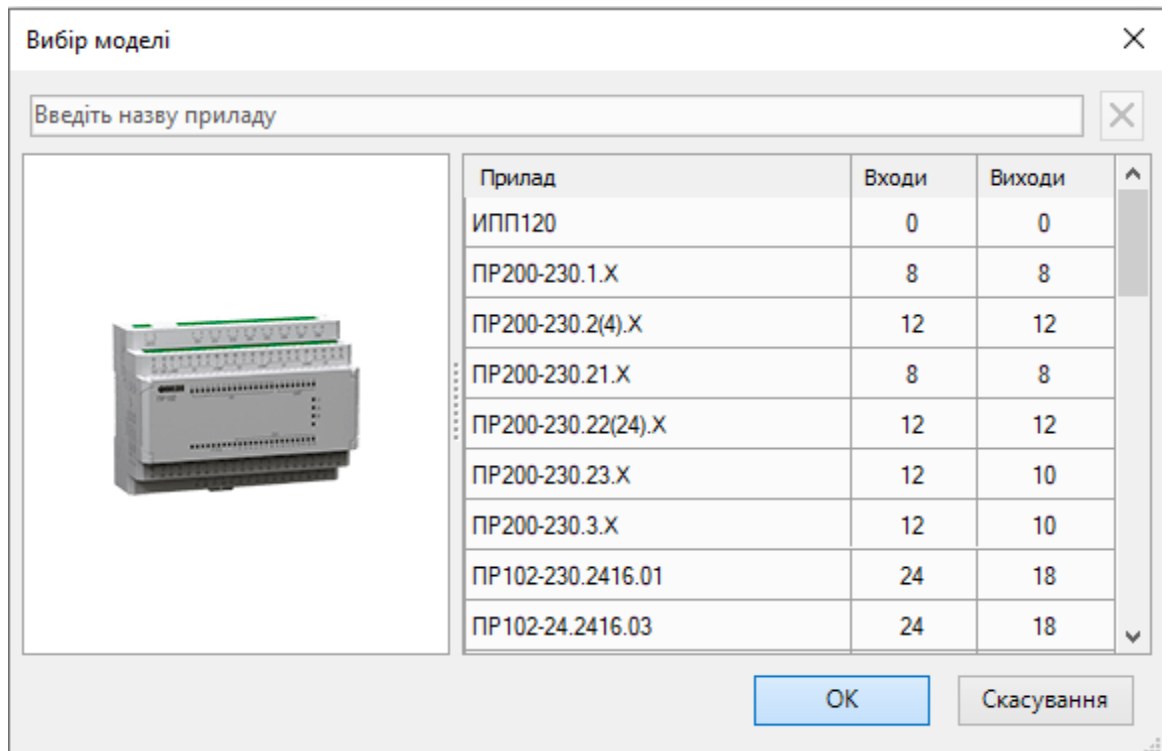


Рисунок 8.1

3. Вибрати пристрій, на який буде проведено зміну платформи. Натиснути кнопку **OK**.
4. Перевірити та відновити розірвані зв'язки, якщо вони є. Роботу програми можна перевірити у [Симуляторі 3.5](#). Зберегти змінений проект.

Зміна цільової платформи відбувається за такими правилами:

- полотно автоматично підлаштовується під зміну кількості входів/виходів;
- налаштоване користувачем розташування входів/виходів залишається. Нові входи та виходи додаються після вже існуючих входів та виходів вихідного проекту;
- зв'язки входів/виходів, у яких змінився тип даних, розриваються;
- якщо кількість входів/виходів збільшується, але тип даних вихідного набору входів/виходів не змінюється, зв'язки залишаються;
- якщо кількість входів/виходів зменшується порівняно з вихідною, зв'язки видалених входів/виходів розриваються;
- якщо у вихідній платформі були додані модулі розширення, то вони переносяться на нову платформу, для них зв'язки залишаються;
- налаштування аналогових входів/виходів переносяться (за наявності аналогових виходів на пристрої, на який йде перехід);
- інтерфейси зв'язку переносяться без змін;
- усі екрани з налаштуваннями переносяться з однієї цільової платформи до іншої;
- усі змінні переносяться з однієї цільової платформи до іншої.

9 Поєднання клавiш

Поєднання клавiш	Дiя
Меню/Файл	
Ctrl + N	Створити новий проект
Ctrl + O	Відкрити iснуючий проект
Ctrl + Alt + S	Зберегти відкритий проект під iншим iм'ям
Ctrl + S	Зберегти відкритий проект
Ctrl + P	Друк
Ctrl + Shift + C	Відкрити менеджер компонентiв
Меню/Вид	
Ctrl + Z	Скасувати останню змiну
Ctrl + Y	Повернути (вiдновити) скасовану дiю
Меню/Прилад	
Ctrl + F7	Записати програму у пристрiй
Ctrl + Shift + V	Відкрити таблицю змiнних
Ctrl + Shift + S	Відкрити налаштування пристрою
Меню/Сервiс	
Shift + F5	Перейти в режим симуляцiї
F6	Запустити симуляцiю
F7	Зупинити симуляцiю
F8	Пауза симуляцiї
F10	Крок на один цикл
Ctrl + F5	Перейти в режим налагодження
Меню/Допомога	
F1	Виклик довідки
Панель вставки	
Ctrl + Shift + F	Створити функцiю на ST
Ctrl + Shift + M	Створити макрос
Ctrl + M	Створити макрос iз видiленої дiлянки
Панель Мiсця використання	
F5	Оновити мiсця використання функцiї на ST
Клавiшi роботи з елементами	
Ctrl + C	Копiювати видiлений елемент(и) у буфер обмiну
Ctrl + V	Вставити з буфера обмiну
Delete	Видалення вибраного елемента
Клавiшi змiни розмiру елемента	
Ctrl + →	Збiльшення ширини видiленого елемента
Ctrl + ←	Зменшення ширини видiленого елемента
Ctrl + ↓	Зменшення висоти видiленого елемента
Ctrl + ↑	Збiльшення висоти видiленого елемента
Масштабування полотна	
Ctrl + Колесо мишi	При обертаннi колеса мишi вiд себе масштаб полотна збiльшується. Якщо «на себе», то зменшується
Ctrl + «+»	Збiльшення масштабу

Поєднання клавiш	Дiя
Ctrl + «←»	Зменшення масштабу
Перехiд мiж вкладками	
Tab + →	Перехiд мiж вкладками
Tab + ←	

10 Приклади проектів

У цьому розділі на прикладах пояснено процес створення програм в AQLogic .

- Включення світла з автоматичним відключенням 10.1;
- Автоматичне керування електродвигуном мішалки 10.2;
- Виведення значення змінної на екран 10.3;
- Задання уставок таймерів з екрану 10.4;
- Підключення ПР200 до OwenCloud через шлюз ПМ210 за протоколом Modbus RTU 10.5;
- Пакування/розпакування біт/цілих чисел 10.6;
- Обробка бітових змінних за допомогою маски 10.7.

10.1 Включення світла з автоматичним відключенням

У підсобних приміщеннях та коридорах електричне світло потрібне обмежений час. Після включення освітлення його нерідко забувають відключити, що призводить до зайвої витрати електроенергії.

Завдання — світло повинне включатися на заданий інтервал часу, наприклад, біля вхідних дверей у квартиру, за таким алгоритмом:

1. Перед вхідними дверима встановлюються датчик освітлення (**F1**) та кнопка включення світла **ТАЙМЕР (SB1)**.
2. Якщо недостатньо природного освітлення, то світильник повинен включатись короткочасним натисканням кнопки **ТАЙМЕР** на одну хвилину – достатньо, щоб знайти замкову щілину та ключем відчинити двері.
3. Якщо утримувати кнопку **ТАЙМЕР** протягом 2 секунд, світильник повинен включатися на три хвилини незалежно від зовнішнього освітлення – цей режим може знадобитися під час прибирання коридору.
4. Світильник повинен керуватися командами від зовнішніх керуючих пристроїв або за допомогою вмикача **СВІТЛО (SA1)**, незалежно від зовнішнього освітлення. Цей режим зручний під час прийому гостей або для подальшої автоматизації квартири в рамках системи «розумний дім».
5. Світильник повинен включатися лише у вечірній та нічний час.

Вибір програмованого пристрою

Для реалізації цього завдання у керуючого пристрою задіюються вбудований годинник реального часу, три входи та один вихід. Цей функціонал забезпечують пристрої із серії ПР110, що мають останню літеру «Ч» у позначенні.

Для реалізації системи на основі програмованого реле ПР110-24.8Д.4Р-Ч можна скористатися схемою на рисунку нижче.

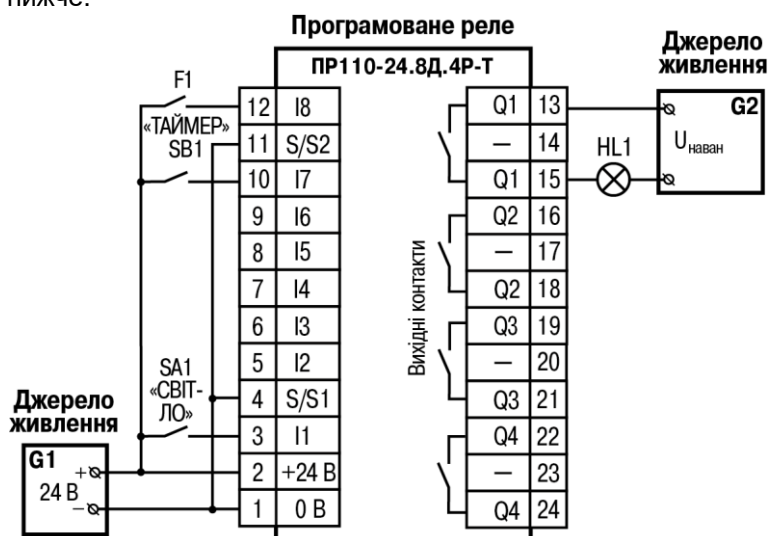


Рисунок 10.1

- **SA1** — перемикач OFF-ON;
- **SB1** — кнопка без фіксації OFF-ON;
- **HL1** — світлова лампа.

Створення програми

Електрична схема для керуючої програми може мати вигляд:

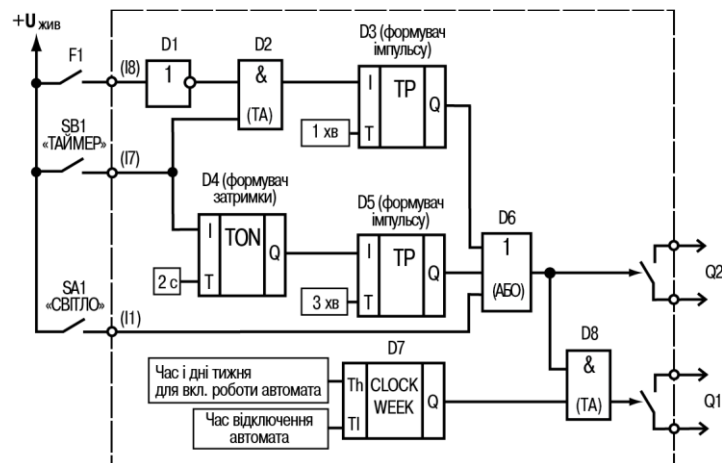


Рисунок 10.2

Вихід **Q2** використовується як контрольний для перевірки функціонування логічної частини (елементів **D1–D6**). Вихід **Q1** є основним і може включитися лише у задані таймером **D7** інтервали часу у разі дотримання логічних умов, що забезпечуються елементами **D1–D6**.

