

**Міністерство освіти і науки України**

**ЦЕНТРАЛЬНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ**

**Кафедра автоматизації виробничих процесів**

## **Методичні вказівки**

*до виконання лабораторних робіт з курсу*

**“Автоматизація технологічних процесів”**

*для студентів спеціальності: 141- Електроенергетика, електротехніка та  
електромеханіка*

Затверджено  
на засіданні кафедри  
Автоматизації виробничих процесів  
Протокол №1 від 30.08.2018

Кропивницький  
2018 р.

Автоматизація технологічних процесів / Методичні вказівки до виконання лабораторних робіт з курсу “Автоматизація технологічних процесів” для студентів спеціальності: 141- “Електроенергетика, електротехніка та електромеханіка”, / С.І. Осадчий, І.В. Волков, – Кропивницький: ЦНТУ, 2018. – 55с.

Укладачі: Осадчий С.І, д.т.н., проф.  
Волков І.В., викл.

Рецензент: Віхрова Л.Г., д.т.н., проф.

# Лабораторна робота №1.

## Дослідження системи управління насосною станцією на базі логічного контролера САУ-МП.

**Мета роботи:** дослідити роботу насосної станції на базі логічного контролера САУ-МП в ручному та автоматичному режимах. Отримати навички складання систем управління водопостачанням на базі логічного контролера. Навчитися програмувати контролер згідно із заданим алгоритмом та конфігурацією системи.

### Короткі теоретичні відомості

#### 1. ПРИЗНАЧЕННЯ

1.1. Прилад САУ МП призначений для створення систем автоматизації технологічних процесів, пов'язаних з контролем і підтримкою заданого рівня рідких або сипучих речовин в різного роду резервуарах, ємностях, контейнерах і т.п.

1.2. Контроль рівня здійснюється за допомогою чотирьох датчиків, що підключаються до входів приладу, які встановлюються користувачем в резервуарі на заданих умовах технологічного процесу позначках. Для візуального контролю за спрацюванням датчиків на лицьовій панелі приладу передбачені чотири світлодіодних індикатори, засвічення кожного з яких сигналізує про спрацювання відповідного датчика.

1.3. При застосуванні відповідного алгоритму прилад може бути застосований для управління групою насосів у системах гарячого і холодного водопостачання.

1.4. В якості входних датчиків можуть бути застосовані:

- кондуктометричні (що контролюють ступінь електропровідності середовища) зонди;
- активні датчики з вихідними ключами n-p-n)типу;
- механічні контактні пристрої;
- датчики наявності тиску типу ДЕМ.

1.5. Для управління технологічним обладнанням прилад оснащений трьома вбудованими електромагнітними реле.

Робота кожного реле може здійснюватися як в автоматичному режимі (за сигналами датчиків), так і по командам оператора (від вбудованих кнопок ручного управління на передній панелі приладу). Алгоритм роботи реле в автоматичному режимі визначається при замовленні приладу або записується користувачем з комп'ютера за допомогою специфічного кабелю.

## 2. ТЕХНІЧНІ ХАРАКТЕРИСТИКИ

Основні технічні характеристики приладу САУ МП наведено в табл.1,а зовнішній вигляд контролера на рис. 1.

Таблиця 1

Найменування	Значення
Номінальна напруга живлення приладу	220 В 50 Гц
Допустимі відхилення напруги живлення від номінального значення	+10 %...15 %
Потужність споживання, не більше	4 ВА
Кількість входів	4
Напруга вбудованого в прилад джерела живлення активних датчиків	$12 \pm 1,2$ В (50 мА макс.)
Кількість вихідних реле	3
Допустиме навантаження на контакти реле (при напрузі 220В і $\cos \varphi > 0,4$ )	8 А
Маса приладу, не більше	0,7 кг

**УВАГА!** Максимальна допустима напруга на вході - 5 В.



Рис.1. Зовнішній вигляд логічного контролера САУ-МП.

### 3.БУДОВА І РОБОТА ПРИЛАДУ

#### 3.1. Функціональна схема приладу

3.1.1. Функціональна схема пристрою САУ-МП представлена на рис. 2.

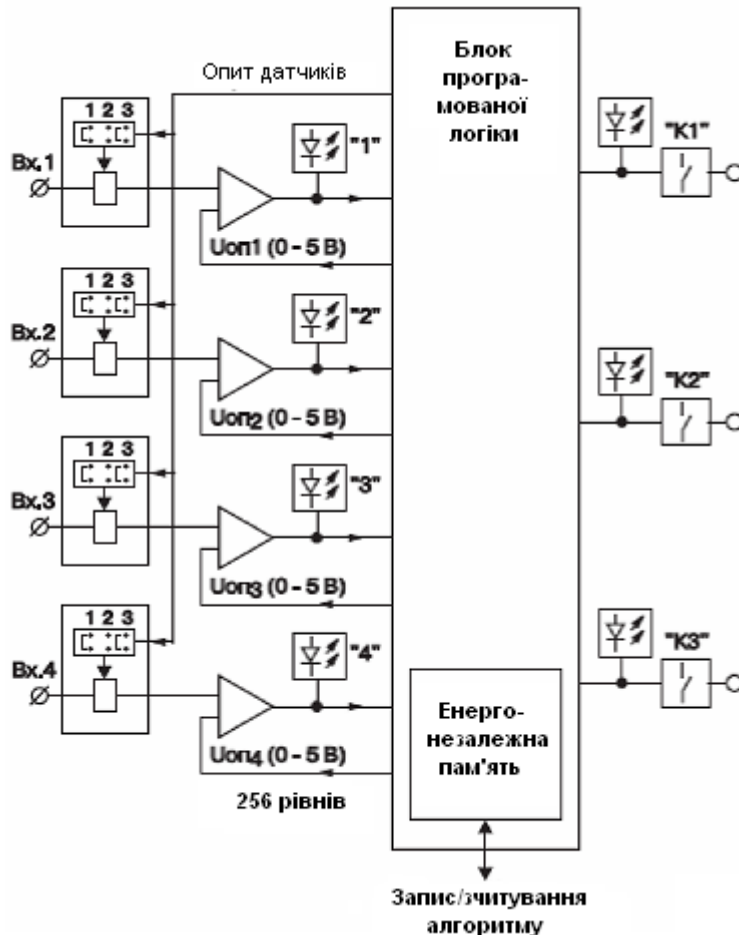


Рис. 2. Функціональна схема логічного контролера САУ-МП

До складу пристрою входять:

- *вхідні пристрої ВУ1...ВУ4*, призначені для прийому сигналів датчиків, порівняння величини прийнятого сигналу з заданим і видачі на блок програмованої логіки відповідного логічного рівня (0 або 1);
- *блок програмованої логіки*, призначений для формування сигналів управління вихідними реле за прийнятим від вхідних пристроїв сигналам згідно із записаним у його енергонезалежну пам'ять алгоритмом, а також для прийому та копіювання алгоритму з іншого пристрою або комп'ютера;
- *вихідні електромагнітні реле*, призначені для управління виконавчими механізмами;
- *світлодіодні індикатори*, які служать для відображення інформації про стан датчиків, вихідних реле та режимах роботи пристрою.

### 3.2. Конструкція стенду.

3.2.1. Прилад САУ-МП встановлений у верхньому правому куті стенду (див рис 3.).

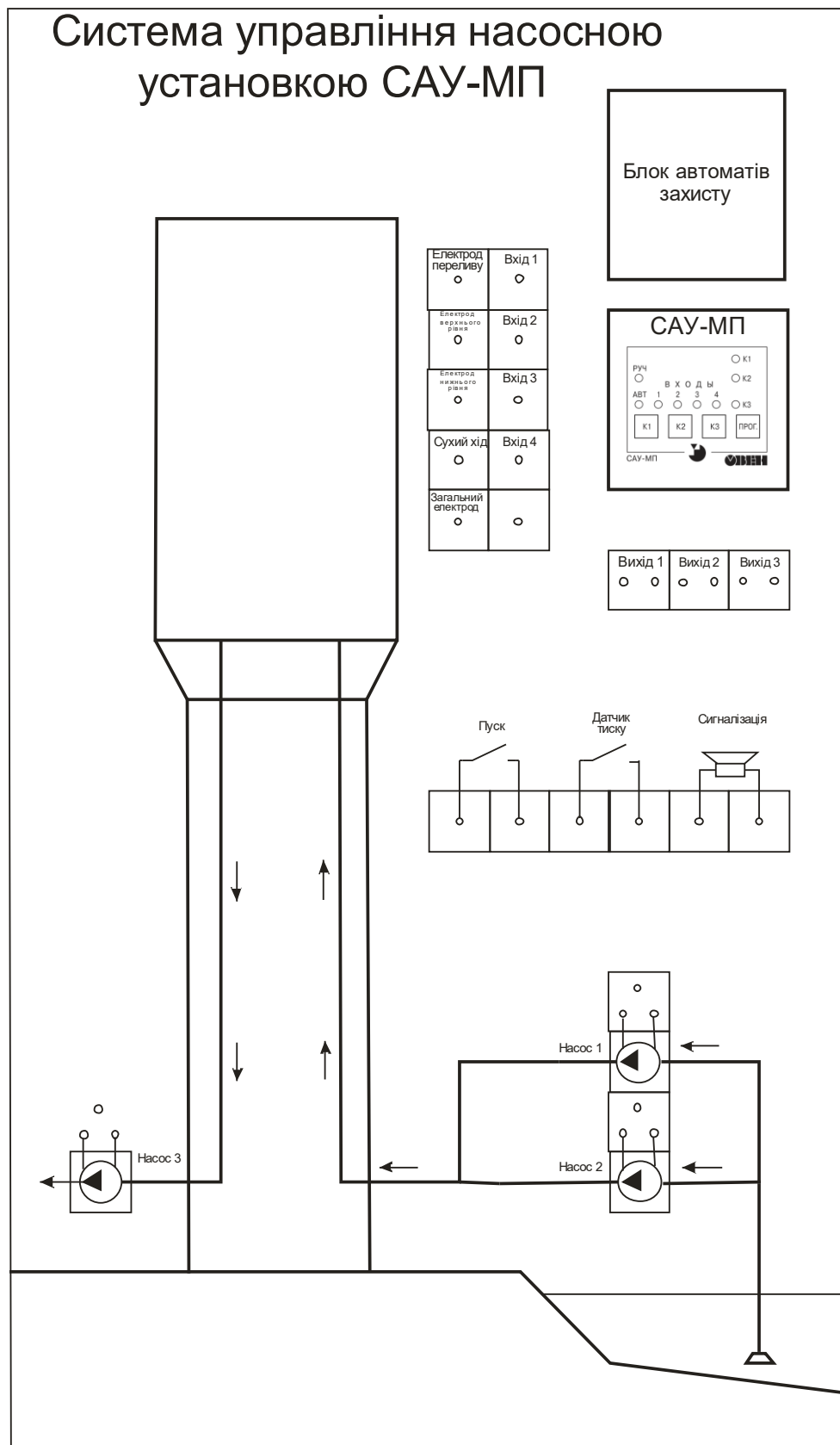


Рис.3. Зовнішній вигляд стенда.

3.2.2 Входи стенда 1-4 виведені на клемні колодки 11,25,14,28 для забезпечення підключення електродів згідно з алгоритмом.

3.2.3. Виходи пристрою виведені на клемні колодки 3-4 (вихід 1), 5-6 (вихід 2). 7-8 (вихід3).

3.2.4. Комутація двигунів насосів 1-3 забезпечується замиканням відповідних контактів, про що сигналізує відповідна індикаторна лампа, встановлена над насосом.

3.2.5. В середній частині стенда встановлені вимикачі «Пуск» та «Датчик тиску» призначені відповідно для запуску алгоритму та імітації датчика тиску в напірному трубопроводі. Виводи вимикачів виведені на клеми.

3.2.6. Замикання клем «Сигналізація» вмикає звуковий оповісник та індикаторну лампу.

3.2.7. В верхній лівій частині стенду розміщено блок автоматичних вимикачів, які забезпечують захист системи управління в цілому та кожний з насосів окремо.

### 3.3. Елементи індикації й керування

3.3.1 На лицьовій панелі приладу (рис. 4) є світлодіодні індикатори: "РУЧ", "АВТ", "К1", "К2", "К3», «ВХОДИ 1,2,3,4», і кнопки , К1, К2,К3 і ПРОГ, призначення яких залежить від режиму роботи приладу.

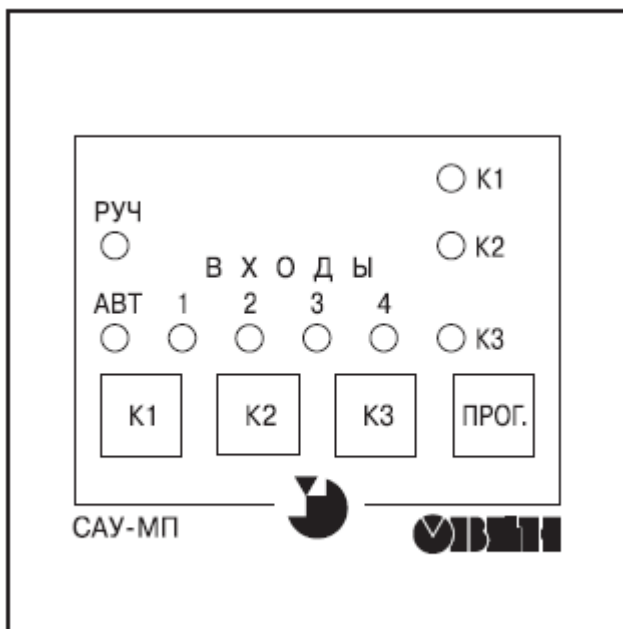


Рис 4. Лицьова панель контролера.

При виконанні заданого алгоритму й у режимі ручного керування світлодіоди "ВХОДИ 1,2,3,4" вказують стан датчиків, а світлодіоди "К1", "К2", "К3" - стан вихідних реле.

Світлодіоди "АВТ" й "РУЧ" загоряються відповідно в режимі автоматичного й ручного керування реле. Кнопка "ПРОГ" служить для переводу приладу з ручного режиму роботи в автоматичний, і назад. У режимі ручного керування включення вихідних реле здійснюється натисканням відповідної кнопки "К1", "К2" або "К3". Вимикання виконується повторним натисканням тієї ж кнопки.

Одночасне миготіння світлодіодів "АВТ" й "РУЧ" вказує на те, що прилад працює в режимі відліку затримки початку виконання алгоритму.

Натискання кнопки "ПРОГ" в цьому режимі приводить до переходу в режим виконання заданого алгоритму.

3.3.2. У режимах зміни уставок таймерів й опорної напруги вхідних пристроїв кнопки слугують для вибору й завдання значень відповідних уставок, а світлодіоди відображають номер або поточне значення обраної уставки.

У режимах прийому-передачі алгоритму використовуються кнопки "ПРОГ" і "К2".

Кнопка "К2" ініціює передачу алгоритму в інший прилад, а "ПРОГ"- прийом і запис алгоритму в енергонезалежну пам'ять.

### **3.4. Робота із приладом**

3.4.1. Прилад САУ-МП може працювати в *ручному* (включений світлодіод "РУЧ" і мигає "АВТ") і *автоматичному* (включений світлодіод "АВТ" і мигає "РУЧ") режимах керування реле.