Опис роботи схеми:

1. Якщо короткочасно (менше 2 с) натиснути кнопку **ТАЙМЕР (SB1)**, логічна «1» надходить на вхід логічного елемента **I (D2)**. Оскільки на другому вході **D2** також є логічна «1», на виході **D2** з'явиться логічна «1», яка запустить формувач імпульсу (**D3**) тривалістю 1 хв. Цей імпульс через елемент АБО (**D6**) надійде на вихід **Q2**.
2. Контакти датчика **F1** замикають ланцюг тільки у разі наявності природного освітлення – тоді логічна «1» на вході **D1** перетворюється на логічний «0» на виході і надходить на вхід логічного елемента **I (D2)**, що блокує його роботу. У цьому випадку натискання кнопки **ТАЙМЕР** не включить вихід **Q2**.
3. Якщо утримувати кнопку **ТАЙМЕР** більше 2 секунд, на виході блоку **D4** з'явиться логічна «1», яка запустить формувач імпульсу (**D5**) тривалістю 3 хвилини. Цей імпульс через елемент АБО (**D6**) надійде на вихід **Q2** (інтервал дії імпульсу не залежить від стану датчика **F1**).
4. Якщо тумблер **СВІТЛО (SA1)** включений, то логічна «1» через елемент АБО (**D6**) надійде на вихід **Q2** (на час включення тумблера **СВІТЛО SA1**).
5. Якщо на другий вхід логічного елемента **I (D8)** надходить логічна «1» (від таймера реального часу **D7**) разом із сигналом логічної «1» з виходу елемента АБО (**D6**), на виході **Q1** з'явиться логічна «1» (вихідне реле **Q1** включиться).

Полотно проекту в AQLogic буде мати вигляд:

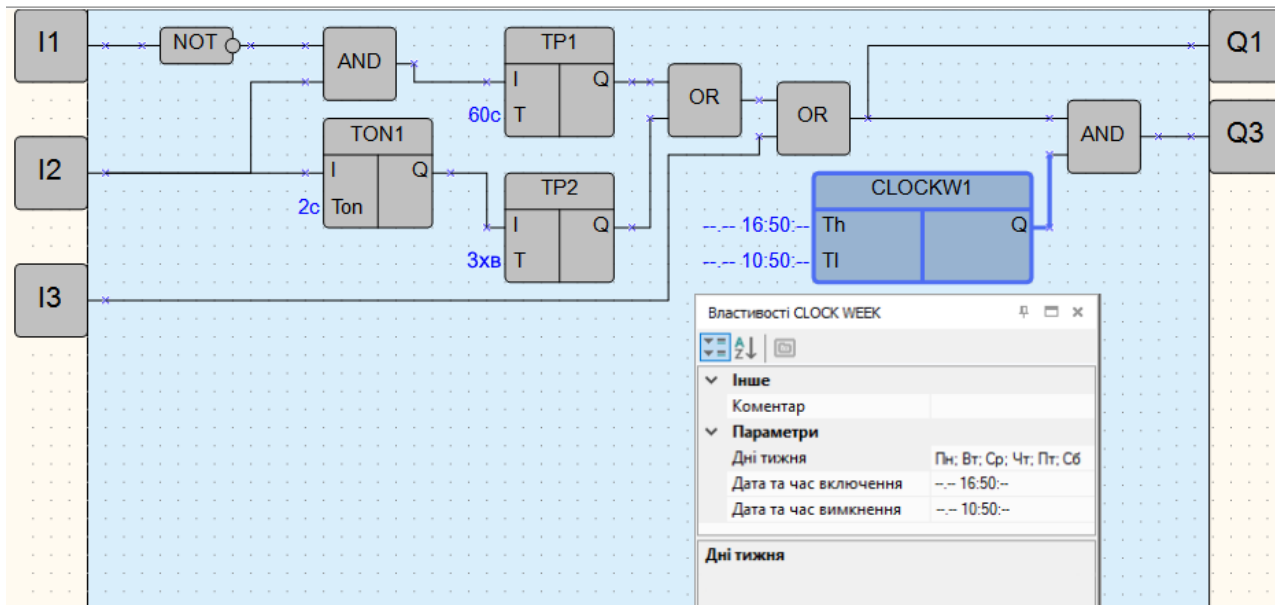


Рисунок 10.3

10.2 Автоматичне керування електродвигуном мішалки

Для виробництва продуктів харчування часто потрібно перемішувати компоненти (наприклад, молоко або вершки на молочній фермі) протягом певного часу.

Завдання забезпечити роботу установки для перемішування за таким алгоритмом:

1. У системі повинні бути два режими роботи: **Автоматичний** та **Ручний**, що переключаються тумблером **РЕЖИМ (SA1)**.
2. В автоматичному режимі електродвигун повинен включатися та відключатися натисканням кнопки **ПУСК (SB1)** через задані інтервали часу (15 с – включений, 10 с – відключений). Установка має відключатися через інтервал у 5 хвилин або за допомогою кнопки **СТОП (SB2)**.
3. У ручному режимі електродвигун повинен керуватися кнопками (без часових інтервалів відключення).
4. У разі перевантаження електродвигуна (на якому встановлюється відповідний датчик – **F1**) установка повинна автоматично відключатися з індикацією режиму **Несправність** лампою **HL1** та звуковим уривчастим сигналом (інтервал повторення звукового сигналу – 3 с).
5. Звуковий сигнал повинен відключатися кнопкою **СКИД (SB3)**.
6. За допомогою кнопки **КОНТРОЛЬ (SB4)** має перевірятися справність елементів сигналізації – робота лампи та звукового сигналу.

Вибір програмованого пристрою

Для реалізації цього завдання керуючий пристрій повинен мати шість входів (за кількістю керуючих сигналів) і три виходи, що забезпечує будь-яка модель пристрою із серії ПР110.

Для реалізації системи на основі програмованого реле ПР110-24.8Д.4Р можна скористатися схемою:

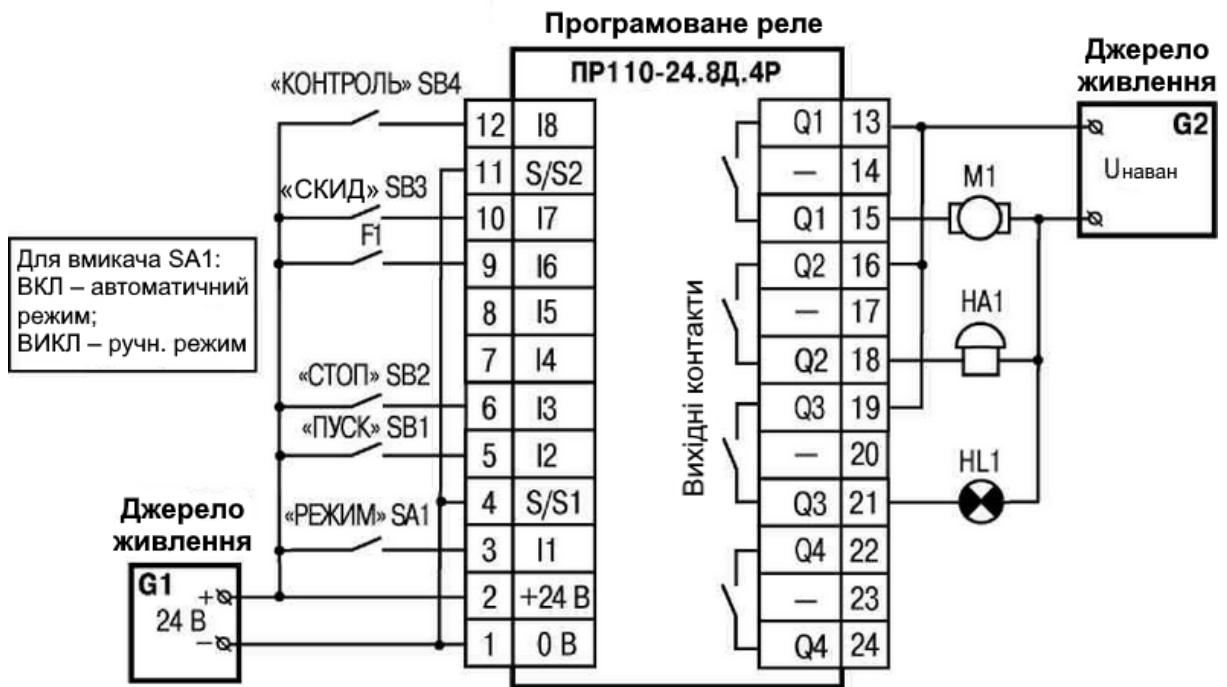


Рисунок 10.4

Електрична схема підключення: SA1 – перемикач OFF-ON, SB1–SB4 – кнопки без фіксації OFF-ON, M1 – електродвигун, HA1 – дзвоник, HL1 – індикаторна лампа.

Роботу програми пристрою можна подати у вигляді ланцюгів схеми, наведеної нижче (виходи Q1–Q3 відповідають двом контактам клемника програмованого реле ПР110-24.8Д.4Р).