3.4.2. Якщо із плати індикації приладу вилучені всі перемички, то після подачі живлення 220 В 50 Гц прилад переходить у режим відліку затримки початку виконання алгоритму, заданої в уставці  $T_0 = 16$  с. Цей режим вказується синхронно миготливими світлодіодами "РУЧ" і "АВТ". Після закінчення відліку затримки або короткочасного натискання на кнопку "ПРОГ" прилад переходить до виконання заданого алгоритму (автоматичний режим). При цьому світлодіод "АВТ" горить безупинно, а світлодіод "РУЧ" мигає із частотою 1 Гц.

3.4.3. Переведення приладу в режим ручного керування здійснюється натисканням й утриманням протягом 3 с кнопки "ПРОГ". Після переходу в цей режим світлодіод "РУЧ" горить безупинно, а "АВТ" мигає із частотою 1 Гц.

3.4.4. При роботі в режимі ручного керування можливе включення вихідних реле приладу при натисканні на однойменну кнопку "К1", "К2" або "К3" незалежно від стану датчиків. Реакція на кнопку відбувається після її відпускання. Тривалість натискання повинна бути не менш 2-х с.

3.4.5. Переведення приладу в режим автоматичного керування виконується повторним натисканням кнопки "ПРОГ".

3.4.6. Контроль за станом входів здійснюється по світлодіодам "ВХОДИ 1,2,3,4,", а за станом виходів по світлодіодам "К1", "К2" й "К3".



#### 4. Копіювання алгоритму з комп'ютера в прилад

4.1.1. З'єднайте паралельний порт комп'ютера із приладом спеціальним кабелем №2, як показано на рис. 5.

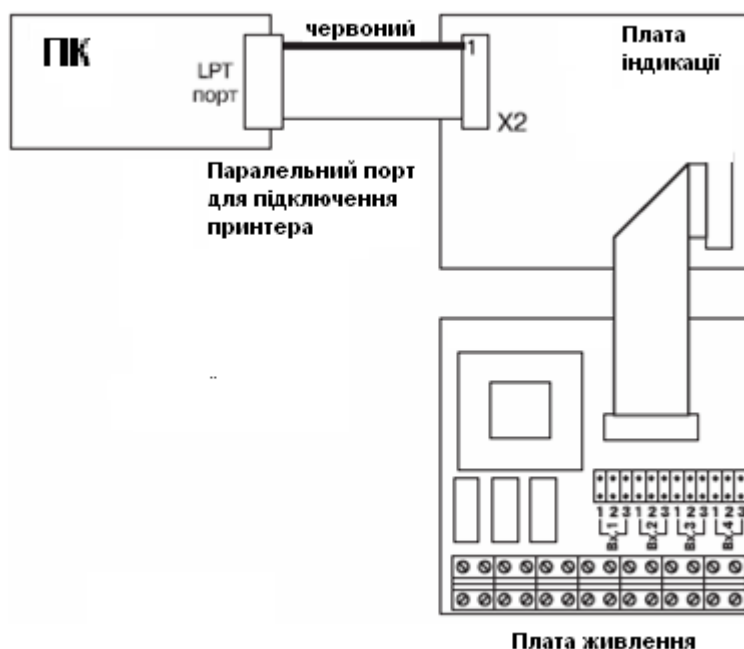


Рис.5. З'єднання приладу з ПК.

**Увага!** З'єднання виконувати тільки при виключеному живленні приладу й комп'ютера.

4.1.2. Подати живлення на прилад і включити комп'ютер.

На чотирьох червоних світлодіодах "1", "2", "3", "4" САУ-МП протягом 2 с індикуються номер режиму, після чого прилад стає в режим прийому алгоритму, про що вказує палаючий світлодіод "АВТ".

4.1.3. Запустіть на комп'ютері програму завантаження алгоритмів для САУ-МП. Користуючись інструкцією до програми виберіть із меню потрібний алгоритм.

4.1.4. На приймаючому приладі натисніть кнопку "К2", щоб перевести його в режим очікування прийому, що індикуюється почерговим миготінням світлодіодів "К1" й "К3".

4.1.5. Натисніть кнопку "ПРОГ". Після цього дайте комп'ютеру команду передачі даних.

По мірі надходження чергового блоку даних синхронно мигають світлодіоди "К1", "К2" й "К3".

4.1.6. Після успішного закінчення передачі припиниться миготіння світлодіодів "К1", "К2" й "К3".

4.1.7. Якщо після закінчення передачі світлодіоди "К1", "К2" й "К3" приймаючого приладу мигають по черзі (рис. 31), операцію запису алгоритму необхідно повторити.

У випадку успішного закінчення процесу зніміть живлення із приладу й від'єднайте кабель від рознімання X2.

## 5. Завдання для виконання.

### 5.1. Робота в ручному режимі.

Підключити насоси 1-3 до відповідних виходів контролера.

Подати на стенд живлення.

Перевести стенд в режим «Ручн».

Включити насос 1, через 1 хв. Включити насос 2. Після заповнення резервуару вимкнути насоси 1 та 2, включити насос 3. Після спорожнення резервуару насос 3 вимкнути.

Вимкнути стенд.

### 5.2. Робота в автоматичному режимі.

Згідно із заданим варіантом з табл. 2 обрати алгоритм роботи пристрою.

Побудувати схему згідно з алгоритмом. Для цього підключити виконавчі механізми, датчики та елементи комутації та сигналізації згідно із схемою (див. додаток).

Провести запис обраного алгоритма в пам'ять контролера (п.4).

5.3. Подати живлення на стенд, запустити алгоритм роботи пристрою.

Після заповнення резервуару відкачування води здійснювати насосом 3 вручну.

Під час виконання алгоритму визначити час спрацювання захисту від «сухого ходу» (якщо цього потребує алгоритм роботи).

Змінити час спрацювання захисту від «сухого ходу» або час зміни насосів шляхом зміни параметрів заданого алгоритма за допомогою конфігуратора (п4.). Повторити п 5.3.

Табл. 2. Варіанти завдань для виконання роботи.

№ варіанта	№ алгоритма	Час спрацювання захисту СХ або ДЕМ	Час зміни насосів
1	11	50 с	1 хв
2	12	40 с	3 хв
3	16	35 с	5 хв
4	06	-	-
5	17	-	4 хв
6	20	40 с	-

## Алгоритми роботи приладу САУ-МП

## Алгоритм 11

Алгоритм призначений для керування основним і резервним насосом у системах водопостачання, що містять у собі два насоси, датчик наявності потоку й, або третій насос, або аварійну сигналізацію. Схема підключення елементів системи до входів приладу вказана на рис.1

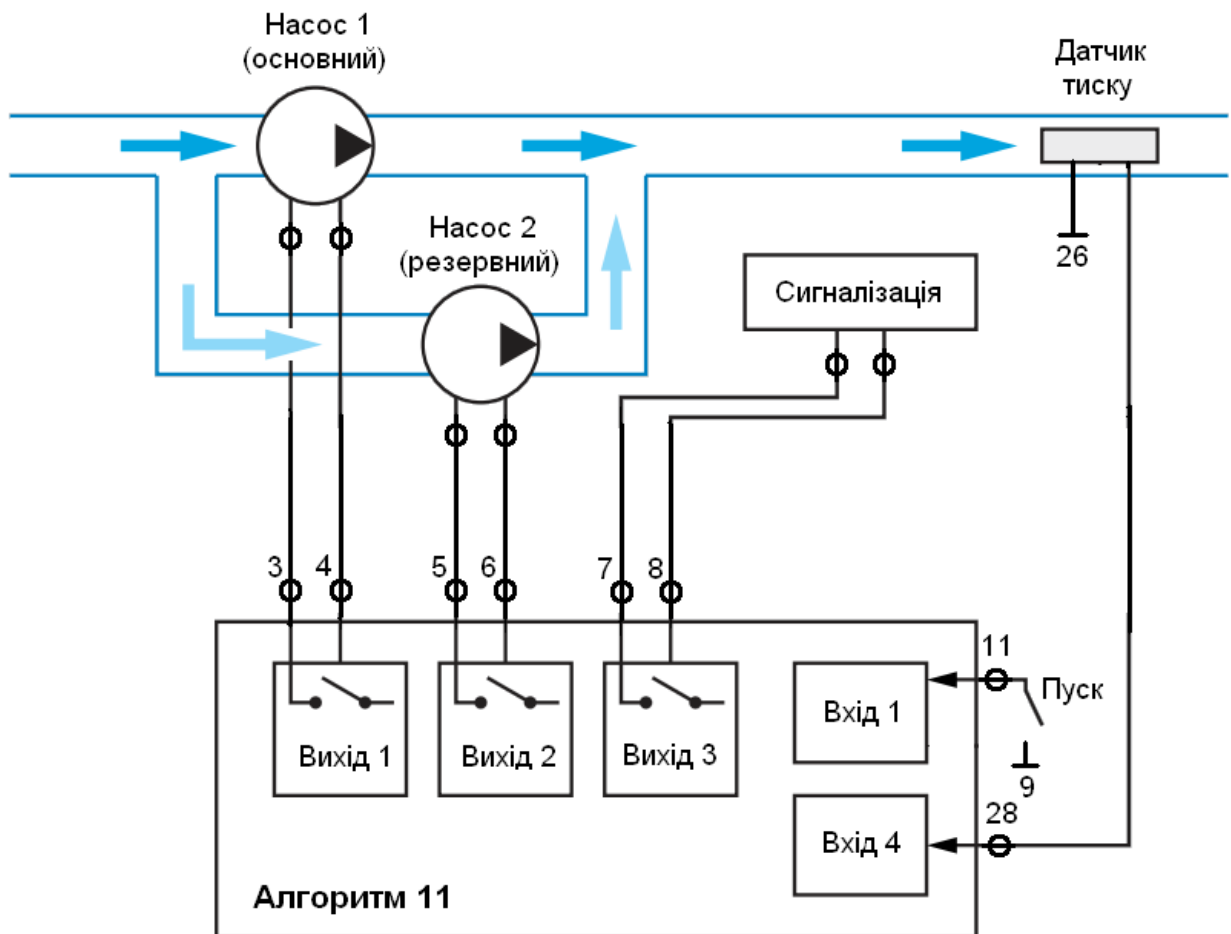


Рис.1. Схема підключень для алгоритму 11.

Реле 1 й 2 управляють роботою основного й резервного насоса. Реле 3 вмикає сигналізацію про аварію або третій насос. Датчик наявності потоку ДЕМ підключається до четвертого входу приладу.

До першого входу може бути підключений тумблер, замикання якого запускає виконання алгоритму.

Після включення живлення САУ-МП відпрацьовує затримку включення  $T_0$ , потім перевіряє стан входу 1 і, якщо тумблер замкнутий, включає перший насос.

Якщо протягом заданого часу в трубі створюється необхідний тиск, контакти датчика наявності потоку замикаються й працює 1-й насос. Після закінчення заданого часу реле 1 вимикається й вмикається реле 2, що управляє другим насосом. Якщо другий насос справний, то контакти ДЕМ виявляться замкнуті, і робота насоса буде продовжена.

Через заданий час прилад вимкне реле 2 і знову ввімкне реле 1, забезпечуючи рівномірне зношування основного й резервного насоса.

Якщо під час роботи одного з насосів контакти ДЕМ розімкнулися на час, більший заданого, або під час пуску двигуна насоса через заданий час контакти ДЕМ не замкнули, прилад вважає, що відбулася аварія й вмикає резервний насос. При цьому несправний насос вимикається, а відповідний йому світлодіод ("K1" або "K2") починає мигати.

Якщо в процесі роботи вийшли з ладу обидва насоси, то вмикається третє реле, до якого може бути підключений третій насос або аварійна сигналізація. У цьому випадку починають мигати обидва світлодіоди "K1" й "K2". Третє реле буде замкнuto до зняття з приладу живлення або до перезапуску алгоритму тумблером, підключеному до входу 1.

Уставки 1-го насоса:

- $T1 = 30$  с - час, протягом якого при запуску двигуна насоса в даному каналі не аналізуються показання датчика тиску;
- $T3 = 2$  с,  $T4 = 2$  с - час, протягом якого при роботі двигуна насоса в даному каналі допускаються "провали" показань датчика тиску;
- $T7 = 1$  с,  $T8 = 1$  с - тривалість періодів включеного й виключеного станів світлодіоди насоса при індикації відмови його двигуна.

Уставки 2-го насоса:

- $T13 = 30$  с - час, протягом якого при запуску двигуна насоса в даному каналі не аналізуються показання датчика тиску;
- $T15 = T16 = 2$  с - час, протягом якого при роботі двигуна насоса в даному каналі допускаються "провали" показань датчика тиску;
- $T19 = T20 = 1$  с - тривалість періодів включеного й виключеного станів світлодіоди насоса при індикації відмови його двигуна.

Загальноприладові установки:

- $T25 = 45$  с,  $T26 = 50$  с - "навчальні" значення; час роботи першого й другого насосів; для реальної роботи повинні мати значення, наприклад, 24 й 24 години.

## Алгоритм 12

Цей алгоритм призначений для підтримки рівня в ємності за показниками двох датчиків. Включення насоса відбувається при осушенні датчика нижнього рівня, а вимикання - при затопленні верхнього. Система містить два насоси, які для забезпечення рівномірності зношування вмикаються приладом по черзі. Контроль дієздатності насосів ведеться по датчику наявності потоку ДЕМ.

Схема підключення елементів системи до входів приладу вказана на рис. 2.

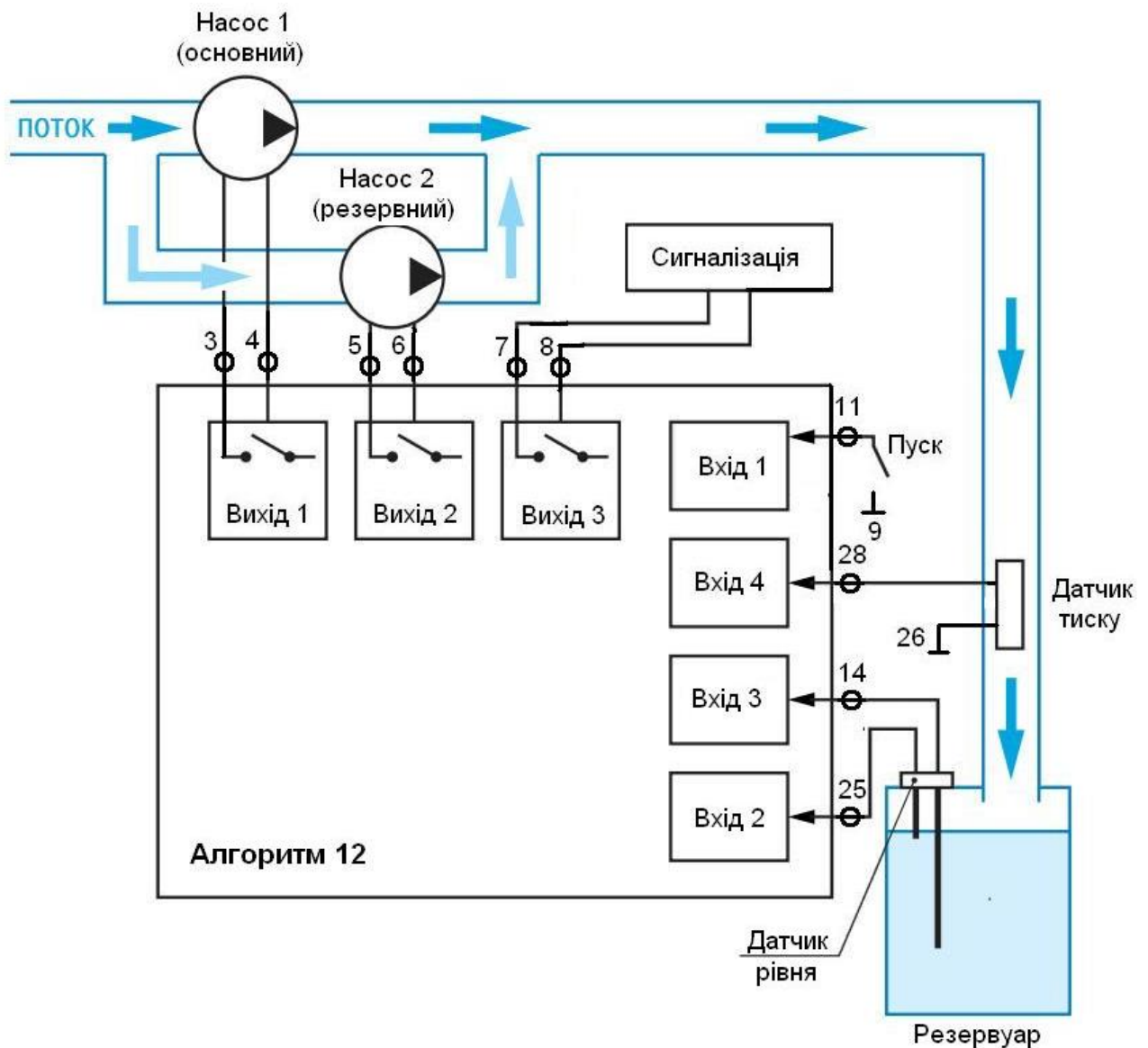


Рис.2. Схема підключень для алгоритму 12.