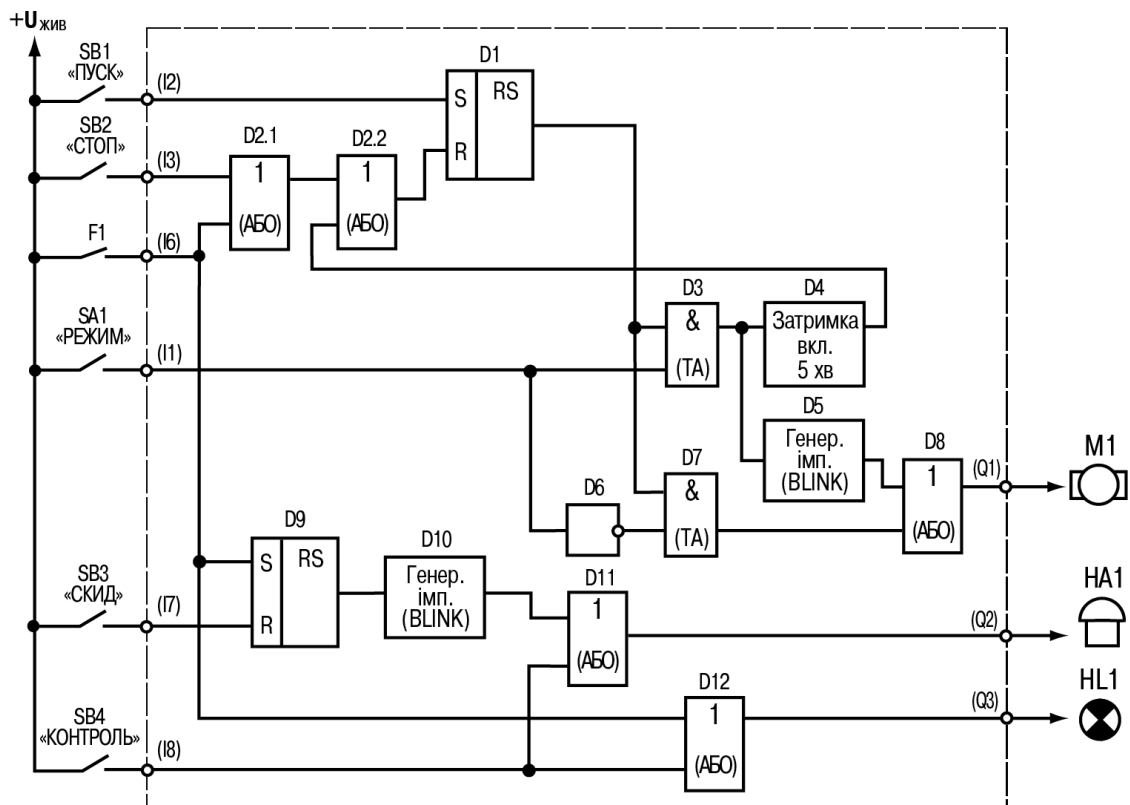


Рисунок 10.5

Опис роботи схеми

Ланцюг входу I2 (включення установки): у разі натискання кнопки ПУСК (SB1) на вході S (D1) з'являється логічна «1» – на виході RS-тригера D1 встановиться логічна «1». Цей сигнал надходить далі залежно від стану вмикача SA1:

- якщо контакти SA1 (РЕЖИМ – Ручний) розімкнуті, логічна «1» проходить через елементи D8, D9 і надходить на вихід Q1 (контактами вихідного реле включиться електродвигун M1);

- якщо контакти **SA1 (РЕЖИМ – Автоматичний)** замкнуті, логічна «1» проходить тільки через елемент **D3** для запуску роботи блоків **D4, D5**.

Ланцюг входу I3 (відключення установки): при натисканні кнопки **СТОП (SB2)** або спрацюванні датчика **F1** на вході R (**D1**) з'являється логічна «1» – на виході RS-тригера (**D1**) встановиться логічний «0» (включення виходу **Q1** блокується).

Ланцюг входу I1 (формування інтервалів роботи електродвигуна): якщо тригер **D1** включено і контакти вмикача **SA1 (РЕЖИМ – Автоматичний)** замкнуті:

- сигнал логічної «1» від **SA1** проходить через елемент **D3** і надходить на **D4** (формував імпульсу із затримкою 5 хв). Цей імпульс, проходячи через елемент АБО (**D2**), надійде на вхід R тригера **D1** і установка відключиться;
- сигнал логічної «1» з виходу **D3** надходить на **D5** – генератор імпульсів з параметрами: логічна «1» – 15 с, логічний «0» – 10 с. Ці імпульси з виходу генератора проходять через елемент АБО (**D8**) і надходять на вихід **Q1** для керування роботою електродвигуна **M1**;
- Якщо контакти вмикача **SA1 (РЕЖИМ – Ручний)** розімкнуті, то логічний елемент **D3** заблоковано, і сигнал з виходу тригера **D1** через елемент **D6** одразу надходить на вихід **Q1**, тобто у цьому режимі стан виходу тригера керує роботою електродвигуна **M1**.

Ланцюг входу I6 (включення звукової сигналізації): у разі спрацювання датчика **F1** на вході S (**D9**) з'являється логічна «1» – на виході RS-тригера **D9** встановиться логічна «1», яка включає генератор **D10** з параметрами: логічна «1» – 3 с, логічний «0» – 3 с. Ці імпульси проходять через елемент АБО (**D12**) і надходять на вихід **Q2** для керування роботою дзвоника **HA1**.

Ланцюг входу I7 (відключення звукової сигналізації): у разі натискання кнопки **СКИД (SB3)** на вході R (**D9**) з'являється логічна «1» – на виході RS-тригера встановиться логічний «0», і генератор **D10** відключається.

Ланцюг входу I8 (контроль роботи лампи і звукової сигналізації): при натиснутій кнопці **КОНТРОЛЬ (SB4)** логічна «1» через елемент АБО (**D11**) надходить на включення виходу **Q3** – лампа **HL1** включиться. Якщо контакти датчика **F1** замкнуті, то вихід **Q3** включиться. При натиснутій кнопці **КОНТРОЛЬ (SB4)** логічна «1» через елемент АБО (**D12**) надходить на включення виходу **Q2**, і дзвінок **HA1** включиться.

Полотно проекту в AQLogic буде мати вигляд:

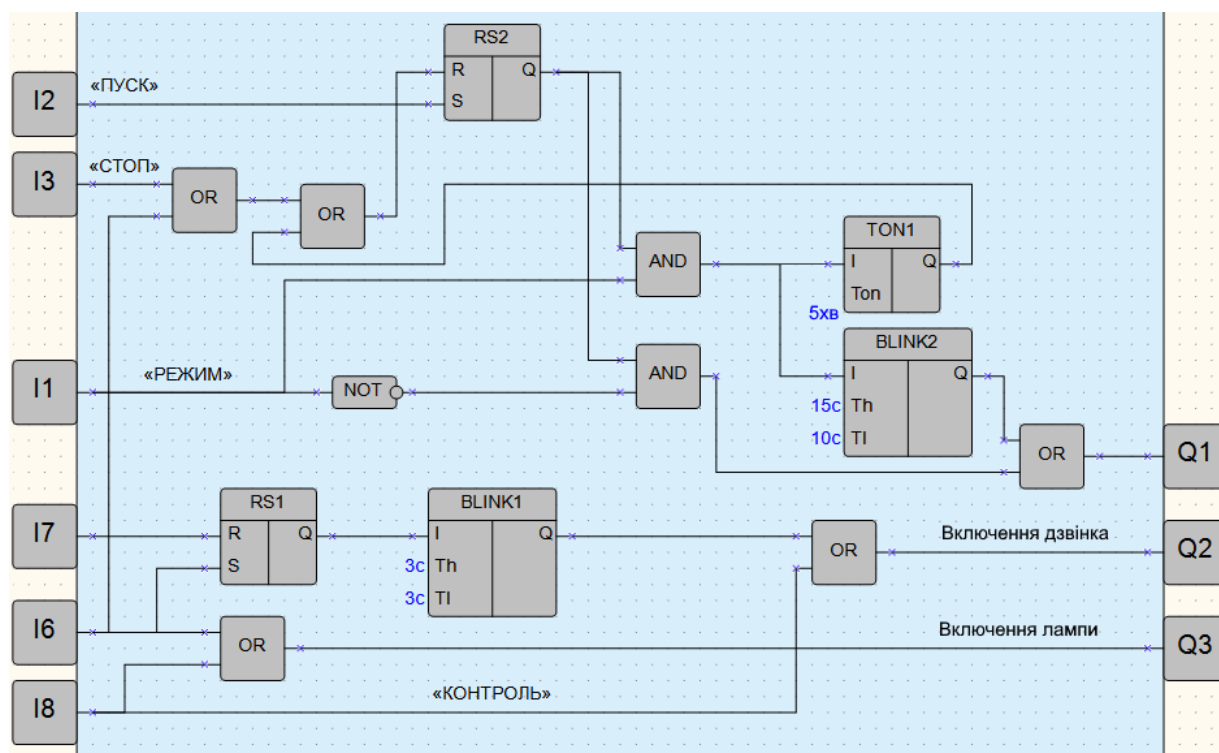


Рисунок 10.6



ПРИМІТКА

- Інші вільні два входи і один вихід можна використовувати для введення додаткових функцій. Наприклад, переключати чотири різні інтервали тривалості автоматичної роботи електродвигуна або змінювати інші параметри установки.
- Технологічний цикл роботи установки можна зробити автоматичним, якщо схему доповнити інкрементним лічильником (СТ), вихідним сигналом якого можна включати тригер **D1**.

10.3 Виведення значення змінної на екран

Завдання – на пристрої з підключеним до нього за інтерфейсом зв'язку модулем вводу необхідно зчитати значення на входах 1 та 2 модуля і вивести їх на дисплей пристрою.

Для розробки програми слід:

1. Створити новий проект в **AQLogic**.
2. Підключити шаблон мережевого пристрою до проекту. Для цього слід запустити **менеджер компонентів** та відсортувати компоненти за категорією **Шаблони мережевих приладів**. Зі згенерованого списку вибрати **MB110-8A** і натисканням на кнопку **Звантажити у проект** скачати шаблон.

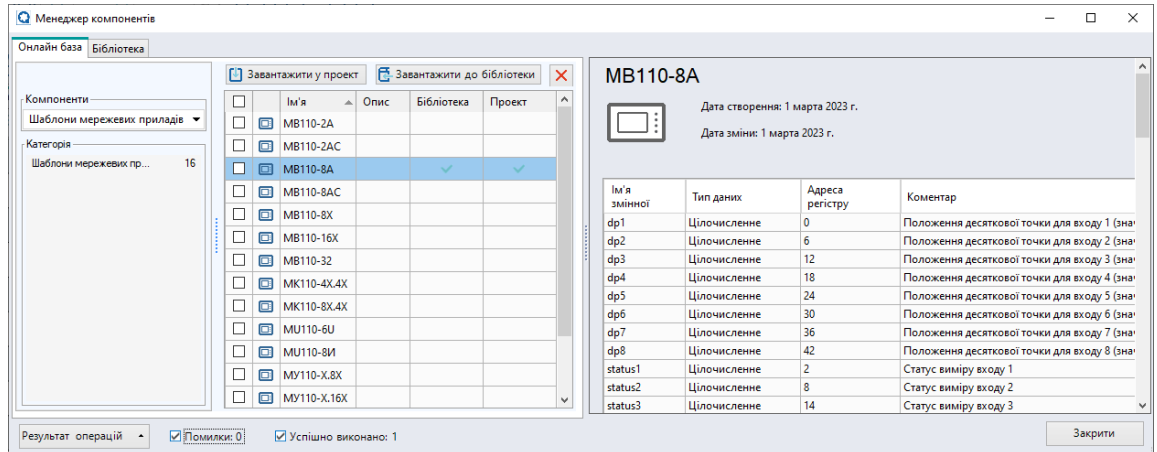


Рисунок 10.7

3. У вікні **налаштувань пристрою** підключити цей шаблон.
4. Після додавання шаблону мережевого пристрою у **таблиці змінних** слід уточнити імена та типи змінних, які планується виводити на екран (**Izm_fl1** і **Izm_fl2**). Це температури, виміряні модулем і передані в пристрій по протоколу Modbus. Змінні мають тип із рухомою комою.

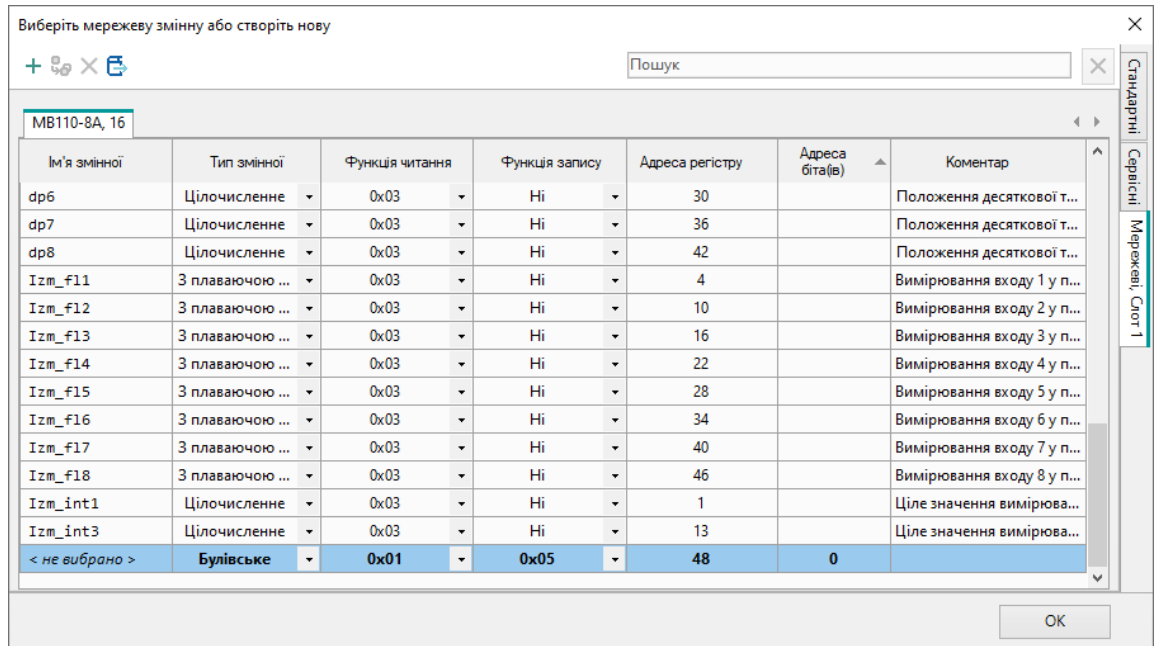


Рисунок 10.8

5. Перейти в **менеджер екранів**.
6. Подвійним натисканням на екрані відкрити редактор екранів.

7. З бібліотеки компонентів додати елемент **Ввід/вивід (int/float)**.
8. Відкрити властивості елемента та вказати тип змінної – **Дійсний**. Потім у рядку **Змінна** натиснути кнопку «...».

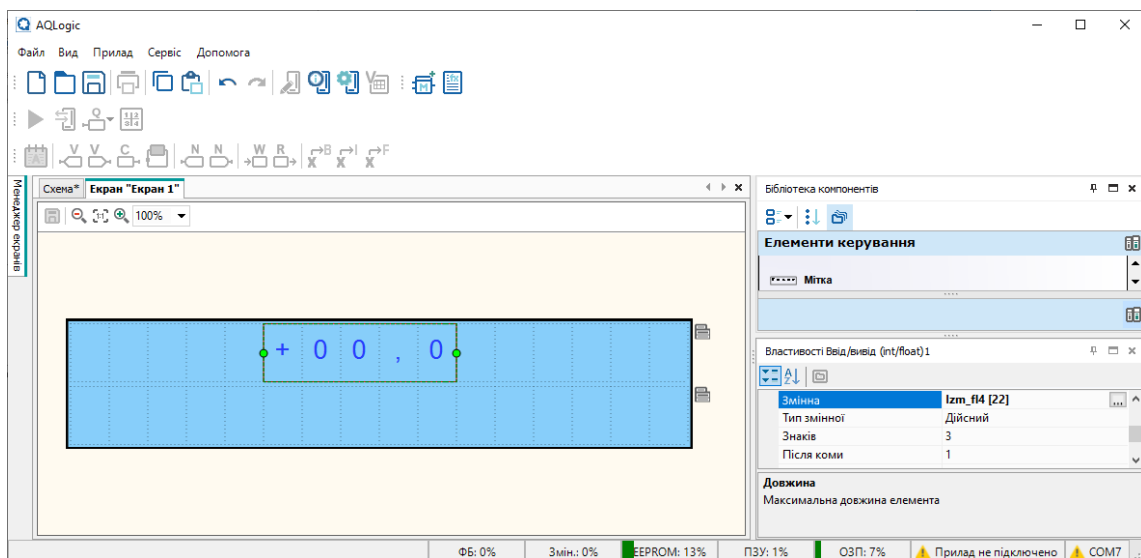


Рисунок 10.9

9. У таблиці змінних вибрати змінну **Izm1** на вкладці **RS-485, Слот1**.
10. У полі **Знаків** у властивостях елемента вказати необхідну кількість знаків для виведення значення вибраної змінної. У полі **Після коми** вказати кількість дробових розрядів. У полі **Довжина** вказати загальну кількість знаків елемента. У полі **Текст до** набрати **T1** і поставити пробіл для відокремлення значення від тексту.
11. У полі **Редагуємо** поставити **Ні**, оскільки змінювати виміряні на модулі значення температури не можна.
12. На цьому налаштування одного елемента **Ввід/вивід (int/float)** завершено. Для спрощення подальшої роботи слід скопіювати елемент з усіма налаштуваннями, натиснувши на нього ПКМ та вибравши в контекстному меню **Копіювати** (або скористатися поєднанням клавіш **Ctrl + C**).

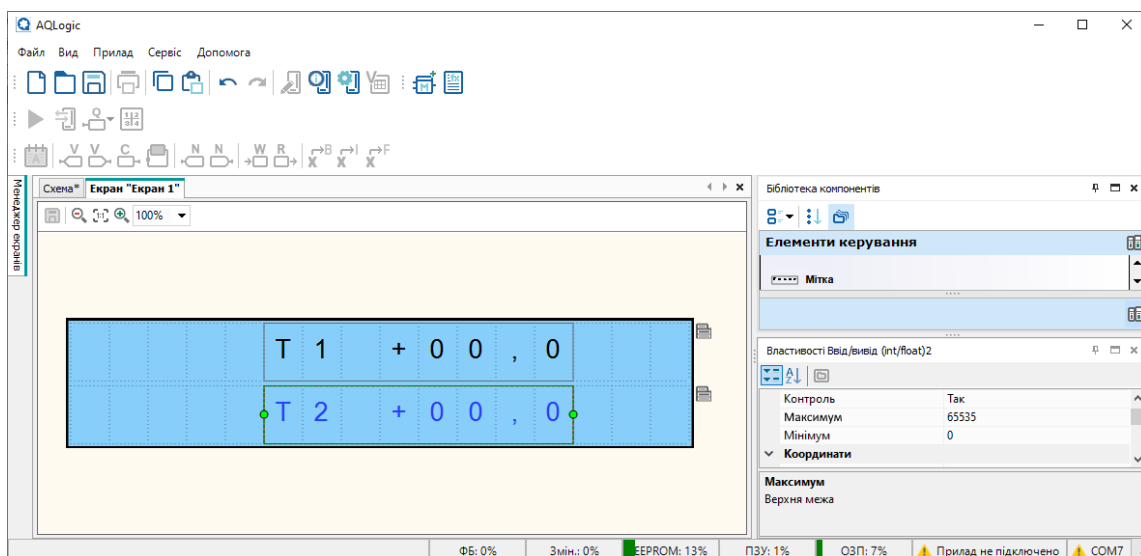


Рисунок 10.10

13. Вставити скопійований елемент поверх існуючого, натиснувши ПКМ на екран і вибравши в контекстному меню **Вставити** (або скористатися клавішами **Ctrl + V**).
14. На вставленому елементі змінити змінну на **Izm_f12** та замінити текстовий блок на **T2**. Програма готова до завантаження у пристрій.

10.4 Задання уставок таймерів з екрану

Завдання – необхідно керувати затримками ФБ Blink (генератор прямокутних імпульсів) та TP (імпульс включення заданої тривалості) з дисплея пристрою. Передбачити можливість перетворення затримок на мілісекунди, секунди або години. Додатково необхідно відображати задану уставку ФБ на дисплеї пристрою.

Для реалізації завдання було вибрано ПР200.

Полотно проекту з елементами наведено на рисунку нижче.

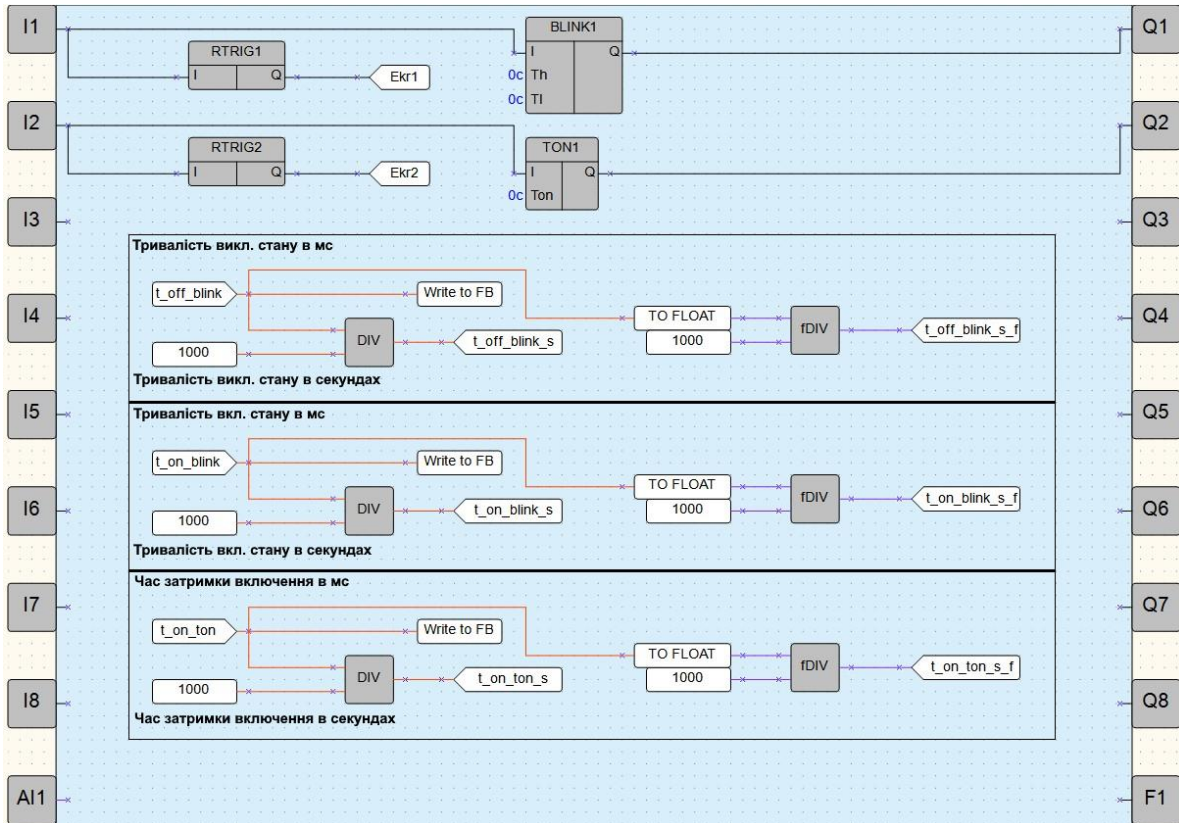


Рисунок 10.11 – Схема алгоритму

ФБ **Blink** (генератор прямокутних імпульсів) і **TP** (імпульс включення заданої тривалості) приймають як уставки числа в цілочисельному форматі з дискретністю 1 мс.