Датчик верхнього рівня підключається до другого входу приладу, датчик нижнього рівня - до третього. До першого входу може бути підключений тумблер, замкнений стан якого дозволяє почати виконання алгоритму. Датчик наявності потоку ДЕМ підключається до четвертого входу приладу.

Реле 1 й 2 управляють роботою основного і резервного насоса.

Після включення живлення САУ-МП відпрацьовує затримку включення  $T_0$ , а потім перевіряє стан входу 1. Якщо тумблер замкнений, то перевіряється стан довгого електрода (вхід 3). Якщо довгий електрод осушений, то вмикається реле 1, що управляє першим насосом. Якщо протягом заданого часу в трубі створюється необхідний тиск, контакти датчика наявності потоку замикаються, і 1-й насос працює до затоплення датчика верхнього рівня. Наступного разу при осушенні датчика нижнього рівня вмикається реле 2, і заповнювати ємність буде другий насос. У такий спосіб забезпечується рівномірне зношування основного й резервного насоса.

Якщо під час роботи одного з насосів контакти ДЕМ розімкнулися на час, більший заданого, прилад вважає, що відбулася аварія й вмикає резервний насос. При цьому несправний насос вимикається, а відповідний йому світлодіод ("K1" або "K2") починає мигати.

Якщо в процесі роботи вийшли з ладу обидва насоси, то починають мигати обидва світлодіоди "K1" й "K2".

Уставки затримок і їхнього значення, установлені в приладі за замовчуванням, наведені нижче:

Уставки 1-го насоса:

- $T_1 = 30$  с - час, протягом якого при пуску двигуна насоса в даному каналі не аналізуються показання датчика тиску;
- $T_3 = T_4 = 2$  с - час, протягом якого при роботі двигуна насоса в даному каналі допускаються "провали" показань датчика тиску;
- $T_7 = T_8 = 1$  с - тривалість періодів включеного й виключеного стану світлодіода насоса при індикації відмови його двигуна.

Уставки 2-го насоса:

- $T_{13} = 30$  с - час, протягом якого при пуску двигуна насоса в даному каналі не аналізуються показання датчика тиску;
- $T_{15} = T_{16} = 2$  с - час, протягом якого при роботі двигуна насоса в даному каналі допускаються "провали" показань датчика тиску;
- $T_{19} = T_{20} = 1$  с - тривалість періодів включеного й виключеного стану світлодіода каналу при індикації відмови його двигуна.

Загальприладові установки:

для правильного виконання алгоритму повинні бути встановлені значення  $T_{27} = 1$  с,  $T_{28} = 0$  с.

## Алгоритм 16

Цей алгоритм призначений для підтримки рівня в ємності по двох датчиках. Вимикання насоса відбувається при осушенні датчика нижнього рівня, а включення - при затопленні верхнього. Система містить два насоси, які для забезпечення рівномірності зношування вмикаються приладом по черзі. Контроль працездатності насосів ведеться по датчику наявності потоку ДЕМ.

Схема підключення елементів системи до входів приладу показана на рис.3.

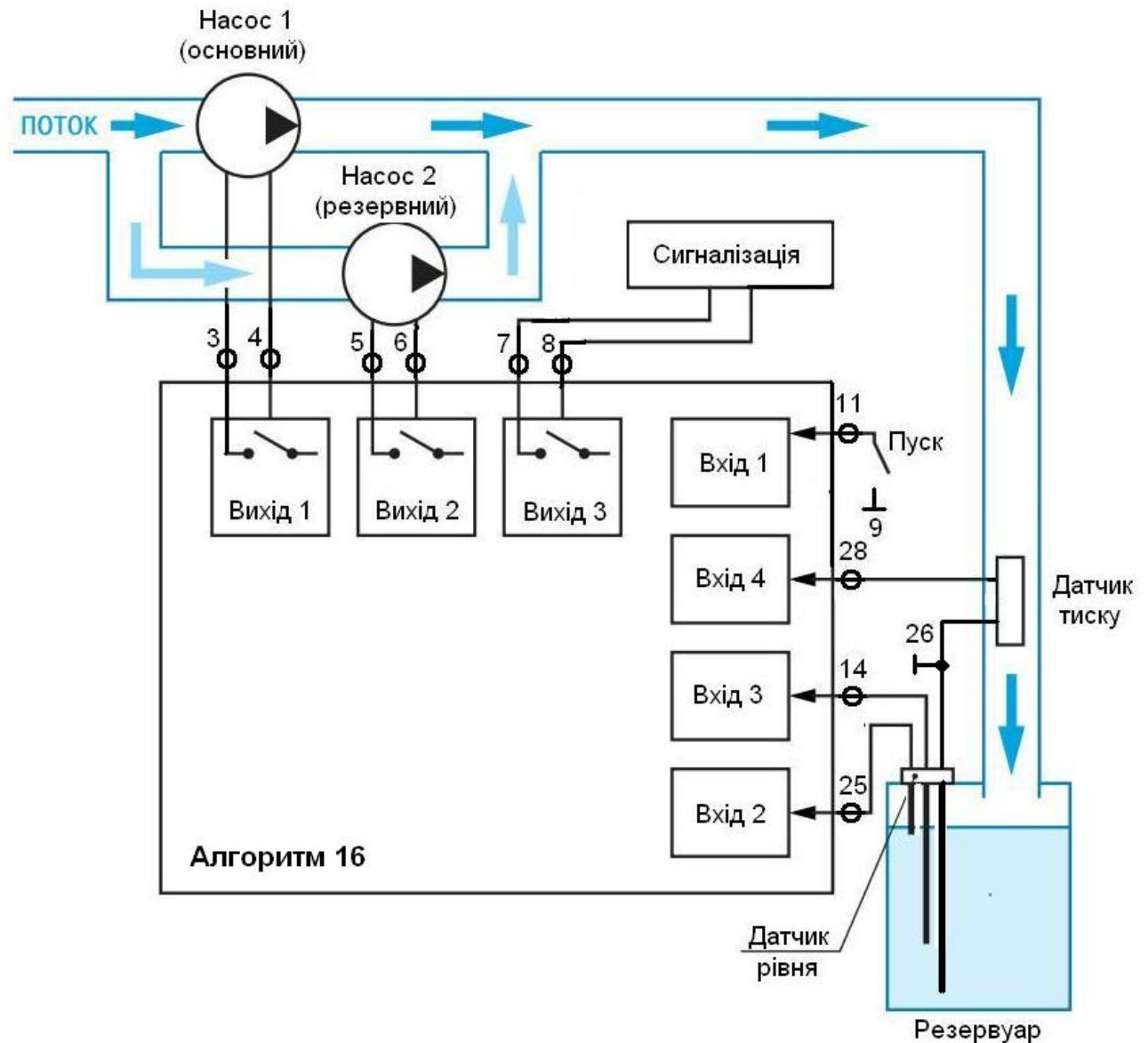


Рис. 3.Схема підключень для алгоритма 16.

Датчик верхнього рівня підключається до другого входу приладу, датчик нижнього рівня - до третього. До першого входу може бути підключений тумблер, замкнений стан якого дозволяє почати виконання алгоритму. Датчик наявності потоку ДЕМ підключається до четвертого входу приладу.

Реле 1 і 2 управляють роботою основного й резервного насоса.

Після включення живлення САУ-МП відпрацьовує затримку включення T0 а потім перевіряє стан входу1. Якщо тумблер замкнений, то перевіряється стан короткого електрода (вхід 2). Якщо короткий електрод замкнений, то включається реле 1, що управляє першим насосом. Якщо протягом заданого часу в трубі створюється необхідний тиск, контакти датчика наявності потоку замикаються й 1-й насос працює до осушення датчика нижнього рівня. Наступного разу при заливанні датчика верхнього рівня вмикається реле 2 й осушувати ємність буде другий насос. У такий спосіб забезпечується рівномірне зношування основного й резервного насоса.

Якщо під час роботи одного з насосів контакти ДЕМ розімкнулися на час, більший заданого, прилад вважає, що відбулася аварія й включає резервний насос. При цьому несправний насос вимикається, а відповідний йому світлодіод ("K1" або "K2") починає мигати.

Якщо в процесі роботи вийшли з ладу обидва насоси, то починають мигати обидва світлодіоди "K1" й "K2", вмикається реле 3 і світлодіод "K3".

Уставки затримок і їх значення, встановлені в приладі за умовчанням, приведені нижче:

Уставки 1-го насоса:

-  $T1 = 30$  с - час, протягом якого при запуску двигуна насоса в даному каналі не аналізуються свідчення датчика тиску;

-  $T3 = T4 = 2$  с - інтервал часу, протягом якого при роботі двигуна насоса в даному каналі допускаються "провали" свідчень датчика тиску;

-  $T7 = T8 = 1$  с тривалість періодів включеного і виключеного станів світлодіода насоса при індикації відмови його двигуна.

Уставки 2- го насоса :

-  $T13 = 30$  с - час, впродовж якого при запуску двигуна насоса в цьому каналі не аналізуються свідчення датчика тиску;

-  $T15 = T16 = 2$  с - час, впродовж якого при роботі двигуна насоса в даному каналі допускаються "провали" свідчень датчика тиску;

-  $T19 = T20 = 1$  с - тривалість періодів ввімкненого і вимкненого станів світлодіода насоса при індикації відмови його двигуна.

Загальноприладові уставки:

для правильного виконання алгоритму мають бути встановлені значення  $T27 = 1$  с,  $T28 = 0$  с.



## Алгоритм 06

Алгоритм 06 призначений для управління трьома незалежними насосами, кожен з яких підтримує рівень рідини в ємності за показаннями датчиків рівня Д1 ... Д3 (рис. 4).

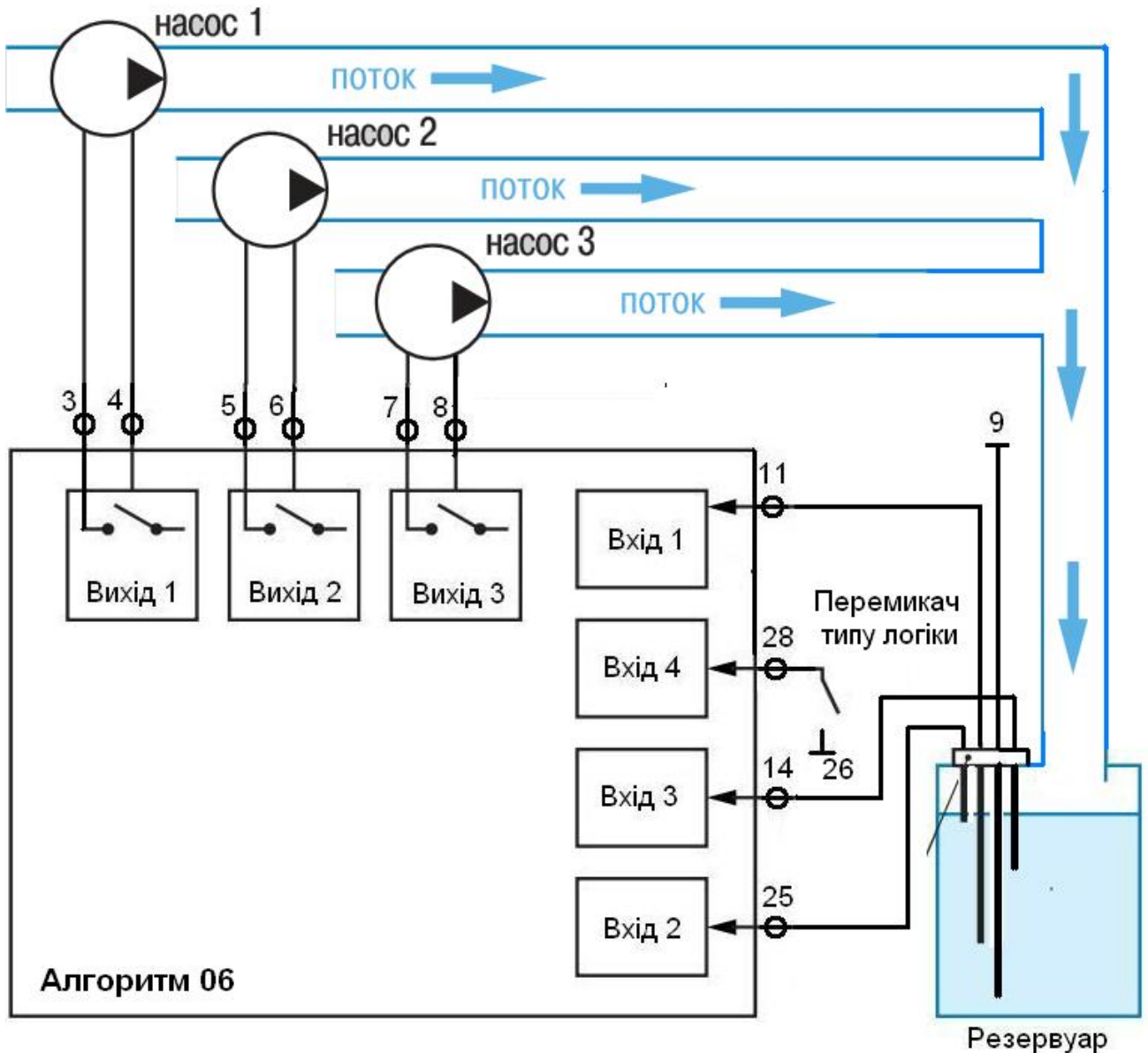


Рис. 4. Схема підключень для алгоритма 06.