Щоб перерахувати уставки під потрібні одиниці часу, слід:

- для відображення значень у секундах, слід розділити введене значення на 1000;
- для відображення уставок у хвилинах, слід розділити введене значення на $(60 \cdot 1000)$;
- для відображення уставок у годинах, слід розділити введене значення на $(60 \cdot 60 \cdot 1000)$.

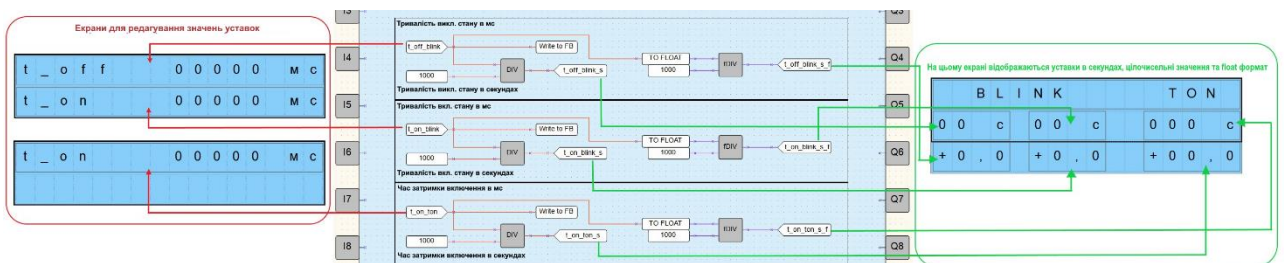


Рисунок 10.12

Якщо потрібно виводити значення уставок з десятковими знаками, слід перевести число в тип з рухомою комою і поділити на відповідний коефіцієнт.

Приклад

Для переведення 2500 мілісекунд на секунди слід: $2500 \text{ мс} \div 1000 = 2,5 \text{ с}$.

10.5 Підключення ПР200 до OwenCloud через шлюз ПМ210 за протоколом Modbus RTU

Вважатимемо, що ПМ210 вже підключений до ПР200 і налаштований для роботи.

Для підключення слід:

1. Створити проект для ПР200 в AQLogic .
2. У вікні [параметрів пристрою 5](#) задати параметри, як на рисунку нижче.

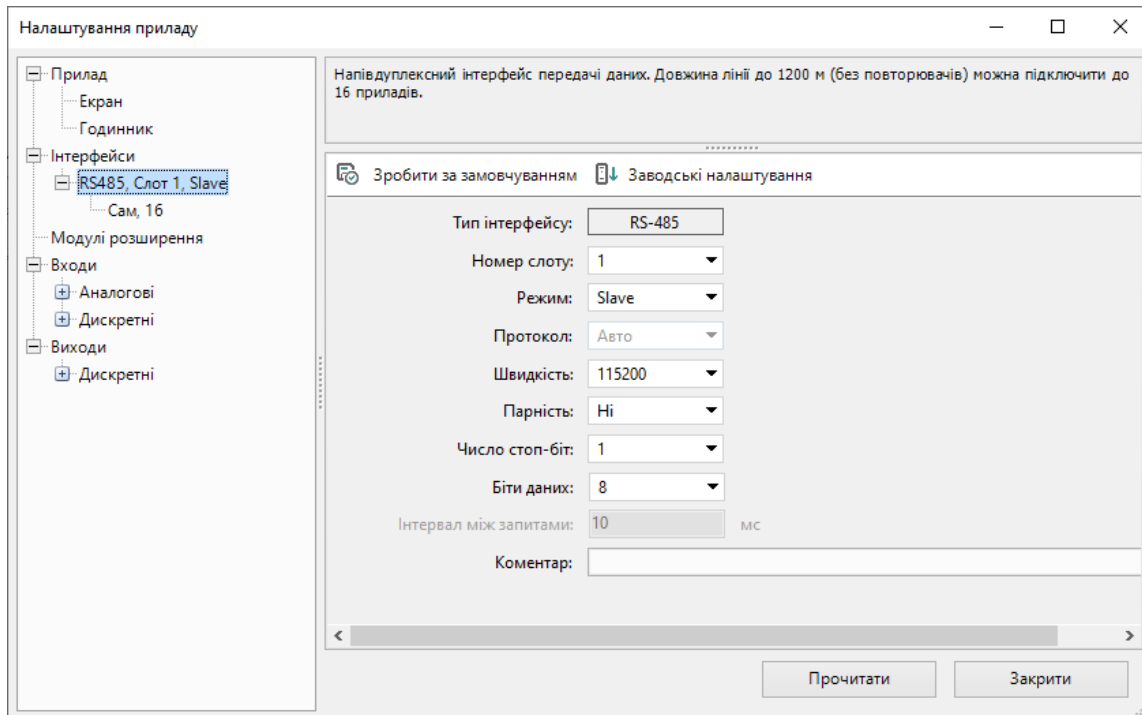


Рисунок 10.13

3. У налаштуваннях слота вказати адресу 1 і додати мережеві змінні, як на рисунку нижче.

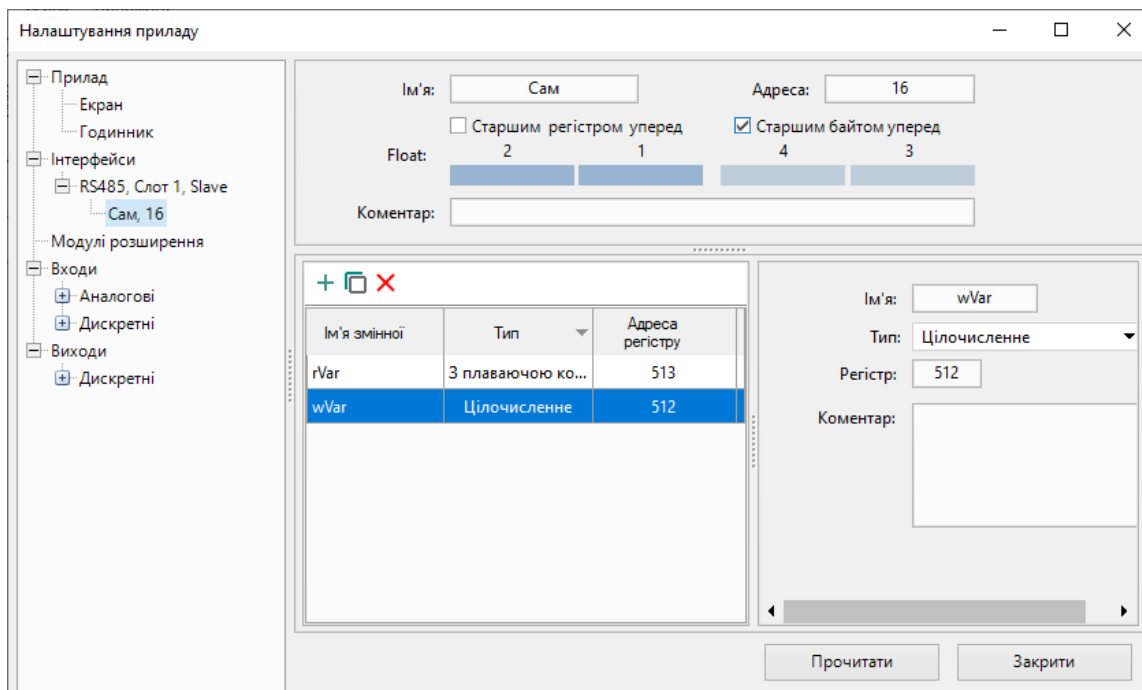


Рисунок 10.14

4. **Слід звернути увагу**, що змінна з рухомою комою (**rVar**) займає два регістри в пам'яті пристрою (у нашому випадку – 512 та 513).
5. Створити екран у **менеджері екранів 2.8** та додати на нього елементи **Ввід-вивід Int** і **Ввід-вивід Float**. Прив'язати до них змінні **wVar** та **rVar**. У налаштуваннях елементів для параметра **Редаговано** слід встановити значення **Так**, щоб мати можливість змінювати їх з дисплея ПР200.

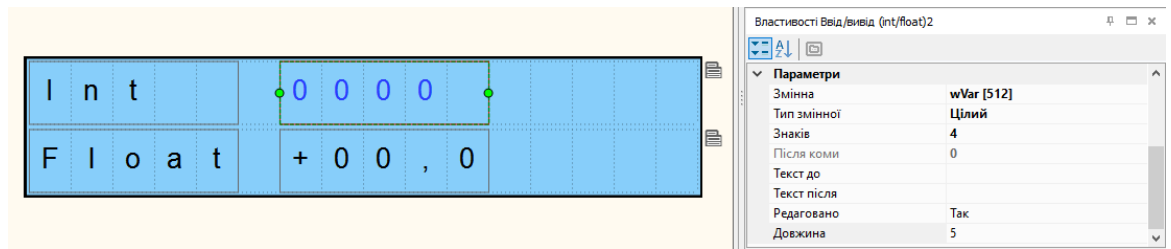


Рисунок 10.15

6. Завантажити проект в ПР200 (**Прилад** → **Записати програму в прилад**).
7. Зайти на головну сторінку **OwenCloud**. Якщо ще не зареєстровані у сервісі, необхідно пройти процедуру реєстрації.
8. Перейти на вкладку **Администрирование**, відкрити вкладку **Приборы** і натиснути кнопку

Добавить прибор

(+ Добавить прибор).

9. Вказати такі налаштування:
 - **Идентификатор** – ввести **IMEI** мережевого шлюзу (вказаний на корпусі шлюзу);
 - **Тип прибора** – вибрати тип **Произвольное устройство Modbus**;
 - **Адрес в сети** – вказати адресу **1**;
 - **Название прибора** – ввести назву пристрою;
 - **Часовой пояс** – вказати часовий пояс, у якому знаходиться пристрій.

Добавление прибора

✕

Идентификатор*

Введите IMEI сетевого шлюза

Тип прибора*

Произвольное устройство Modbus

Адрес в сети*

1

Заводской номер

Целое, не более 17 знаков

Название прибора*

ПР200

Часовой пояс*

GMT+3:00

Время на странице прибора будет смещаться в зависимости от часового пояса.

Отменить

Добавить

Рисунок 10.16

10. Натиснути кнопку **Добавить**.
11. Натиснути на назву пристрою, щоб перейти до його налаштування:

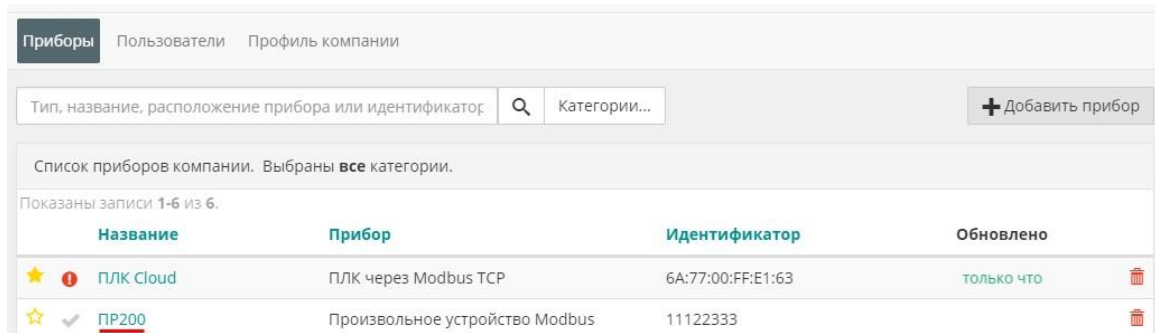


Рисунок 10.17

12. На вкладці **Общие/Общие настройки** вказати швидкість опитування та налаштування СОМ-порту пристрою відповідно до налаштувань в AQLogic . Натиснути кнопку **Сохранить** , щоб застосувати нові налаштування. У разі потреби можна змінити інші налаштування (наприклад, період опитування).

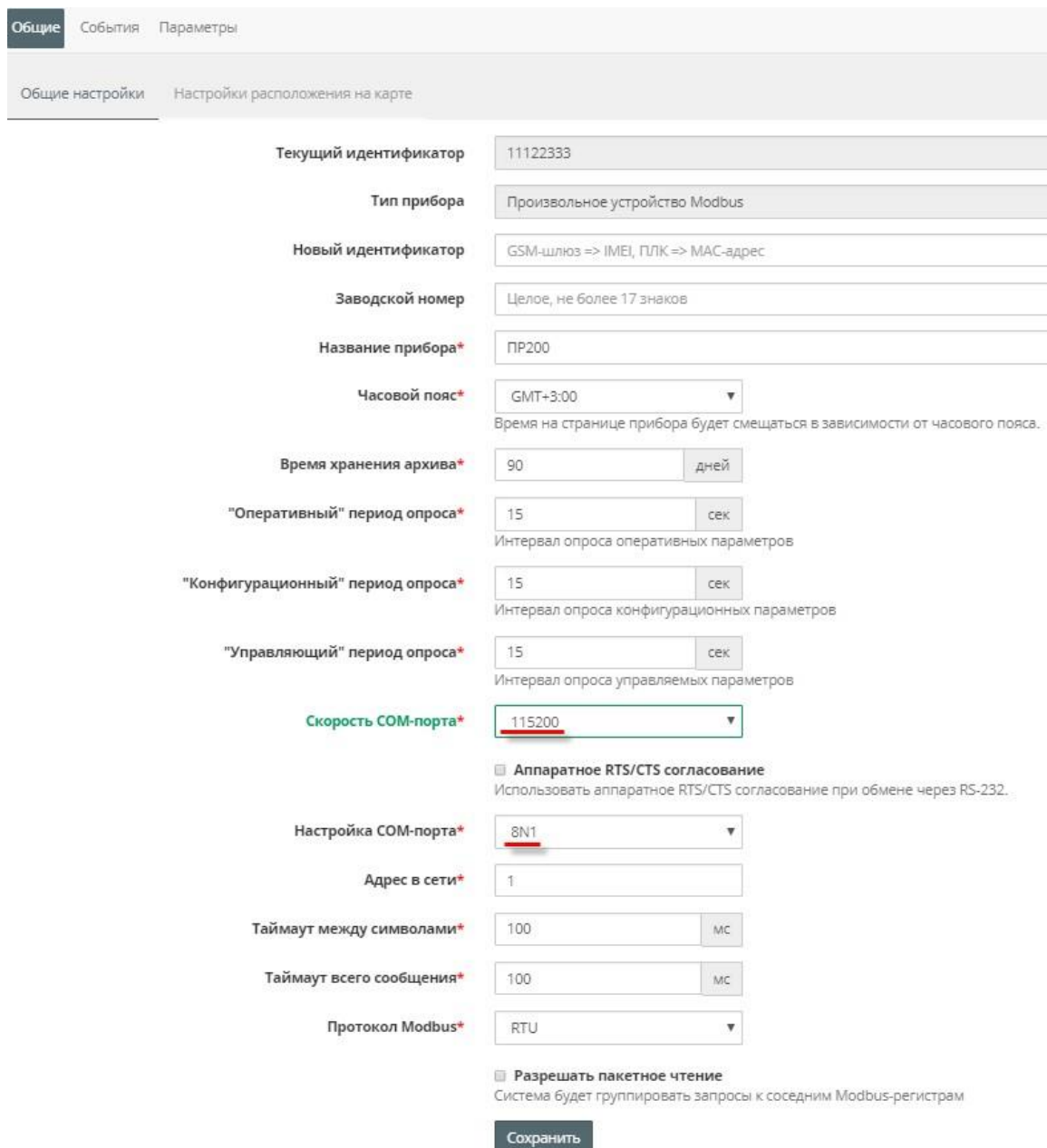


Рисунок 10.18

13. На вкладці **Параметри/Настройки параметров Modbus** додати параметри вручну. **Слід звернути увагу**, що адреси реєстрів вказуються в шістнадцятковій системі – тому значення, що використовуються, відрізняються від тих, які наведені в AQLogic – 512 (DEC) = 200 (HEX).





Управление прибором: ПР200

Общие События **Параметры**

Настройка параметров Modbus Общая настройка параметров

Добавить + Удалить все параметры 🗑️ Импортировать... ▾

Всего 2 записи.

Код параметра	Функция чтения	Функция записи	Адрес регистра	Формат хранения	Порядок хранения байт	Отображаемая единица измерения	Название
rVar	03	16	201	float	Младший байт сзади	попе: 3 цифр после точки, без единиц	FLOAT  
wVar	03	06	200	uint16	Младший байт сзади	попе: 3 цифр после точки, без единиц	INT  



Экспорт в JSON 

Рисунок 10.19

14. Перейти до перегляду значень параметрів пристрою. Змінити значення змінних з дисплея ПР200 та спостерігати відповідні зміни в **OwenCloud**.
15. У разі потреби зміни значень перейти на вкладку **Запись параметров**.

ПР200-24.4.2.0 ☆  обновлено только что ⚙️

Параметры Таблицы Графики Лог активных событий Запись параметров

Код параметра	Параметр	Значение
rVar	rVar	5.500
wVar	wVar	9.000

Рисунок 10.20

10.6 Пакування/розпакування біт/цілих чисел

Завдання – необхідно зчитувати стан входів пристрою та виводити на екран стан вибраного входу.

Нижче на рисунку наведено полотно проекту з використанням макросів з [менеджера компонентів](#)

[3.10](#) та налаштування властивостей елементів виведення на екран.

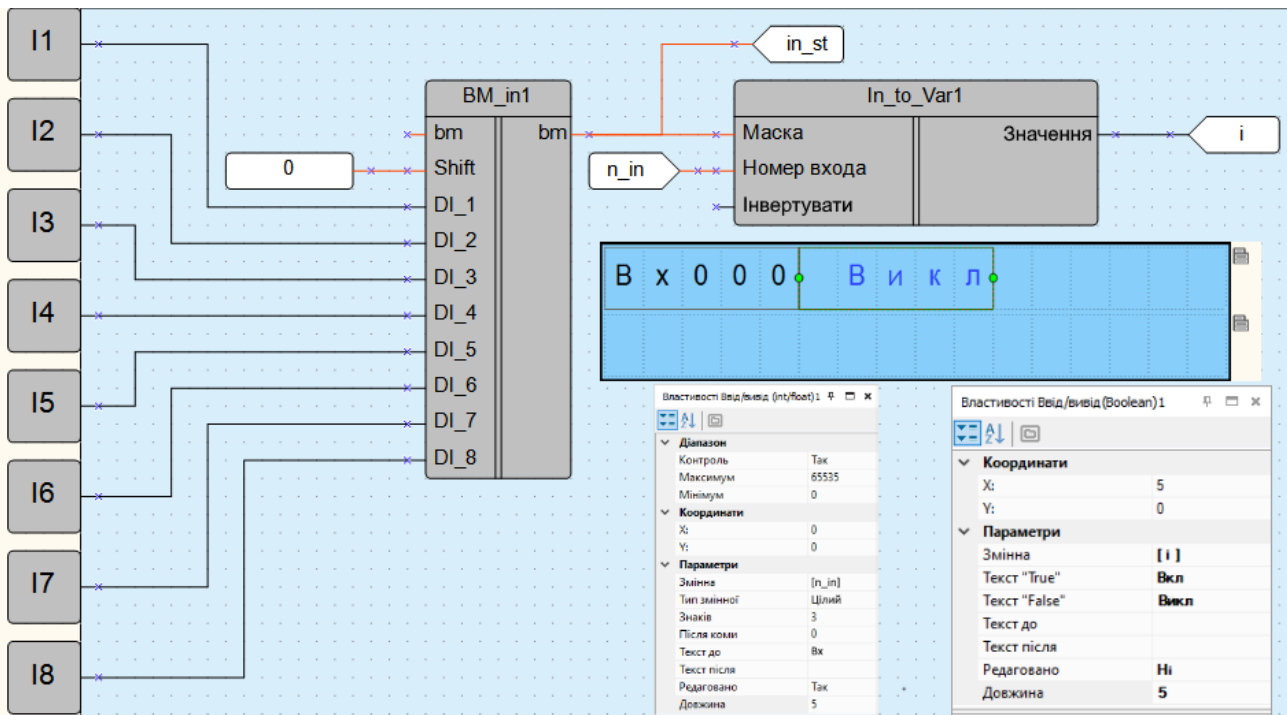


Рисунок 10.21

10.7 Обробка бітових змінних за допомогою маски

Завдання – за сигналом виходи включаються або відключаються. Якщо реалізується додаткова умова – спрацює вхід 1, то виходи 2, 4, 6 та 8 повинні приймати інвертований стан відносно поточного.

Для задання станів виходів слід використовувати макроси з [менеджера компонентів 3.10](#) — **BOOL_INT** та **INT_BOOL**.