Датчики рівня підключені до входів 1 ... 3 приладу.

Прилад може працювати за двома типами логіки - прямий і зворотній. Якщо встановлена перемичка на вході 4, то реалізується пряма логіка, якщо перемичка на вході 4 відсутня - зворотна логіка.

**Примітка.** Всі три насоси працюють по одному типу логіки.

При *прямій логіці* насос вмикається при розмиканні контактів датчика, тобто насос починає накачувати в бак рідину тоді, коли її рівень опуститься нижче рівня контактів датчика.

При *зворотній логіці* насос вмикається при замиканні контактів датчика, тобто насос починає відкачувати рідину з ємності, коли її рівень стане вище рівня контактів датчика.

При зміні стану датчика на час, більше заданого (див. затримки нижче), відбувається вмикання / вимикання насоса згідно заданої логіки. При зміні стану датчика на час, менший уставки затримки, стан насоса не змінюється.

Наприклад, при замиканні контактів датчика на час, більше 7 с (значення уставки затримки по замовчуванню) при прямій логіці працюючий насос 1 вимикається. Ввімкнеться він, якщо контакти датчика Д1 будуть розімкнуті довше 7 с (по замовчуванню). Аналогічним чином працюють і інші насоси.

Уставки затримок та їх значення наступні:

- для 1-го насоса - T1, T2;
- для 2-го насоса - T13, T14;
- для 3-го насоса - T25, T26.

Тут

-  $T1 = T13 = T25 = 7$  с (за замовчуванням) - час, протягом якого не відбувається зміна сигналу з 0 на 1 (викл / вкл);

-  $T2 = T14 = T26 = 7$  с (за замовчуванням) - час, протягом якого не відбувається зміна сигналу з 1 на 0 (вкл / викл).

Значення інших уставок рівні 0, оскільки в роботі даного алгоритму вони не використовуються.

### Алгоритм 17

Алгоритм 17 призначений для управління насосною установкою, яка містить три насоса, які вмикаються по чергові і працюють на одну спільну магістраль, при цьому кожен насос має свій власний датчик тиску, замикання контактів якого свідчить про нормальну роботу насоса. Схема підключення елементів системи до приладу така ж, як у САУ-МП 14 (див. рис. 5).

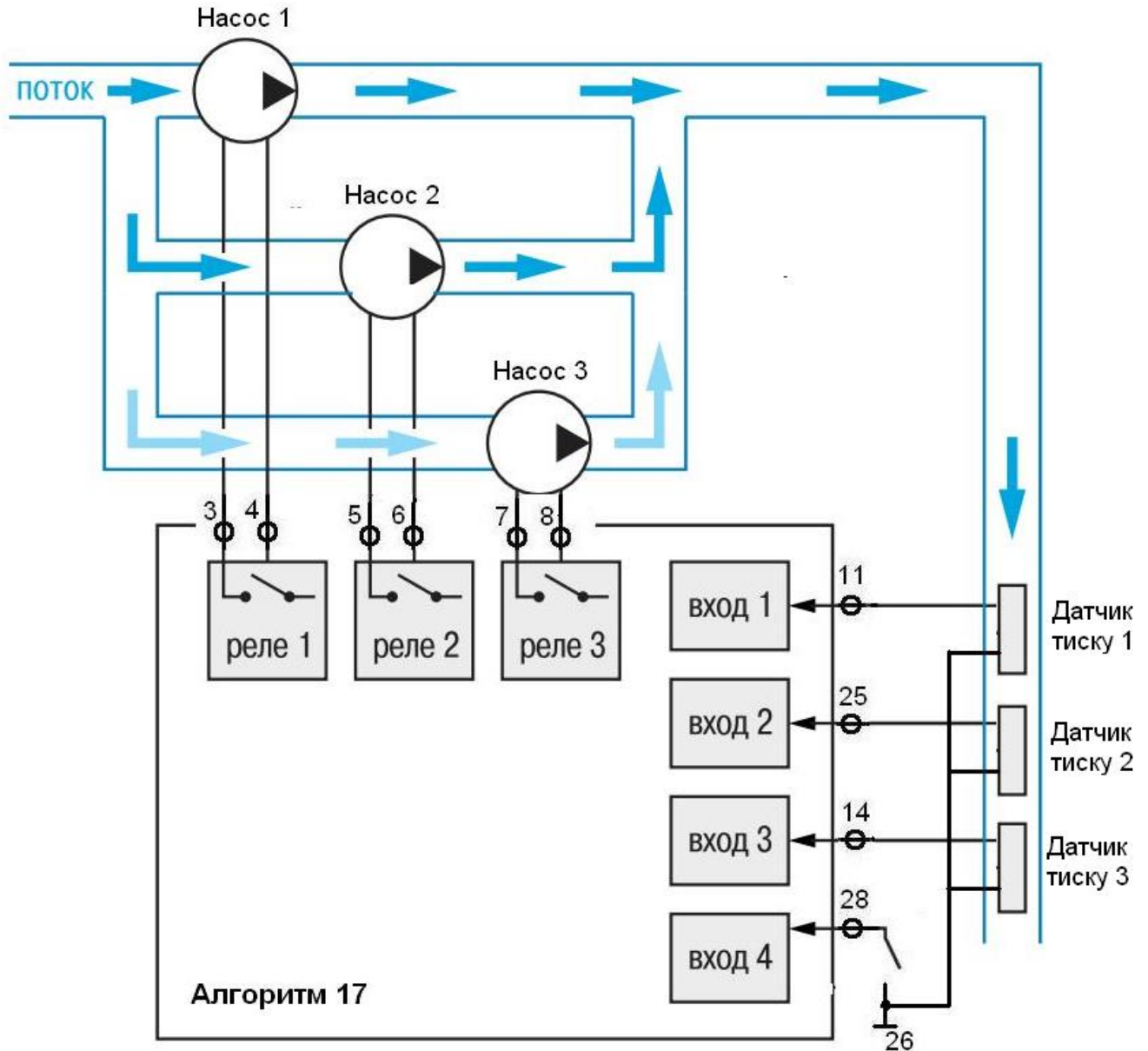


Рис.5. Схема підключень для алгоритму 17.

В автоматичному режимі одночасно працює тільки один насос, по завершенню заданого часу роботи насоса відбувається його вимикання і вмикання наступного насоса в порядку: 1-й – 2-й – 3-й – 1-й – 2-й ... При включенні насоса в роботу протягом певного часу (уставки T1, T13 і T25 для 1-го, 2-го і 3-го насосів, відповідно) не проводиться контроль замикання контактів датчика тиску даного насоса для того, щоб насос набрав необхідні оберти. Якщо ж після закінчення даного інтервалу контакти датчика тиску не замкнулися, то відбувається блокування цього насоса, про що сигналізує

миготінням відповідний світлодіод. Після чого вмикається наступний по порядку з наявних справних насосів. Те ж саме відбувається, якщо під час роботи насоса контакти його датчика тиску розмикаються на час, більший заданого (уставки T3, T15 і T28 для 1-го, 2-го і 3-го насосів, відповідно).

Надалі працюють два насоса що залишилися справними.

При виході з ладу ще одного з решти насосів продовжує працювати останній справний насос, не вимикаючись.

Уставки затримок та їх значення, встановлені в приладі за замовчуванням, наведені нижче.

Уставки 1-го насоса:

- T1 = 10 с - час, протягом якого не контролюється стан датчика тиску при вмиканні насоса;
- T3 = T4 = 2 с - час, протягом якого допускається короткочасне розмикання (замикання) контактів датчиків тиску під час роботи насоса;
- T7 = T8 = 1 с - тривалість періодів ввімкненого і вимкненого стану світлодіода насоса при індикації відмови даного насоса;
- T9 = ... = T12 = 0, оскільки при роботі даного алгоритму вони не використовуються.

Уставки 2-го насоса:

- T13 = 10 с - час, протягом якого не контролюється стан датчика тиску при включенні насоса;
- T15 = T16 = 2 с - час, протягом якого допускається розмикання контактів датчика тиску під час роботи насоса;
- T19 = T20 = 1 с - тривалість періодів ввімкненого і вимкненого стану світлодіода насоса при індикації відмови даного насоса;
- T21 = ... = T24 = 0, оскільки при роботі даного алгоритму вони не використовуються.

Уставки 3-го насоса:

- T25 = 10 с - час, протягом якого не контролюється стан датчика тиску при вмиканні насоса;
- T27 = T28 = 2 с - час, протягом якого допускається розмикання контактів датчика тиску під час роботи насоса;
- T29 - час, протягом якого затримується сигнал блокування при відмові 3-го насоса; необхідно для забезпечення правильної роботи алгоритму приладу;

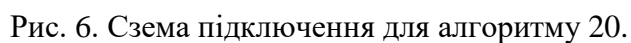
**Увага!** Затримку T29 користувачеві змінювати категорично не рекомендується!

- T35 = T36 = 1 с - тривалість періодів ввімкненого і вимкненого станів світлодіода насоса при індикації відмови даного насоса.

-  $T_{31} = T_{32} = 30$  с - тривалість періодів роботи насосів;

- T33 = T34 = 2T31 - затримки, необхідні для забезпечення належного функціонування ритму приладу.

Алгоритм призначений для підтримки (доливання) рівня рідини в ємності, а також для сигналізації про переповнення і сигналізації або захисту насоса від сухого ходу. Схема підключення наведена на рис. 6.



У ємності встановлюється п'ятиелектродний кондуктометричний датчик. До входу 1 підключається електрод сухого ходу, до входів 2 і 3 підключаються датчики нижнього і верхнього

робочих рівнів, відповідно, до входу 4 підключається електрод переливу. П'ятий електрод здійснює функцію спільного.

Вихід 1 приладу здійснює захист насоса від сухого ходу (послідовно включається в ланцюг управління насосом). Вихід 2 здійснює включення насоса залежно від рівня в ємності (система працює на доливання від нижнього до верхнього робочого рівня).

Вихід 3 здійснює сигналізацію про перелив.

Для запобігання передчасного спрацьовування захисту від сухого ходу введені затримки на вмикання / вимикання реле 1 при змочуванні / осушенні датчика «сухого ходу» ( $T_1$ ,  $T_2$ ).

Для запобігання передчасного спрацьовування захисту при переливу введені затримки на вмикання / вимикання реле 3 при змочуванні / осушенні датчика «аварійного переливу» ( $T_{25}$ ,  $T_{26}$ ).

Уставки затримок та їх значення, встановлені в приладі за замовчуванням, наведені нижче.

$T_1 = 5\text{с}$  - час, протягом якого не відбувається замикання реле 1 при змочуванні датчика «сухого ходу»;

$T_2 = 5\text{с}$  - час, протягом якого допускається осушення електродів датчика «сухого ходу» при роботі насоса;

$T_{25} = 5\text{с}$  - час від моменту заливання датчика аварійного переливу до замикання реле 3;

$T_{26} = 5\text{с}$  - час від моменту осушення датчика аварійного переливу до розмикання реле 3.

## **Лабораторна робота № 2.**

### **Дослідження програмованого регулятора температури МПР-51.**

**Мета роботи:** Вивчити структуру пристрою та основні органи керування. Отримати навички з програмування регулятора за заданим алгоритмом зміни температури в об'єкті управління.

#### **Теоретичні відомості.**

##### **Призначення**

Програмований регулятор типу МПР51 призначений для керування багатоступінчастими температуровологісними режимами технологічних процесів при виробництві м'ясних і ковбасних виробів, в хлібопекарській промисловості, в інкубаторах, при сушінні макаронів, виготовленні залізобетонних конструкцій, сушінню деревини, у кліматичних камерах й ін. по заданій користувачем програмі.

МПР51 - Щ4 дозволяє:

- вимірювати температуру за допомогою термоперетворювачів опору, підключених до входів Тсух, Тволог, Тпрод;
- визначати поточне положення засувки при наявності в них резистивних датчиків положення;
- вимірювати відносну вологість повітря за допомогою датчика психрометричного типу;
- регулювати температуру по двох незалежних каналах;
- задавати програму регулювання із захистом її від несанкціонованого доступу;
- сигналізувати про обрив або коротке замикання в лінії «прилад - датчик»;
- реєструвати контрольовані параметри на IBM-сумісному комп'ютері (за допомогою адаптера інтерфейсу АС2 ОВЕН).

##### **Пристрій і робота приладу**

Структурна схема приладу наведена на рис. 1.

Структурна схема містить:

- 5 входних пристроїв (входів);
- 2 регулятори;
- 4 компаратори;
- 6 цифрових фільтрів;
- пристрій для обчислення різниці температур  $\Delta T$  і відносної вологості  $\psi$ ;
- задатчик програм керування (програм технолога);
- 8 вихідних транзисторних ключів;
- 5 вихідних реле;
- пристрій-сигналізатор про закінчення виконання програми технолога, пристрій зв'язку з комп'ютером, індикатори.

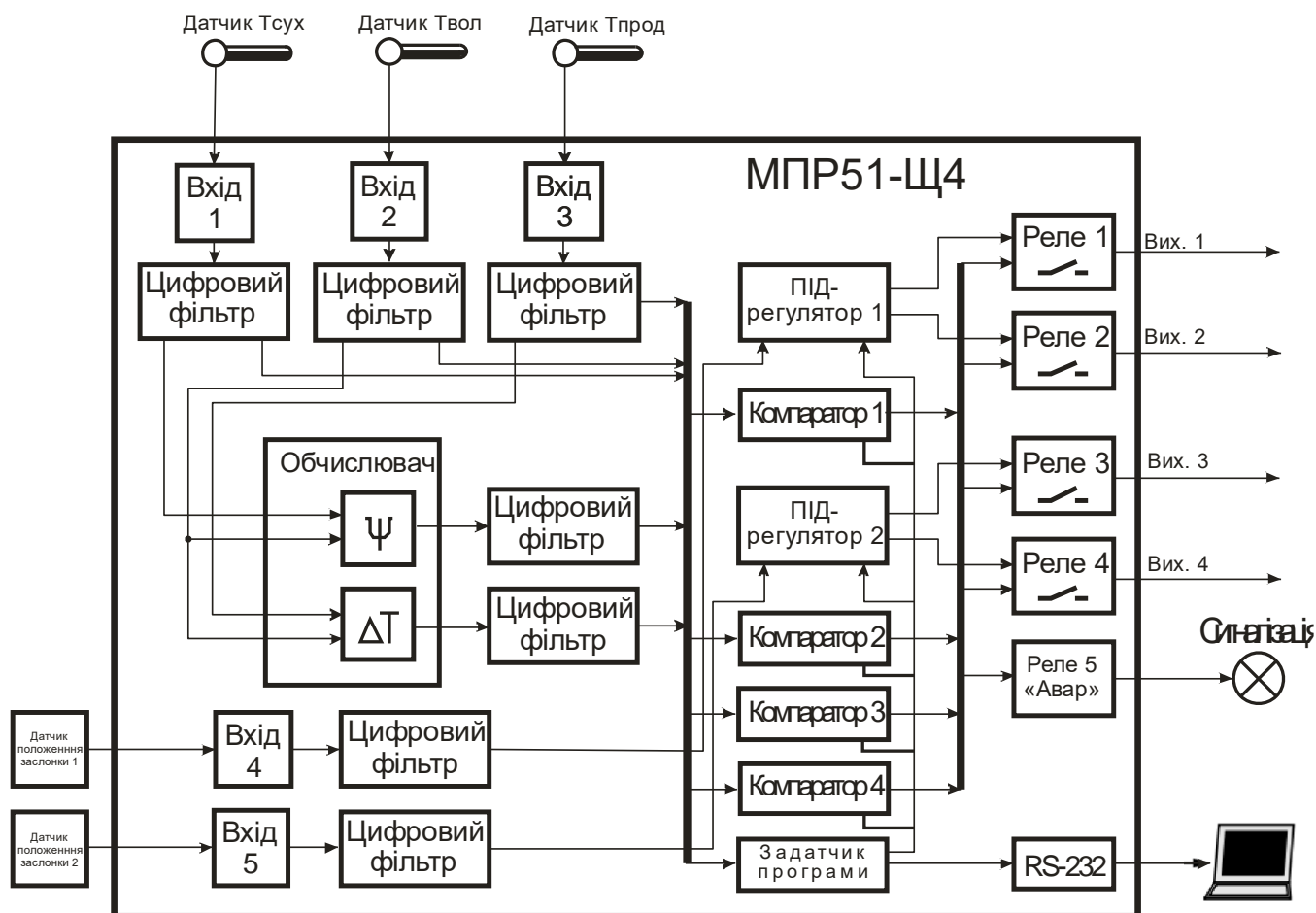


Рис.1 Структура МПР-51.

На входні пристрої приладу надходять сигнали від датчиків температури «сухого» термометра  $T_{\text{сух}}$ ; «вологого» термометра  $T_{\text{волог}}$ ; продукту  $T_{\text{прод}}$  і датчиків положення «Засувка 1» й «Засувка 2».

### Вхідні пристрої

#### Вимір температури

Температура вимірюється за допомогою термоперетворювачів опору.

Це можуть бути мідні датчики TCM 50M ( $R_0 = 50 \text{ Ом}$ ), TCM 100M ( $R_0 = 100 \text{ Ом}$ ) і платинові датчики TСП 50П ( $R_0 = 50 \text{ Ом}$ ), TСП 100П ( $R_0 = 100 \text{ Ом}$ ).

Робота таких датчиків заснована на температурній залежності електричного опору металів. Датчик, як правило, виконаний у вигляді котушки з тонкої мідного або платинового дроту на каркасі з ізоляційного матеріалу, укладеної в захисну гільзу.

Термоперетворювачі опору характеризуються двома параметрами:  $R_0$  – опір датчика при  $0^\circ\text{C}$  і  $W_{100}$  – відношення опору датчика при  $100^\circ\text{C}$  к його опору при  $0^\circ\text{C}$ .

У приладах МПР51;Щ4 використовується трипровідна схема підключення термоперетворювачів опору (додаток. В, рис. В.1). До одному з виводів термоопору приєднуються два проведення, а третій підключається до іншому виводу. Така схема при дотриманні умови рівності опору всіх трьох проводів (якщо проведення одного



перетину, то рівності їхньої довжини), дозволяє компенсувати опір сполучних проводів.

Термоперетворювачі опору можуть підключатися до приладу з використанням двупровідної схеми підключення (див. додаток В, рис. В2), але при цьому відсутня компенсація опору сполучних проводів, тому буде спостерігатися деяка залежність показань приладу від коливань температури проводів і від збільшення довжини лінії зв'язку «датчик - прилад».

### **Визначення положення засувки за допомогою резистивного датчика**

Опір датчика повинен перебувати в межах від 100 до 1000 Ом; при більших значеннях опору завадостійкість приладу буде знижуватися.

Положення засувки відображається на індикаторі «ПАРАМЕТР» у відсотках: закритому стану засувки відповідає показання 0 %, відкритому - 100 %.

Оскільки крайні положення движка змінного резистора датчика положення не завжди відповідають повному відкриттю або закриттю засувки, передбачається процедура калібрування датчика при повністю закритому стані (min) - 0 % і повністю відкритому стані (max) - 100 %; показання на індикаторі між цими положеннями лінійно залежать від опору змінного резистора.

Порядок дій користувача при калібруванні датчика положення приводиться у додатку. Г.

### **Фільтрація вхідних величин**

Для ослаблення впливу зовнішніх імпульсних перешкод на якість регулювання у приладі передбачена фільтрація результатів вимірів вхідних величин. Фільтрація здійснюється незалежно для кожного каналу виміру, а також при розрахунку вологості.

У приладі використовується модель фільтра першого порядку. Для кожного каналу виміру окремо можна при необхідності вводити свою постійну часу фільтра, рис. 2 (програмувальні параметри d01, d02, d03, d04, d05, d06, додаток Д). При значенні програмувального параметра, рівному нулю, фільтрації не відбувається. При збільшенні значення параметра зростає інерція зміни показань датчика.

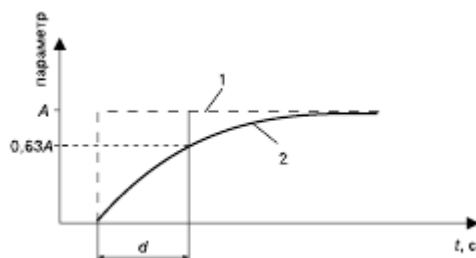


Рис. 2. Принцип дії фільтра:

- 1 - обмірюване значення параметра;
- 2 - значення параметра після фільтра;
- d – постійна часу фільтра;
- A – чергове обмірюване значення параметра

Для постійної часу фільтрів стає помітною при її значеннях, що перевищують час опитування датчиків (час опитування - приблизно 4 с при значеннях програмувального параметра  $\alpha_4 = 001, 002, 003$  й 6 с при  $\alpha_4 = 004, 005$  (див. додаток Д).

**ПРИМІТКА.** Значення постійних часу фільтра для «сухого» й «вологого» термометрів впливають на індикацію відносної вологості навіть при значенні параметра  $\alpha_4 = 000$ .

### Обчислення вологості й різниці температур камери й продукту

Пристрій для обчислення  $\Delta T$  й  $\psi$  крім виміру трьох вхідних величин  $T_{\text{сух}}$ ,  $T_{\text{волог}}$ ,  $T_{\text{прод}}$  обчислює різниця між  $T_{\text{сух}}$  і  $T_{\text{прод}}$ , рівну  $\Delta T$ , а також, використовуючи величини  $T_{\text{сух}}$  і  $T_{\text{волог}}$ , обчислює відносну вологість повітря  $\psi$ .

Прилад обчислює відносну вологість повітря по психрометричному методу. Для цього використовуються значення температури «сухого» й «вологого» термометрів. «Сухий» термометр – це звичайний термометр, що вимірює температуру газового середовища в камері. «Вологий» термометр – це термометр із зволженим ґнотом на своєму чутливому елементі. На основі напівемпіричних формул виведена загальноприйнята формула розрахунку відносної вологості повітря  $\psi$ , %.

$$\psi = \frac{A_{\text{дівіа}}}{A_{\text{пдоо}}} - \frac{A \times D \times (O_{\text{пдоо}} - O_{\text{дівіа}})}{A_{\text{пдоо}}},$$

де  $E_{\text{волог}}$  – максимально можливий парціальний тиск водяної пари при температурі повітря  $T_{\text{волог}}$ , °C;

$E_{\text{сух}}$  – максимально можливий парціальний тиск водяної пари при температурі повітря  $T_{\text{сух}}$ , °C;

$P$  – атмосферний тиск, Па (при обчисленні прийняте  $P = 101308$  Па);

$T_{\text{сух}}$  – температура «сухого» термоперетворювача, °C;

$T_{\text{волог}}$  – температура «вологого» термоперетворювача, °C;

$A$  – психрометричний коефіцієнт (психрометрична постійна).

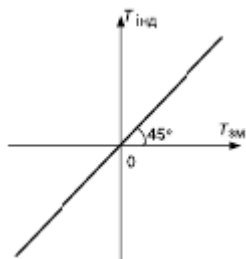


Рис.3. Характеристика справного датчика

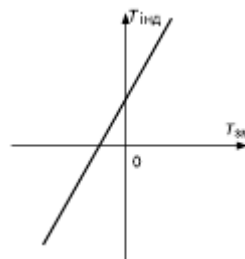


Рис.4. Характеристика датчика, зміненої внаслідок погрешностей

### Корекція показань датчиків температури

Справний датчик повинен мати характеристику (залежність що індукується приладом температури  $T_{\text{інд}}$  від вимірюваної температури  $T_{\text{зм}}$ ), зображену на рис. 3.

Внаслідок погрешності виготовлення, викликану погрешністю намотування вимірювального резистора, невідповідністю матеріалу датчика стандарту, старіння

датчика, характеристика може змінюватися й, у загальному випадку, може прийняти вид, показаний на рис. 4. Для «виправлення» характеристики в приладі передбачено корекцію показань датчиків температури.

Для зрушення характеристики у вертикальному напрямку (рис. 5) треба ввести коригувальне значення  $\delta$ , програмувальні параметри «зрушення характеристики» 1с1 (2с1, 3с1) (див. додаток Д).

Якщо характеристику треба зрушити нагору (збільшити значення що індукується), то задається позитивний знак виправлення (параметри 1с2 (2с2, 3с2) = 000); якщо – униз, то негативне значення, (параметри 1с2 (2с2, 3с2) = 001), див. додаток Д.



Рис.5. Корекція  
(«зсув характеристики»)



Рис.6. Корекція  
(«нахил характеристики»)

Для повороту графіка навколо точки, що відповідає 0 °С (рис. 6), треба ввести коригувальне значення  $\alpha$ , програмувальні параметри «нахил характеристики» 1с3 (2с3, 3с3) (див. додаток Д). Якщо треба повернути графік по годинниковій стрілці (зменшити значення що індукується), то задається від'ємний знак виправлення (параметри 1с4 (2с4, 3с4) = 001), якщо проти годинникової стрілки (збільшити значення що індукується), то знак виправлення додатний (параметри 1с4, 2с4, 3с4 = 000) (див. додаток Д).

У результаті корекції значення температури, виведене на індикатор, буде дорівнює, °С:

$$\dot{O}_{\text{за}} = (\dot{O}_{\text{ср}} + \delta) \times (1 + \alpha / 100\%).$$

Необхідно мати на увазі, що  $\delta$  задається в градусах, а  $\alpha$  - у відсотках від обмірюваної величини.

Приклад корекції показань температурного датчика наведений у додатку Ж.

## Компаратори

Компаратор – це пристрій, призначений для порівняння вхідної величини з еталонною; при цьому компаратор має два стани вихідного сигналу: стан вихідного сигналу змінюється, якщо вхідна величина перетинає граничний рівень (еталон).

### Способи використання компараторів:

- для оповіщення про аварійні або близькі до аварійних ситуаціях (тому їх ще називають компараторами сигналу ТРИВОГА);
- для двопозиційного регулювання;

- для включення (вимикання) устаткування при досягненні керуючим сигналом встановленого рівня або виході його за встановлені межі.
- для включення (вимикання) устаткування після закінчення заданого часу.

Робота кожного із чотирьох компараторів регулюється наступними

### **ПРОГРАМОВАНИМИ ПАРАМЕТРАМИ:**

- вхідна величина компаратора (C01);
- стан виходу компаратора (C02);
- уставки компаратора (C03...C04);
- логіка роботи компаратора (C05);
- блокування спрацьовування компаратора до 1-го досягнення уставки (C06, C07);
- блокування спрацьовування компаратора до зняття зовнішнього впливу (C08);
- параметри затримки спрацьовування компаратора (C09, C10).

Програмовані параметри c01...c10 (див. додаток Д) для кожного компаратора встановлюються на рівні L1 (див. розд. 4) для кожної програми в цілому (окремо для кожного кроку задати уставки й інші параметри компаратора не можна). Якщо компаратор не використовується, то його вихід треба відключити, задавши значення параметра c02 = 000.

### **Вхідна величина компаратора**

На вхід компаратора подається одна з п'яти що вимірюються або що обчислюються вхідних величин ( $T_{\text{сух}}$ ,  $T_{\text{прод}}$ ,  $T_{\text{волог}}$ ,  $\Delta T$ ,  $\psi$ ) безпосередньо або вхідний сигнал, що поступає на вхід одного з регуляторів (параметр c01).

### **Вихід компаратора**

За допомогою параметра що програмується c02 (див. додаток Д) користувач визначає, на яке реле буде подаватися вихідний сигнал компаратора. На одне і те ж реле можуть бути подані вихідні сигнали від декількох компараторів (тобто може бути об'єднання їхніх виходів за схемою «АБО»).

### **Уставки компаратора**

За допомогою параметрів що програмуються c03 й c04 (див. додаток Д) задають значення, відповідно, верхньої й нижньої уставок компаратора, тобто гранично припустимі для даного технологічного процесу значення контрольованої величини. Верхньою уставкою автоматично стає уставка з більшим значенням.

### **Логіка роботи компаратора**

Закон, відповідно до якого буде працювати компаратор, встановлюється за допомогою параметра що програмується c05 (див. додаток Д).

При c05 = 000 компаратор замикає своє вихідне реле у випадку перевищення контрольованою величиною встановленого значення верхньої уставки, розмикає реле при зменшенні вихідної величини нижче нижньої установки. Цей закон використовується, наприклад, при керуванні «холодильником»; для сигналізації про перевищення припустимої температури нагрівання.

При c05 = 001 компаратор замикає своє вихідне реле у випадку, коли контрольована величина стає менше значення нижньої уставки, розмикає реле, коли вхідна величина перевищить верхню уставку. Цей закон використовується,

наприклад, для підтримки температури за допомогою «нагрівача»; для сигналізації про остигання об'єкта нижче певної температури.

Закони використовуються також для включення устаткування по досягненні певної температури.

При  $s05 = 002$  компаратор використовується для сигналізації про досягнення контрольованою величиною (наприклад, температурою) встановлених меж.


При  $s05 = 003$  компаратор використовується для сигналізації про вихід контрольованої величини за встановлені межі.

### **Блокування спрацювання компаратора до 1-го досягнення уставки**

У випадках, коли на початку технологічного процесу, наприклад, при нагріванні, різниця між вихідною температурою й уставкою набагато перевершує встановлене значення верхньої уставки компаратора, повинне відбутися аварійне спрацювання компаратора. Щоб уникнути цього, у приладі передбачений параметр що програмується  $s06$  (див. додаток Д), що дозволяє ігнорувати перше «непотрібне» спрацювання компаратора, а аналогічна ситуація може виникнути при переході від одного кроку програми до іншого, якщо уставки цих кроків далеко відстоять один від одного.

Блокувати спрацювання компаратора до 1-го досягнення уставки можна як на початку програми (за допомогою параметра що програмується  $s06$ ), так і на початку кожного кроку програми (за допомогою параметра  $s07$ ).

### **Блокування спрацювання компаратора до зняття зовнішнього впливу**

Прилад може працювати в режимі, коли після першого спрацювання компаратор більше не реагує на свій вхідний сигнал і залишається в такому положенні (коли його вихідне реле замкнене) до закінчення програми (до натискання кнопки ). Цей режим роботи компаратора реалізується задаванням параметру  $s08$  значення «001» (див. додаток Д).

За допомогою цього параметра можливо фіксувати факт виходу керованої величини за встановлену межу (межі) протягом однієї програми.

### **Час затримки спрацювання компаратора**

При виході керованої величини за встановлені межі можна затримати у часі момент спрацювання компаратора:

- якщо час знаходження регульованої величини  $t > t_1$  (часу затримки спрацювання компаратора), то компаратор спрацює;
- якщо час знаходження регульованої величини  $t < t_1$ , то компаратор не спрацює.