Додаткову умову цього прикладу можна реалізувати за допомогою побітного винятку XOR. Як маску можна використати цілочисельну константу — **170** (10101010 у двійковому вигляді).

Полотно проекту в режимі симуляції наведено нижче.

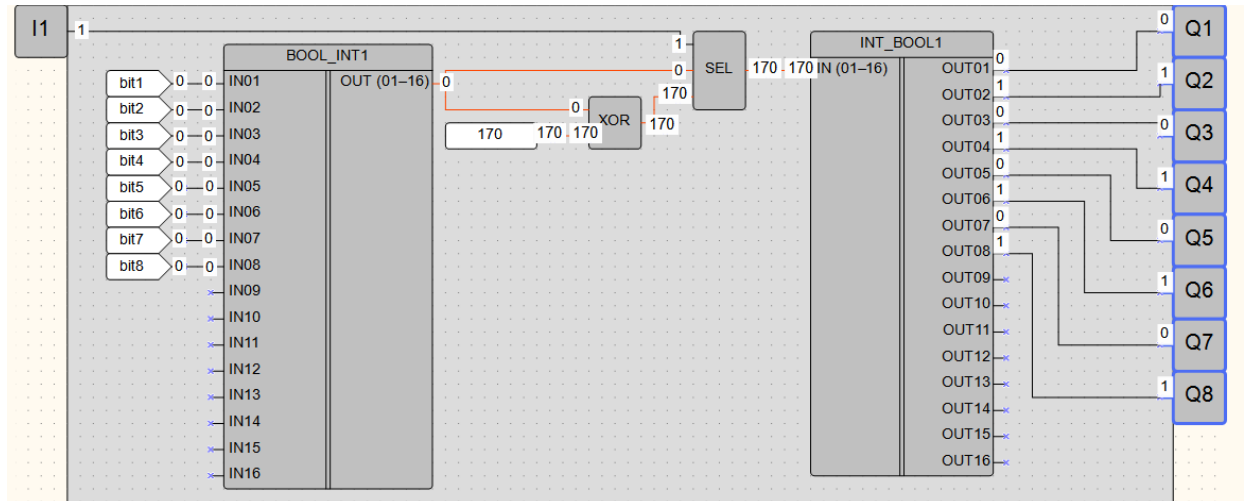


Рисунок 10.22

11 Мова програмування ST

- [Загальні відомості 11.1](#)
- [Типи даних 11.2](#)
- [Конструкції мови 11.3](#)

11.1 Загальні відомості

ST (Structured Text) – це текстова мова високого рівня загального призначення, за синтаксисом схожа з мовою Pascal. Може використовуватись у програмах, у тілі функції або функціонального блоку, а також для опису дії та переходу всередині елементів SFC.

Відповідно до IEC 61131-3 ключові слова мають бути введені у символах верхнього регістру. Пробіли та мітки табуляції не впливають на синтаксис, вони можуть бути використані скрізь.

Структура виразу: **[variable] := [value];**

Вирази складаються з операндів та операторів. Операндом може бути літерал, змінна, структурована змінна, компонент структурованої змінної, звернення до функції або пряма адреса.

11.2 Типи даних

Тип даних змінної визначає рід інформації, діапазон представлень та безліч допустимих операцій.



ПРИМІТКА

Будь-яку змінну можна використовувати лише після її оголошення. Надавати значення однієї змінної інший можна, тільки якщо вони одного типу. Інакше використовуються перетворювачі типів.



ПОПЕРЕДЖЕННЯ

Перетворення великих типів на дрібніші може призвести до втрати інформації.

Типи даних, підтримані в AQLogic, наведені в таблиці нижче:

Назва	Тип	Можливі значення	Розмір у байтах
BOOL	Логічний	FALSE, TRUE	1 біт
UDINT	Цілочисельний	0...4294967295	4 байти
REAL	Дійсний	-1,2×10 ⁻³⁸ ...3,4×10 ³⁸	4 байти

11.3 Конструкції мови

До конструкцій мови ST відносяться:

- [арифметичні операції 11.3.1;](#)
- [логічні \(побітові\) операції 11.3.2;](#)
- [операції порівняння 11.3.3;](#)
- [операція присвоєння 11.3.4;](#)
- [конструкція IF – ELSIF – ELSE 11.3.5;](#)
- [конструкція CASE 11.3.6;](#)
- [цикл FOR 11.3.7;](#)
- [цикл WHILE 11.3.8;](#)
- [цикл REPEAT UNTIL 11.3.9.](#)



ПРИМІТКА

При записі виразів допустимо використовувати змінні (локальні і глобальні) та константи.

11.3.1 Арифметичні операції



ПРИМІТКА

Чим вище пріоритет операції, тим раніше вона виконується

Символ	Операція	Пріоритет
+	додавання	1
-	віднімання	
*	множення	2
/	ділення	
mod	залишок від цілочисельного ділення	
()	взяття у дужки	3*

**ПРИМІТКА**

* При записі арифметичних виразів допустиме використання дужок для вказівки порядку обчислень. Дія, взята у дужки, має найвищий пріоритет при обчисленнях.

11.3.2 Логічні операції

Позначення	Операція
OR	Логічне (побітове) додавання
AND	Логічне (побітове) множення
XOR	Логічне (побітове) «виключне АБО»
NOT	Логічне (побітове) заперечення

11.3.3 Операції порівняння

Символ	Операція
=	Порівняння на рівність
<>	Порівняння на нерівність
>	Більше
>=	Більше або дорівнює
<	Менше
<=	Менше або дорівнює

11.3.4 Присвоєння

Для позначення присвоєння використовується парний знак «:=». У правій та лівій частині виразу мають бути операнди одного типу (автоматичного приведення типів не передбачено). У лівій частині виразу (приймаюча сторона) може бути використана лише змінна. Права частина може містити вираз чи константу.

11.3.5 Конструкція IF – ELSIF – ELSE

**ПРИМІТКА**

Для опису деяких конструкцій мови використані фігурні та квадратні дужки:

- вираз у фігурних дужках може використовуватися нуль або більше разів поспіль;
- вираз у квадратних дужках не обов'язковий для використання.

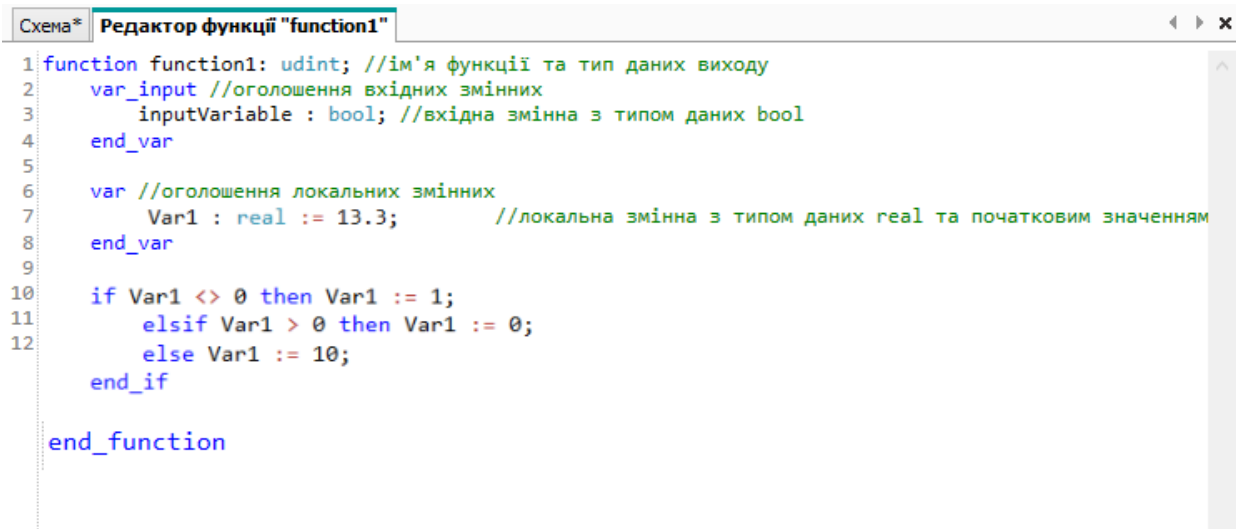
Конструкція IF – ELSIF – ELSE має такий формат:

```

if <boolean expression> then <statement list>;
  [elseif <boolean expression> then <statement list>];
  [else <statement list>];
end_if;

```

Рисунок 11.1



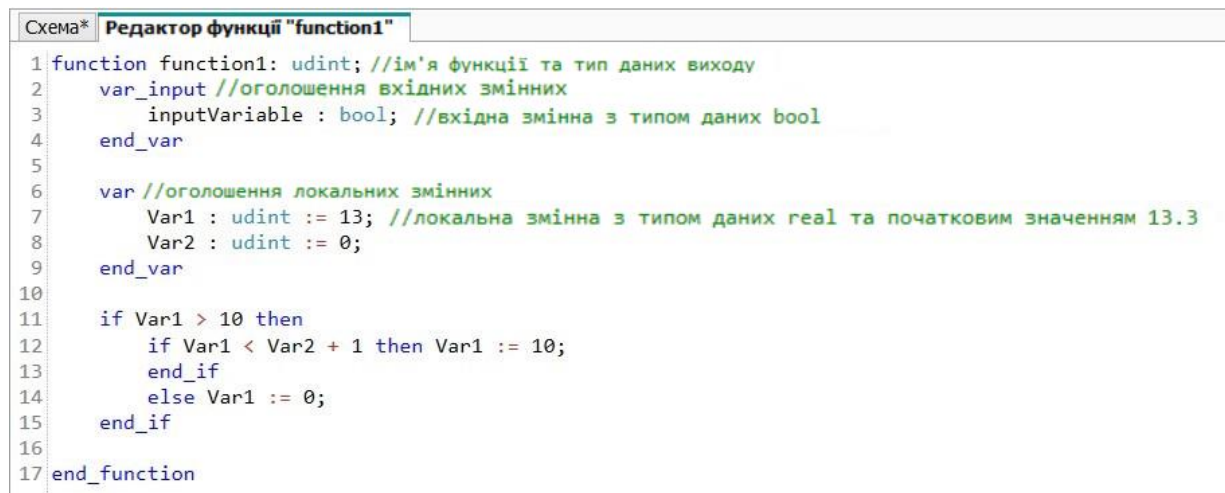
```

Схема* Редактор функції "function1"
1 function function1: udint; //ім'я функції та тип даних виходу
2 var_input //оголошення вхідних змінних
3   inputVariable : bool; //вхідна змінна з типом даних bool
4 end_var
5
6 var //оголошення локальних змінних
7   Var1 : real := 13.3; //локальна змінна з типом даних real та початковим значенням
8 end_var
9
10  if Var1 <> 0 then Var1 := 1;
11     elseif Var1 > 0 then Var1 := 0;
12     else Var1 := 10;
13 end_if
14
15 end_function

```

Рисунок 11.2 – Приклад реалізації конструкції

Конструкція припускає вкладеність, тобто всередині одного IF може бути ще один і т. д. Наприклад:



```

Схема* Редактор функції "function1"
1 function function1: udint; //ім'я функції та тип даних виходу
2 var_input //оголошення вхідних змінних
3   inputVariable : bool; //вхідна змінна з типом даних bool
4 end_var
5
6 var //оголошення локальних змінних
7   Var1 : udint := 13; //локальна змінна з типом даних real та початковим значенням 13.3
8   Var2 : udint := 0;
9 end_var
10
11  if Var1 > 10 then
12     if Var1 < Var2 + 1 then Var1 := 10;
13     end_if
14     else Var1 := 0;
15     end_if
16
17 end_function

```

Рисунок 11.3

11.3.6 Конструкція CASE



ПРИМІТКА

Для опису деяких конструкцій мови використані фігурні та квадратні дужки:

- вираз у фігурних дужках може використовуватися нуль або більше разів поспіль;
- вираз у квадратних дужках не обов'язковий для використання.