Цей режим роботи компаратора реалізується шляхом задавання параметру  $s09$  ненульового значення (див. додаток Д).

### **Час затримки відпускання компаратора**

Аналогічно задається затримка відпускання компаратора:

- якщо час знаходження керованої величини нижче встановленої межі після спрацювання компаратора менше  $t_2$  (встановленого часу затримки відпускання компаратора), то відпускання не відбувається.

- якщо більше, то після закінчення часу  $t_2$  відбувається відпускання компаратора.

Цей режим роботи компаратора реалізується задаванням параметру  $c_{10}$  на ненульового значення (див. додаток Д).

### **Вихідні реле**

На п'ять вихідних реле приладу надходять вихідні сигнали:

- на перше й друге реле - з першого регулятора;
- на третє й четверте реле - із другого регулятора.

П'яте реле «АВАРІЯ» автоматично замикається при виникненні аварійної ситуації (див. нижче пп. 8.6.2, 8.6.4). Реле «АВАРІЯ» може також використовуватися для виводу вихідного сигналу компараторів.

Якщо регулятор управляє засувками або одночасно «нагрівачем» й «холодильником», то він використовує обидва «своїх» реле. Якщо регулятор управляє тільки «нагрівачем» або тільки «холодильником», то він використовує одне «своє» реле. При цьому перший регулятор використовує друге реле, другий регулятор використовує четверте реле. У цьому випадку перше й третє реле залишаються вільними й можуть використовуватися для виводу на них вихідних сигналів з компараторів.

Вихідний сигнал компаратора на відміну від вихідного сигналу регулятора можна направити на кожне реле. Реле, на яке призначається вихід компаратора, повинно бути вільним. У цьому випадку регулятор повинен бути або відключений, і тоді звільняються обидва його реле, або регулятор повинен управляти виконавчим пристроєм типу «нагрівач» або «холодильник», і тоді вільне одне його реле.

На одне вільне реле можна призначати виходи декількох компараторів, поєднуючи виходи цих компараторів за схемою «АБО», при цьому реле буде замикатися від сигналу будь-якого компаратора, вихід якого призначений на це реле.

### **Задатчик програм керування**

Задатчик програм керування служить для задавання уставок регуляторів і компараторів, режимів роботи транзисторних ключів при виконанні програм керування (програм технолога).

Програми керування (програми технолога) створюються користувачем в відповідності з технологічним процесом, кожної стадії технологічного процесу відповідають окремі ділянки програми технолога - кроки.

Параметри програми технолога можна розділити на дві групи – параметри власне програми й параметри кроку.

**Примітка.** Параметри компараторів описані вище.

### **Параметри програм**

Параметри програм включають параметри, що відносяться безпосередньо до програми (Н01 й Н02), і параметри компараторів –  $c_{01} \dots c_{10}$  (див. додаток Д).

За допомогою параметра Н01 задається число кроків програми.

За допомогою параметра Н02 можливе створення циклів. Цикли можуть складатися з декількох кроків, наприклад, при організації добового циклу роботи

теплиць, або з одного кроку, наприклад, при регулюванні температури в приміщеннях для зберігання продукції, де підтримуються постійні кліматичні умови невизначено довгий час.

### **Параметри кроків**

Кожен крок програми описується трьома групами параметрів.

Перша група параметрів n01...n08 (див. додаток Д) задає режими роботи транзисторних ключів (див. п. 3.2.4).

Друга група параметрів Y01...Y05 задає умови переходу від одного кроку до іншого.

Третя група параметрів E01...E05 (E...01... E.05) задає уставки для 1-го (2-го) регуляторів.

### **Умова переходу до наступного кроку (умова закінчення кроку)**

**Крок програми керування** (програми технолога) може бути заданий:

- по тривалості (за часом), при цьому параметр Y01 = 001;
- по досягненні встановленого значення керованої величини, при цьому параметр Y01 = 000;
- по виконанні перших двох умов; крок закінчиться по події, що відбудеться останнім: якщо минув встановлений час кроку, то по досягненні керованою величиною встановленого значення, і навпаки; параметр Y01 = 002;
- по виконанні перших двох умов; крок закінчиться по події, що відбудеться першим: після закінчення часу кроку, якщо не досягнуто встановленого значення керованої величини, і навпаки, при цьому параметр Y01 = 003.

Умови переходу до наступного кроку залежно від значення температур Tсух, Tволог, Tпрод, ΔT і вологості ψ по відношенню до їх уставок задаються параметром Y02.

Уставка температури (або вологості), по досягненні якої відбудеться перехід до наступного кроку, задається параметром Y03.

Час тривалості кроку задається параметрами Y04 й Y05. Максимально можлива уставка для годин - 63 години, а для хвилин - 59 хвилин.

### **Параметри 1-го (2-го) регуляторів на даному кроці**

Вхідна величина регулятора задається параметрами E01 (для 1-го регулятора) і E.01 (для 2-го регулятора).

Уставка вхідної величини 1-го (2-го) регуляторів задається параметрами E02 (E.02) – ціла частина, E03 (E.03) – дробова частина.

Швидкість виходу керованої величини на уставку задається параметрами E04 (E.04) – значення швидкості виходу на уставку; E05 (E.05) – знак швидкості (позитивний - ріст вхідної величини, негативний - зниження вхідної величини).

### **Завдання для виконання дома:**

1. Вивчити структуру та призначення блоків пристрою.
2. Накреслити схему системи управління температурою в об'єкті у звіті.
3. Визначити за допомогою яких реле здійснюється рулювання нагрівом та охолодженням.

### **Завдання для виконання у лабораторії:**

1. Побудувати криву розгону об'єкта. Для цього необхідно:

- встановити значення спрацювання компаратора 1 (нагрів) та компаратора 2 (охолодження) на рівні 100 °С. Для цього перевести пристрій в режим програмування і ввести наступні параметри:

- H01 – 001 – кількість кроків в програмі 1;

- H02 – 000 – початковий крок програми;

Настройка компаратора 1 (нагрів)

- C01 – 001 – вх. значення - температура продукту;

- C02 – 001 – вихід компаратора 1-е реле (нагрів);

- C03 – 100 – нижнє значення температури уставки компаратора;

- C04 – 105 – верхнє значення температури уставки компаратора;

- C05 – 001 – реле розмикається при досягненні верхньої уставки;

- C06, C07, C08 – 000 – блокування відключене;

- C09, C10 – 000 – час затримки;

Для інших компараторів параметр C02 – 000 – компаратори відключені;

- перевести регулятор в робочий режим;

- за допомогою перемикача вибрати потужність нагрівача ( задається викладачем).

- після включення нагрівача за показаннями годинника і часів зняти дані нагріву об'єкта. Дані знімаються доти, доки температура в об'єкті перестає підвищуватись протягом 5 – 10 хв.

Дані спостережень звести в таблицю.

Побудувати графік залежності  $T_{\text{прод}} = F(t)$ . За графіком визначити стале температури в об'єкті, час запізнення, постійну часу..

2. Побудувати криву природнього охолодження охолодження об'єкта.

Для цього кнопкою «СТОП» зупинити виконання програми.

Зняти показання температури в часі.

Побудувати графік залежності  $T_{\text{прод}} = F(t)$ . °С

Дані спостережень занести в таблицю.

3. Скласти програму автоматичного двопозиційного регулювання температури .

За уставку температури нагрівача прийняти значення на 5-8 °С нижче за отримане значення в п.1 (верхнє значення  $t_{\text{вн}}$ ). Значення нижньої уставки компаратора прийняти  $t_{\text{нн}} - 5$  °С. Для компаратора охолоджувача прийняти наступні значення уставок:

Нижня температура виключення охолоджувача  $t_{\text{он}} = t_{\text{нн}}$

Верхня температура включення охолоджувача  $t_{\text{ов}} = t_{\text{вн}}$ .

Примітка: параметр C05 для компаратора 2 (охолодження) дорівнює 000 ( реле замикається при значенні температури вище за встановлену і розмикається при значенні меншу за встановлену.

Записати програму в пристрій.

Зняти дані регулювання температури для 2-х повних циклів регулювання.

Дані спостережень занести в таблицю.



Побудувати графік залежності  $T_{\text{прод}} = F(t)$ .

**Склад звіту:**

- Назва, мета роботи.
- Теоретичні відомості.
- Таблиці результатів спостережень, графіки.
- висновки.

## ДОДАТКИ

### Приложение Д (обязательное)

#### Программируемые параметры

Обознач.	Название	Допустимые значения	Комментарии
----------	----------	---------------------	-------------

#### ► Уровень L1. Параметры программы технолога

##### Параметры программ

NO1	Количество шагов в программе	001...099	—
-----	------------------------------	-----------	---

NO2	Номер шага программы, который является начальным шагом цикла	000...099	—
-----	--	-----------	---

##### Параметры компараторов C1...C4

C01	Входная величина компаратора	001	Температура продукта ( $T_{\text{prod.}}$ ), [град.]
		002	Температура камеры (сухого термометра, $T_{\text{сух.}}$ ), [град.]
		003	Температура влажного термометра ( $T_{\text{влаж.}}$ ), [град.]
		004	$\Delta T = T_{\text{сух.}} - T_{\text{влаж.}}$ , [град.]
		005	Относительная влажность $\Psi$ , [%]
		006	Входная величина 1-го регулятора
		007	Входная величина 2-го регулятора

C02	Состояние выхода компаратора	000	Выход компаратора отключен
		001...004	1...4-е реле
		005	5-е реле, светодиод «Авария» не горит
		006	5-е реле, светодиод «Авария» горит

C03	1-я (верхняя) уставка компаратора	-99...+999	[град.]
		0...99	[%]

C04	2-я (нижняя) уставка компаратора	-99...+999	[град.]
		0...99	[%]

C05	Логика работы компаратора	000	Реле компаратора замыкается при значении контролируемой величины, большем верхней уставки, а размыкается — при меньшем нижней уставки
		001	Реле замыкается при значении величины, меньшем нижней уставки, а размыкается — при большем верхней уставки
		002	Реле замыкается при значении контролируемой величины, находящемся между нижней и верхней уставками
		003	Реле замыкается при выходе значения контролируемой величины за пределы, заданные верхней и нижней уставками

C06	Блокировка срабатывания компаратора до 1-го достижения уставки в начале программы	000	Откл.
		001	Вкл.

C07	Блокировка срабатывания компаратора до 1-го достижения уставки в начале шага	000	Откл.
		001	Вкл.

C08	Блокировка срабатывания компаратора до нажатия кнопки «СБРОС»	000	Откл.
		001	Вкл.

Обознач.	Название	Допустимые значения	Комментарии
$c09$	Время задержки срабатывания компаратора	000...999 [с]	
$c10$	Время задержки отпускания компаратора	000...999 [с]	
<b>Параметры шагов</b>			
$n01...n08$	Режимы 1-го...8-го транзисторных ключей на данном шаге	000 001 002...015	Ключ не замкнут (транзистор закрыт) Ключ замкнут непрерывно (транзистор открыт) 1...14-й импульсные режимы работы ключа, задаваемые в параметрах F02... F15 (F.02... F.15)
$y01$	Условие перехода к следующему шагу	000 001 002 003	По достижении установленного в параметре Y02 значения температуры или влажности По достижении установленного в параметрах Y04 и Y05 времени По выполнении условий 000 и 001 По выполнении условий 000 или 001
$y02$	Условия перехода к следующему шагу по температурам или влажности	001 (002) 003 (004) 005 (006) 007 (008) 009 (010)	$T_{\text{прод.}} > T_{\text{уст.прод.}} (T_{\text{прод.}} < T_{\text{уст.прод.}})$ , [град.] $T_{\text{сух.}} > T_{\text{уст.сух.}} (T_{\text{сух.}} < T_{\text{уст.сух.}})$ , [град.] $T_{\text{влаж.}} > T_{\text{уст.влаж.}} (T_{\text{влаж.}} < T_{\text{уст.влаж.}})$ , [град.] $\Delta T > \Delta T_{\text{уст.}} (\Delta T < \Delta T_{\text{уст.}})$ , [град.] $\Psi > \Psi_{\text{уст.}} (\Psi < \Psi_{\text{уст.}})$ , [%]
$y03$	Уставка условия перехода к следующему шагу по температурам или влажности	-99...+999	в диапазоне измерения датчиков
$y04$	Длительность шага	0...63 [ч]	
$y05$	Длительность шага	0...59 [мин]	
<b>Параметры 1-го (2-го) регуляторов на данном шаге</b>			
$E01 (E.01)$	Входная величина регулятора	001 002 003 004 005	Температура продукта, [град.] Температура сухого термометра в камере, [град.] Температура влажного термометра, [град.] $\Delta T = T_{\text{сух.}} - T_{\text{прод.}}$ , [град.] Относительная влажность $\Psi$ , [%]
$E02 (E.02)$	Уставка входной величины (целая часть) – XXX.	-99...+999 0...99	[град.], пользователь задает только целую часть XXX. уставки XXX.X [%]
$E03 (E.03)$	Уставка входной величины (дробная часть) – .00X	00.0...00.9 00.1...00.9	[град.], пользователь задает только дробную часть .X уставки XXX.X [%]
$E04 (E.04)$	Скорость выхода на уставку	00.0...99.9	[град./мин]; [%/мин]
$E05 (E.05)$	Знак скорости выхода на уставку	000 001	Положительный (рост входной величины) Отрицательный (снижение входной величины)

Обознач.	Название	Допустимые значения	Комментарии	
► Уровень L2. Общие параметры				
Sc	Параметр секретности	001	Вход в уровень только через пароль	
		002	Вход в уровень по паролю для записи без установки пароля для чтения	
		003	Вход в уровень без установки пароля для чтения и записи	
Параметры импульсных режимов транзисторных ключей				
F02 ... F15	Длительность импульса 1...14-го режимов	000...999	[с]	
F.02... F.15	Длительность паузы между импульсами 1...14-го режимов	000...999	[с]	
Коррекция показаний термометров				
Сдвиг характеристики датчика				
1c1	для термометра продукта	000...51.1	Прибавляется к измеренному значению, [ед. измер.]	
2c1	для сухого термометра			
3c1	для влажного термометра			
Знак сдвига характеристики				
1c2	для термометра продукта	000	Положительный	
2c2	для сухого термометра	001	Отрицательный	
3c2	для влажного термометра			
Изменение наклона характеристики датчика				
1c3	для термометра продукта	000...25.5	Умножается на измеренное значение, [% от измер. велич.]	
2c3	для сухого термометра			
3c3	для влажного термометра			
Знак наклона характеристики				
1c4	для термометра продукта	000	Положительный	
2c4	для сухого термометра	001	Отрицательный	
3c4	для влажного термометра			
Параметры цифровых фильтров				
d01... d03	Постоянные времени цифровых фильтров влажного, сухого термометров и термометра продукта, соответственно	000...064	[с]	
d04	Постоянная времени фильтра относительной влажности	000...064	[с]	
d05, d06	Постоянные времени фильтров датчиков положения задвижек 1 и 2	000...064	[с]	
Основные параметры работы прибора				
o01	Тип температурных датчиков, подключаемых к прибору	000	ТСМ [50М/100М]	W <sub>100</sub> = 1,426
		001	ТСП [50П/100П]	W <sub>100</sub> = 1,385
		002	ТСП [50П/100П]	W <sub>100</sub> = 1,391
		003	ТСМ [50М/100М]	W <sub>100</sub> = 1,428

o 11	Тип переключения каналов индикации	000	Автоматическое переключение
		001	Ручное переключение

#### Параметры 1-го (2-го) ПИД-регуляторов

<i>PD1 (P.D1)</i>	Постоянная времени дифференцирования	000...999	[с]
<i>PD2 (P.D2)</i>	Постоянная времени интегрирования	000...999	[с]
<i>PD3 (P.D3)</i>	Полоса пропорциональности	000...999	[град]
		0...99	[%]
<i>PD4 (P.D4)</i>	Период следования выходных импульсов	000...120	[с]
<i>PD5 (P.D5)</i>	Зона действия интегральной составляющей	000...999	[град]
<i>PD6 (P.D6)</i>	Ограничение максимальной мощности	000...99	[%]
<i>PD7 (P.D7)</i>	Тип исполнительного устройства	000	Коммутирует свои реле: одно как «нагреватель», на выходе регулятора другое как «холодильник»
		001	Регулятор отключен
		002	Коммутирует свое реле как «нагреватель»
		003	Коммутирует свое реле как «холодильник»
		004	Управляет задвижкой по прямо пропорциональному закону
		005	Управляет задвижкой по обратно пропорциональному закону
		006	Управляет задвижкой по прямо пропорциональному закону с обратной связью по положению
		007	Управляет задвижкой по обратно пропорциональному закону с обратной связью по положению
<i>PD8 (P.D8)</i>	Зона нечувствительности	000...099	[град]
		0...99	[%]

#### ► Уровень L3. Самонастройки 1-го (2-го) ПИД-регуляторов

<i>SLF (SLF.)</i>	Входная величина, для которой производится самонастройка	001	Температура $T_{\text{прод.}}$
		002	Вход $T_{\text{сух.}}$
		003	Вход $T_{\text{алек.}}$
		004	$\Delta T = T_{\text{сух.}} - T_{\text{прод.}}$
		005	Относительная влажность $\Psi$

#### ► Уровень L4. Калибровки датчиков

<i>CAL</i>	Калибровки входов термодатчиков и датчиков положения	001	Калибровка $T_{\text{сух.}}$ (и $T_{\text{прод.}}$ )
		002	Калибровка $T_{\text{алек.}}$
		003	Калибровка входа КЗР1 при полностью закрытой задвижке 1
		004	Калибровка входа КЗР1 при полностью открытой задвижке 2
		005	Калибровка входа КЗР2 при полностью открытой задвижке 2
		006	Калибровка входа КЗР2 при полностью закрытой задвижке 2

## Лабораторна робота № 3

### Дослідження SCADA-систем.

**Мета роботи:** ознайомитися з програмним комплексом CoDeSys, здійснити конфігурування ПЛК, навчитися складати програми мовою LD-програм для заданого технологічного процесу, отримати навички роботи в режимі емуляції, навчитися завантажувати програму в ПЛК, дослідити роботу ПЛК під управлінням складеної програми.

#### **Вступ.**

Програмовані логічні контролери (ПЛК), програмування яких здійснюється з вбудованого або зовнішнього пульта, зустрічаються сьогодні досить рідко. Як правило, це прості, спеціалізовані ПЛК. Усе програмування таких контролерів зводиться звичайно до завдання набору констант.

Для програмування ПЛК універсального призначення застосовують персональний комп'ютер. Процес розробки й налагодження програмного забезпечення відбувається за допомогою спеціалізованих комплексів програм, що забезпечують комфортне середовище для роботи прикладного програміста [1.2].

Традиційно всі провідні виробники ПЛК мають власні фірмові наробітки в області інструментального програмного забезпечення.

Однак відкритість стандарту МЭК 61131 – з одного боку, і складність реалізації висококласних комплексів програмування – з іншого, призвели до появи універсальних інструментів програмування ПЛК.

Одним із самих популярних у світі універсальних комплексів програмування є комплекс CoDeSys, який підтримує стандарт МЭК і враховує фірмові особливості більш 150 моделей ПЛК.

Мета лабораторної роботи – вивчення основних прийомів роботи в програмному середовищі CoDeSys і придбання практичних навичок програмування контролера моделі ПЛК150 фірми OVEN мовою РКС (LD).

### **1. Основні відомості про програмний комплекс CoDeSys.**

#### **1.1 Призначення й можливості CoDeSys.**

Комплекс CoDeSys (Controllers Development System) фірми 3S (Smart Software Solutions) представляє проектувальнику зручне середовище для програмування контролерів на мовах МЭК [3 - 5]. Використані редактори й налагоджувальні засоби базуються на широко відомих принципах.

CoDeSys дозволяє використовувати мови: IL, ST, LD, SFC, FBD і CFC. В даній роботі будемо використовувати тільки графічну мову LD.

Текстові редактори CoDeSys роблять автоматичне оголошення змінних, тип яких задається в діалоговому вікні, і інші дії.

Графічний редактор автоматично виконує розміщення компонентів схеми (контактів, котушок реле, таймерів та ін.) і трасування їх з'єднань; нумерацію ланцюгів; масштабування зображення, що дозволяє побачити всю LD - діаграму (програму) або якусь її частину й виділяти кольором активні ланцюги.

Вбудовані емулятор і елементи візуалізації дають можливість виконувати налагодження проекту без самих апаратних засобів.

#### **1.2 Компоненти організації програм (POU).**

Компоненти створюють під прикладне програмне забезпечення ПЛК. Компоненти організації програм **POU (Program Organization Unit)** містять функції, функціональні блоки й програми. Компонент виступає як «чорний ящик», внутрішній пристрій і зміст якого знати не потрібно. У графічному зображенні він представлений прямокутником з входами (ліворуч) і виходами (праворуч).

Вибір потрібного POU проводиться у вікні оголошень (Рис. 1) в рядках **Програма**, **Функціональний блок** або **Функція**. Для LD будемо використовувати тільки **Програма**, тому що нам будуть потрібні тільки стандартні компоненти (контакти, котушки реле, FB).

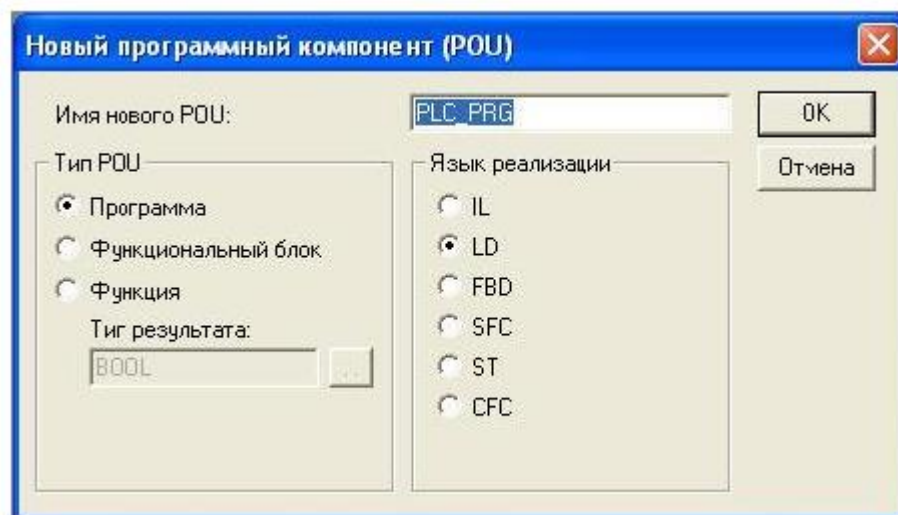


Рис. 1 – Вибір мови програмування й завдання імені програми.

### 1.3 Запуск CoDeSys.

Зробимо перший запуск середовища CoDeSys. У вікні **Target Settings** напроти рядка **Configuration** вибираємо тип логічного контролера **PLC 150.I-L**, оскільки саме він використовується в лабораторній роботі, і натискаємо **ОК**. У вікні, що з'явилося, **Новий програмний компонент (POU)** (Рис. 1) вибираємо тип POU - **Програма** й мова, на якій буде здійснюватися написання програми - **LD**. Ім'я програми залишаємо без зміни. Підтверджуємо вибір натисканням на кнопку **ОК**. Після виконання всіх вищеповисаних дій відкриється головне вікно (Рис. 2) середовища CoDeSys.

Його можна розділити на різні області (у вікні вони розташовані зверху вниз):

- меню (Рис. 3);
- панель інструментів, яка містить кнопки для швидкого виклику команд меню (Рис. 4);
- організатор об'єктів, що має вкладки «POU», «Типи даних», «Візуалізації» і «Ресурси»;

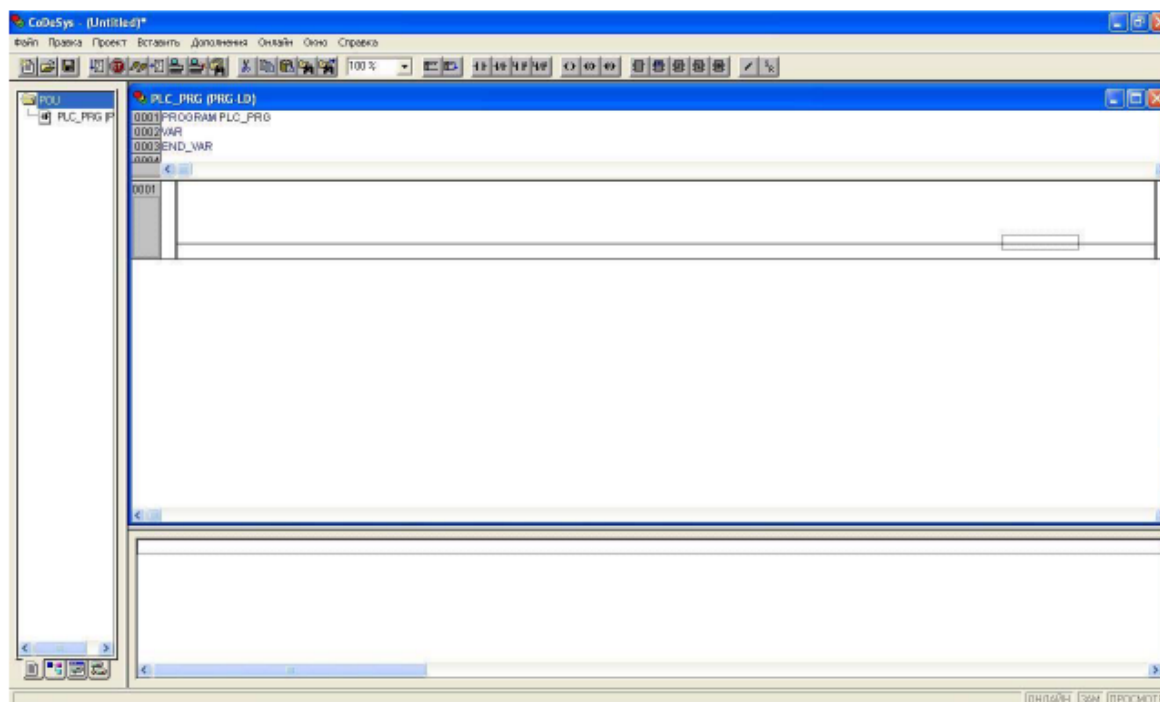


Рис.2. Головне меню CoDeSys.

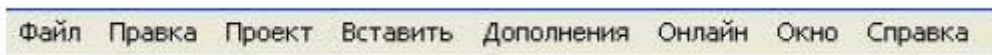


Рис. 3 – Меню середовища.



Рис. 4 – Панель інструментів

- роздільник організатора об'єктів і робочої області CoDeSys;
- робоча область, у якій перебуває редактор;
- вікно повідомлень;
- рядок статусу, що містить інформацію про поточний стан проекту.

Меню перебуває у верхній частині головного вікна. Воно містить усі команди CoDeSys.

Кнопки на панелі інструментів забезпечують більш швидкий доступ до командам меню. Викликана за допомогою кнопки на панелі інструментів команда автоматично виконується в активному вікні. Команда виконається, як тільки натиснута на панелі інструментів кнопка буде відпущена. Якщо ви помістите вказівник мишки на кнопку панелі інструментів, то через невеликий проміжок часу побачите назву цієї кнопки в підказці. Кнопки на панелі інструментів різні для різних редакторів CoDeSys. Одержати інформацію щодо призначення цих кнопок можна в описі редакторів. Кнопки на панелі інструментів дуже важливі, і їх використання спрощує складання програми мовою LD. В інших мовах ця панель виглядає інакше. Призначення кнопок на панелі інструментів наведені в таблиці 1.

При бажанні панель інструментів можна відключити (у меню - «**Проект**», у списку, що випадає, - «**Опції...**», категорія «**Робочий стіл**», прибрати галочку «**Панель інструментів**»).

Організатор об'єктів (Рис. 5) завжди перебуває в лівій частині головного вікна CoDeSys. У нижній частині організатора об'єктів перебувають вкладки «**POU**», «**Типи даних**», «**Візуалізації**» і «**Ресурси**». Перемикатися між відповідними об'єктами можна за допомогою мишки, натискаючи на потрібну вкладку.

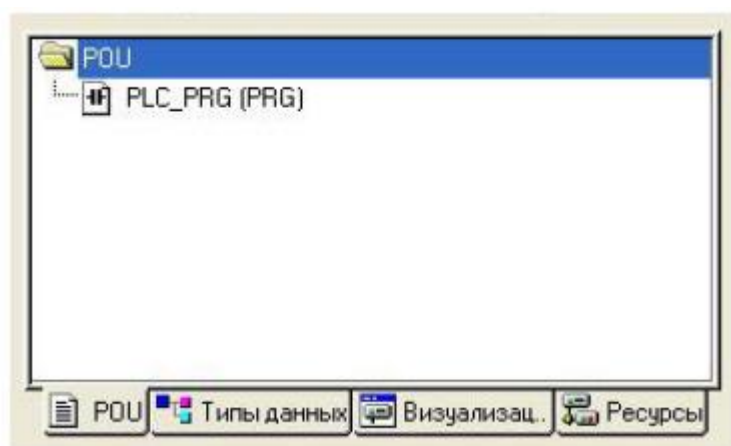


Рис. 5 – Організатор об'єктів



Таблица 1 – Кнопки на панели инструментов

Графическое изображение кнопки	Название	Назначение	Графическое изображение кнопки	Название	Назначение
	Создать	Создает новый проект		Цепь (вперед)	Вставляет цепь перед текущей
	Открыть	Открывает проект		Цепь (назад)	Вставляет цепь после текущей
	Сохранить	Сохраняет содержимое измененного объекта		Контакт	Вставляет последовательный замыкающий контакт
	Старт	Запускает ПЛК		Инверсный контакт	Вставляет последовательный инверсный (размыкающий) контакт
	Стоп	Останавливает ПЛК		Параллельный контакт	Вставляет параллельный замыкающий контакт
	Шаг по верху	Перешагивает через текущую инструкцию, даже если это вызов подпрограммы		Параллельный инверсный (размыкающий) контакт	Вставляет параллельный инверсный (размыкающий) контакт
	Переключить точку останова	Устанавливает/убирает точку останова в текущей позиции		Обмотка	Вставляет обмотку (катушку) «Реле»
	Подключение	Устанавливает связь с ПЛК и включает режим On-Line		«Set» обмотка	Вставляет Set обмотку (катушку)
	Отключение	Отключает режим On-Line		Reset обмотка	Вставляет Reset обмотку (катушку)
	Глобальный поиск	Ищет заданную строку по всему проекту		Функциональный блок	Вставляет функциональный блок
	Вырезать	Перемещает выделенную область в буфер обмена		Элемент с «EN»	Вставляет элемент со входом разрешения
	Копировать	Копирует выделенную область в буфер обмена		Детектор переднего фронта	Вставляет детектор переднего фронта
	Вставить	Вставляет содержимое буфера обмена в текущую позицию		Детектор заднего фронта	Вставляет детектор заднего фронта
	Найти	Ищет заданную строку в текущем окне		Таймер «TON»	Вставляет таймер «TON»
	Найти далее	Повторяет последний поиск		Инверсия	Инвертирует выбранный выход или вход
	Масштаб	Увеличение/уменьшение масштаба		Установка/сброс	Преобразует выход в Set/ Reset выход

#### 1.4 Роздільник екрана.

Розділити екрана - це межа між двома вікнами. В CoDeSys є наступні роздільники: між організатором об'єктів і робочою областю, між розділом оголошень і розділом коду POU, між робочою областю й вікном повідомлень. Ви можете переміщати роздільники за допомогою мишки, нажавши її ліву кнопку.