Конструкція CASE слугує для організації вибору діапазону значень. Формат конструкції:

```

case <Expression> of
  CASE_ELEMENT
  {CASE_ELEMENT}
  [else <Statement List>]
end_case;

```

Рисунок 11.4

CASE_ELEMENT – це список значень, перерахованих через кому. Елементом списку може бути ціле число чи діапазон цілих чисел (*початкове значення .. кінцеве значення*).

Значення **EXPRESSION** може бути тільки цілим.

```

1 function function1: udint; //ім'я функції та тип даних виходу
2   var_input //оголошення вхідних змінних
3     inputVariable : bool; //вхідна змінна з типом даних bool
4   end_var
5
6   var
7     Var1 : udint :=1;
8     Var2 : udint :=0;
9   end_var
10
11   case Var1 of
12     1:
13       Var1 := Var1 * 2;
14     2..5:
15       Var1 := Var1 * 5;
16       Var2 := Var1 * 1;
17     6, 9..20:
18       Var1 := Var1 - 1;
19     else
20       Var1 := 20;
21       Var2 := 0;
22   end_case
23
24 end_function

```

Рисунок 11.5

**ПРИМІТКА**

Рядок 14 містить діапазон значень. Якщо значення **Var1** належить числовому відрізку [2, 5], то будуть виконані рядки 15 та 16.

У рядку 17 використано список значень. Рядок 18 виконається, якщо значення **Var1** дорівнюватиме 6 або буде належати числовому відрізку [9, 20].

Рядки 20 і 21 будуть виконані у тому випадку, якщо **Var1** < 1, або 6 < **Var1** < 9, або **Var1** > 20 (у цьому випадку спрацює пропозиція ELSE).

При заданні списку значень необхідно виконувати такі умови:

- набори значень усередині однієї конструкції CASE не повинні перетинатися;
- при вказівці діапазону значень початок діапазону має бути меншим від його кінця.

Дії, передбачені для обробки кожного з випадків CASE, можуть використовувати цикли, оператори [IF 12.3.5](#) і [CASE 12.3.6](#).

11.3.7 Цикл FOR**ПРИМІТКА**

Для опису деяких конструкцій мови використані фігурні та квадратні дужки:

- вираз у фігурних дужках може використовуватися нуль або більше разів поспіль;
- вираз у квадратних дужках не обов'язковий для використання.

Конструкція з FOR слугує для задання циклу з фіксованою кількістю ітерацій. Формат конструкції:

```

for <Control Variable> := <expression1> to <expression2> by <expression3> do
  <statement_list>;
end for

```

Рисунок 11.6

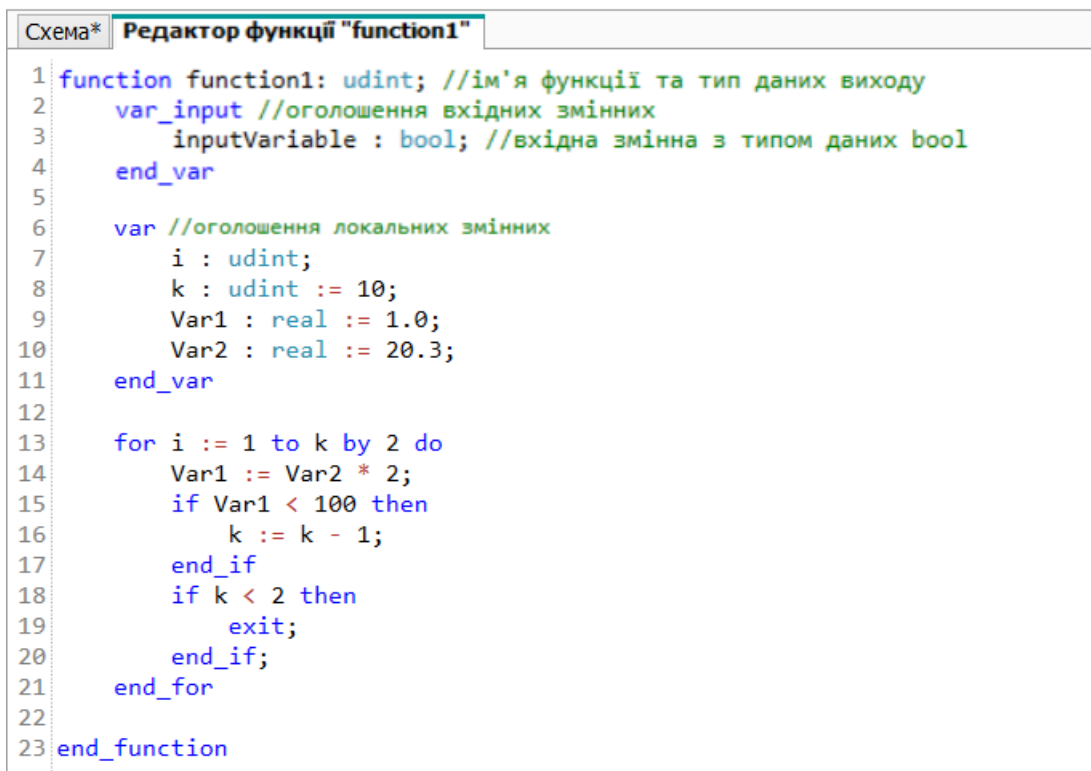
де:

- **Control Variable** – змінна циклу;
- **expression1** – початкове значення змінної циклу;
- **expression2** – кінцеве значення змінної циклу. Вихід із циклу відбудеться, якщо значення змінної циклу перевищить це значення;
- **expression3** – збільшення змінної циклу. Якщо оператор BY не вказаний, то збільшення дорівнює 1.

Змінна циклу, початкове і кінцеве значення та збільшення є цілочисельними.

Всередині циклу можуть бути використані інші оператори циклу й оператори [IF 12.3.5](#) та [CASE 12.3.6](#).

Для виходу з циклу можна використовувати оператор EXIT. Наприклад:



```

1 function function1: udint; //ім'я функції та тип даних виходу
2   var_input //оголошення вхідних змінних
3     inputVariable : bool; //вхідна змінна з типом даних bool
4   end_var
5
6   var //оголошення локальних змінних
7     i : udint;
8     k : udint := 10;
9     Var1 : real := 1.0;
10    Var2 : real := 20.3;
11  end_var
12
13  for i := 1 to k by 2 do
14    Var1 := Var2 * 2;
15    if Var1 < 100 then
16      k := k - 1;
17    end_if
18    if k < 2 then
19      exit;
20    end_if;
21  end_for
22
23 end_function

```

Рисунок 11.7



ПРИМІТКА

Початкове та кінцеве значення змінної циклу, а також значення збільшення обчислюються до входу в цикл. Зміна значень змінних, що входять до будь-якого з цих виразів, не приведе до зміни числа ітерацій.

11.3.8 Цикл WHILE



ПРИМІТКА

Для опису деяких конструкцій мови використані фігурні та квадратні дужки:

- вираз у фігурних дужках може використовуватися нуль або більше разів поспіль;
- вираз у квадратних дужках не обов'язковий для використання.

Конструкція WHILE слугує для визначення циклу з передумовою. Цикл буде виконуватися доти, доки вираз у реченні WHILE повертає TRUE. Формат конструкції:

```

while <Boolean Expression> do
  <statement lits>;
end_while

```

Рисунок 11.8

Значення <Boolean-Expression> перевіряється на кожній ітерації. Завершення циклу відбудеться, якщо <Boolean-Expression> поверне FALSE.

Всередині циклу можуть використовуватись інші цикли, оператори IF 12.3.5 та CASE 12.3.6. Для дострокового завершення циклу використовується оператор EXIT.

```

Схема* Редактор функції "function1"
1 function function1: udint; //ім'я функції та тип даних виходу
2   var_input //оголошення вхідних змінних
3     inputVariable : bool; //вхідна змінна з типом даних bool
4   end_var
5
6   var //оголошення локальних змінних
7     i : udint;
8     k : udint := 10;
9     Var1 : real := 1.0;
10    Var2 : real := 20.3;
11  end_var
12
13  while Var1 < 100.0 do
14    Var1 := Var2 * 2;
15    if Var1 < 100 then
16      k := k - 1;
17    end_if
18    if k < 2 then
19      exit;
20    end_if;
21  end_while
22
23 end_function

```

Рисунок 11.9

11.3.9 Цикл REPEAT UNTIL



ПРИМІТКА

Для опису деяких конструкцій мови використані фігурні та квадратні дужки:

- вираз у фігурних дужках може використовуватися нуль або більше разів поспіль;
- вираз у квадратних дужках не обов'язковий для використання.

Конструкція REPEAT UNTIL слугує для визначення циклу з постумовою. Завершення циклу відбудеться тоді, коли вираз у реченні UNTIL поверне FALSE. Формат конструкції:

```

repeat
  <statement lits>
until <Boolean Expression>
end_repeat

```

Рисунок 11.10

Всередині циклу можуть використовуватись інші цикли, оператори IF та CASE. Для дострокового завершення циклу використовується оператор EXIT.

```
Схема* Редактор функції "function1"
1 function function1: udint; //ім'я функції та тип даних виходу
2   var_input //оголошення вхідних змінних
3     inputVariable : bool; //вхідна змінна з типом даних bool
4   end_var
5
6   var
7     Var1 : real := 13.3;
8     Var2 : real := 3.3;
9   end_var
10
11  repeat
12    Var1 := Var1 + Var2;
13    Var2 := Var1 / 2;
14    if Var1 > 30 then
15      exit;
16    end_if;
17  until Var2 < 1
18  end_repeat
19
20 end_function
```

Рисунок 11.11



61153, м. Харків, вул. Гвардійців Широнинців, 3А
тел.: (057) 720-91-19; 0-800-21-01-96 (багатоканальний)
тех. підтримка: support@aqteck.com.ua
відділ продажу: sales@aqteck.com.ua
aqteck.com.ua
реєстр.: 2-УК-1117-1.2