Роздільник зберігає своє положення навіть при зміні розмірів вікна. Якщо ви більше не бачите роздільника на екрані, виходить, варто змінити розміри вікна.

Робоча область (Рис. 2) перебуває в правій частині головного вікна CoDeSys. Усі редактори, а також менеджер бібліотек відкриваються саме в цій області. Ім'я відкритого об'єкта перебуває в заголовку вікна.

#### 1.5 Вікно повідомлень.

Вікно повідомлень відділене від робочої області роздільником. Саме в цьому вікні з'являються повідомлення компілятора, результати пошуку й список перехресних посилань.

При подвійному клацанні лівою клавішею миші або при натисканні клавіші **Enter** на повідомлення буде відкритий об'єкт, до якого відноситься дане повідомлення.

Далі скорочено операції із кнопками миші будемо записувати так:

1ЛКМ, якщо одне натискання на ліву клавішу мишки, 2ЛКМ - якщо два натискання;

1 ПКМ якщо одне клацання правою кнопкою.

#### 1.6 Статусний рядок.

Статусний рядок знаходиться в нижній частині головного вікна CoDeSys і надає інформацію про проект і команди меню.

Якщо ви помістили вказівник на пункт меню, то в рядку статусу з'являється його короткий опис.

Якщо працюєте в режимі **Онлайн**, то напис **Онлайн** у рядку статусу виділяється чорним кольором. В іншому випадку напис сірий. За допомогою статусного рядка в режимі **Онлайн** можна визначити, у якому стані перебуває програма. Наприклад, «Емул.» - у режимі емуляції.

Статусний рядок можна прибрати або включити (у меню «Проект», «Опції...», «Робочий стіл»).

#### 1.7 Контекстне меню.

Альтернативою використання головного меню для виклику команд може стати контекстне меню (Рис. 6). Це меню, викликається 1ПКМ на робочій області, містить найбільш часто використовувані команди.

Вырезать	Ctrl+X
Копировать	Ctrl+C
Вставить	Ctrl+V
Удалить	Del
Цель (перед)	
Цель (после)	Ctrl+T
Контакт	Ctrl+K
Инверсный контакт	Ctrl+G
Параллельный контакт	Ctrl+R
Параллельный контакт (инверсный)	Ctrl+D
Функциональный блок...	Ctrl+B
Детектор переднего фронта	
Детектор заднего фронта	
Таймер (TON)	
Обмотка	Ctrl+L
'Set' обмотка	Ctrl+I
'Reset' обмотка	
Элемент с EN	
Вставка в блоки	
Переход	
Возврат	
Комментарий	
Инверсия	Ctrl+N
Set/Reset	
Масштаб	Alt+Enter
Открыть экземпляр	

Рис. 6 – Контекстне меню програми CoDeSys.

## 2. Розробка програми керування для ПЛК мовою LD.

### 2.1 Мова сходових діаграм LD.

Мова сходових діаграм LD (Ladder Diagram) або релейно-контактних схем (РКС) - графічна мова, що реалізує структури електричних ланцюгів [1, 2]. На початку 70-х рр. XX століття пристрої керування, побудовані на релейних елементах, почали поступово витіснятися програмованими контролерами. Якийсь час ті й інші працювали одночасно й обслуговувалися тим самим персоналом.

Так з'явилося завдання прозорого переносу релейних електричних схем у ПЛК. Різні варіанти програмної реалізації релейних схем створювалися практично всіма провідними виробниками ПЛК. Завдяки простоті представлення, мова LD знайшла заслужену популярність, що й стало основною причиною включення її в стандарт МЭК.

Графічно LD -діаграма (програма мовою РКС) представлена в виді двох вертикальних шин живлення. Між ними розташовані ланцюги, утворені з'єднанням контактів. Навантаженням кожного ланцюга звичайно служить реле (обмотка). Кожне реле має контакти, які можна використовувати в інших ланцюгах. Кількість контактів у ланцюзі довільна, реле одне. Якщо послідовно з'єднані контакти замкнені, струм іде по ланцюгу й реле включається. При необхідності можна включити паралельно декілька реле, послідовне включення реле не допускається.

Важливо розуміти, що контакти, обмотки реле, ланцюги, шини живлення та ін. в LD-діаграмі це не фізичні пристрої, а елементи, які реалізовані **програмно**.

В LD кожному контакту ставиться у відповідність логічна змінна, яка визначає його стан. Якщо контакт замкнутий, то змінна має значення **ІСТИНА** (логічна 1). Якщо розімкнений — **НЕПРАВДА** (логічний 0). Ім'я змінної пишеться над контактом і фактично є його назвою.

Послідовне з'єднання контактів або ланцюгів рівноцінно логічній операції **І**. Паралельне з'єднання утворює монтажне **АБО**.

Ланцюг може бути або замкнено (ON), або розімкнено (OFF). Це як раз і відображується на обмотці реле й відповідно на значенні логічної змінної обмотки (ІСТИНА/НЕПРАВДА або TRUE/FALSE).


Контакт може бути інверсним — нормально замкненим. Такий контакт позначається за допомогою символу  $\neg$  і замикається, якщо значення змінної є НЕПРАВДА. Інверсний контакт рівнозначний логічній операції **НІ**.

Обмотки реле також можуть бути інверсними, що позначається символом  $\neg$ . Якщо обмотка інверсна, то у відповідну логічну змінну копіюється інверсне значення стану ланцюга.

Логічно послідовне (І), паралельне (АБО) з'єднання контактів і інверсія (НІ) утворюють базис алгебри Буля. У результаті LD ідеально підходить для програмної реалізації комбінаційних логічних систем керування. Завдяки можливості включення в LD -програми функцій і функціональних блоків, виконаних на інших мовах, сфера застосування мови практично не обмежена.

### 2.2 Основні елементи програми.

Після відкриття головного вікна CoDeSys на моніторі з'являється робоча область (Рис. 2), у якій і будемо зображувати багатоступінчасту схему (сходову діаграму).

Ця схема являє собою набір горизонтальних ланцюгів, які нагадують, сходять, з'єднують вертикальні шини, живлення. Якщо необхідно збільшити розмір робочої області, то підводимо курсор до кнопки  у верхньому правому куті цієї області й натискаємо ІЛКМ.

Перший ланцюг з'являється в робочій області відразу (Рис. 2). Ліворуч на сірому фоні автоматично виникне її номер: 0001. Наявність пунктирного прямокутника в правій частині ланцюга свідчить про те, що вона активована, тобто готова приймати внесені в неї компоненти: контакти, функціональні блоки FB, котушки реле.

Будемо вважати, що сама система логіко-програмного керування дискретним процесом у звичайному, тобто релейно-контактному виконанні, вже є, і наше завдання перенести її в LD - діаграму. У якості такої системи керування оберемо схему, представлену на рис. 7.

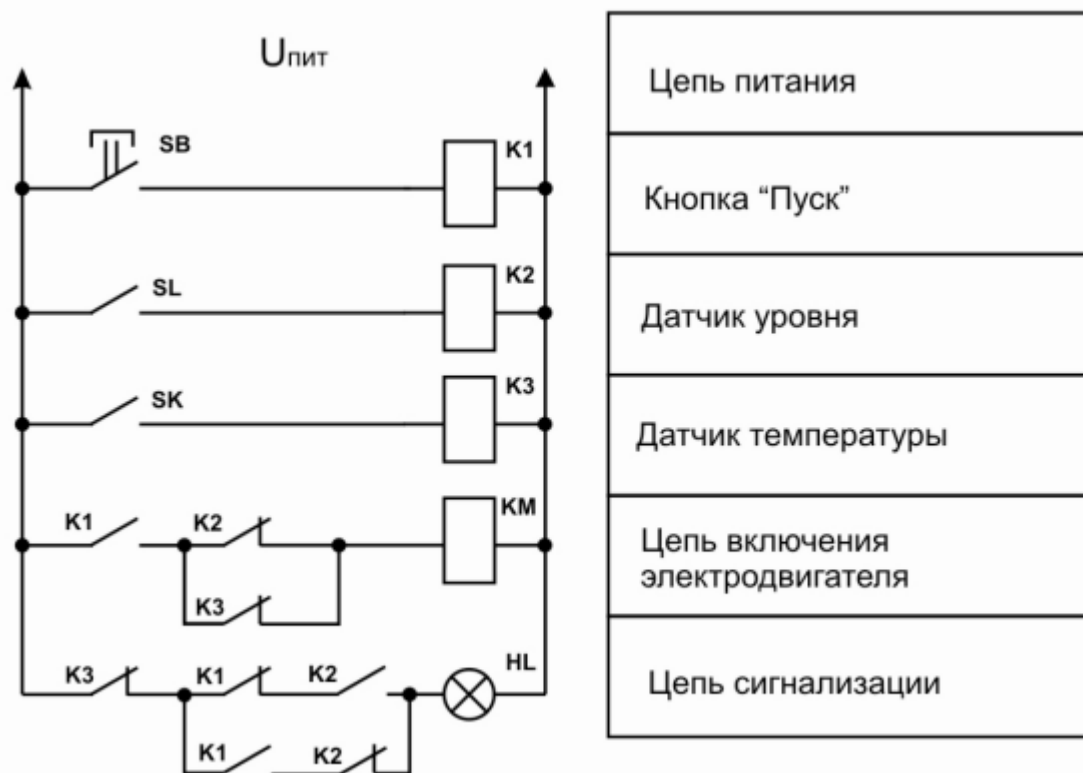


Рис. 7 – Релейно-контактна схема системи логіко-програмного керування дискретним процесом.

До складу схеми входять:

- 1 Три джерела вхідних дискретних сигналів (кнопка керування «Пуск» SB, два електроконтактні дискретні (не аналогових) датчика – датчик температури SK і рівня SL).
- 2 Три проміжні реле K1, K2, K3.
- 3 Два вихідні пристрої – обмотка магнітного пускача KM і сигнальна лампа HL.

### 2.3 Методика програмування мовою LD у середовищі CoDeSys.

За аналогією з вихідним варіантом (Рис. 7) у багатоступінчастій LD - програмі, швидше за все, також буде потрібно п'ять ланцюгів

У перший ланцюг необхідно внести контакт SB і котушку реле K1.

1 Наводимо курсор на кнопку в панелі інструментів із зображенням замикаючого контакту, натискаємо ІЛКМ, і цей контакти з'являється в ланцюзі с трьома знаками питання червоного кольору (Рис. 8). Ці ??? запитують ім'я або ідентифікатор внесеного в ланцюг компонента.

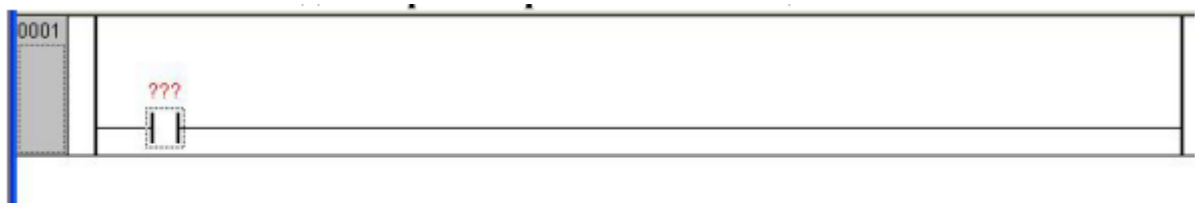


Рис. 8 – Перший ланцюг з уведенням контактом.

2 Наводимо курсор на ???, клацаємо 1ЛКМ. Питання стають білими на фоні синього прямокутника. За допомогою клавіатури англійською мовою задаємо ім'я. У нашому прикладі «SB». Букви російського алфавіту використовувати не можна!

3 Натискаємо «Enter». Відкривається вікно **Оголошення змінної**, запитує, до якого класу змінних буде віднесений наш елемент (Рис. 9).

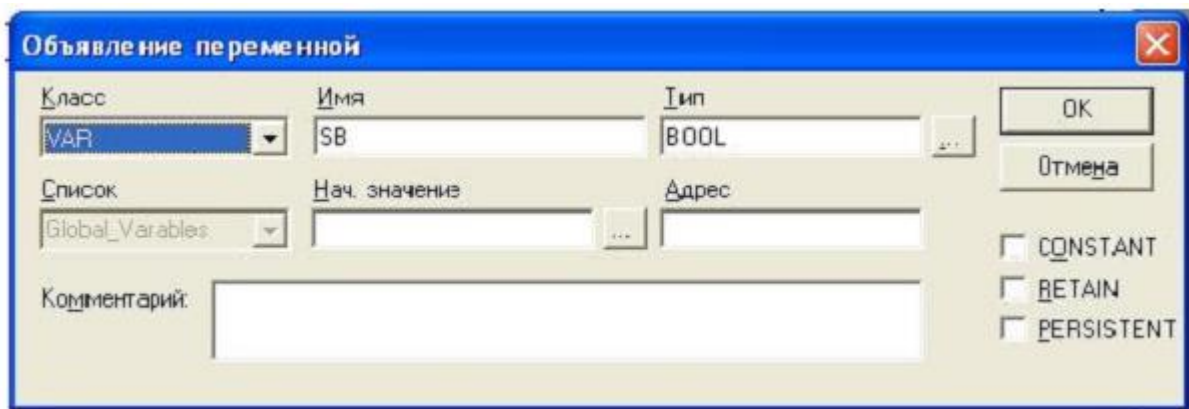


Рис. 9 – Вікно оголошення змінної.

Якщо проектувана схема призначена для навчальних цілей і буде працювати тільки в режимі емуляції, то можна відразу натиснути 1ЛКМ на ОК, і ім'я з'явиться над елементом (Рис. 10).

Потрібно звернути увагу на пунктирний квадратик, що охоплює контакт **SB**, і на зникнення пунктирного прямокутника наприкінці самого ланцюга. Це свідчить про те, що активований сам контакти **SB**, тобто можна (якщо була б така необхідність) послідовно й/або паралельно цьому контакту (як буде показано при створенні четвертого й п'ятого ланцюгів) підключати до нього інші контакти.

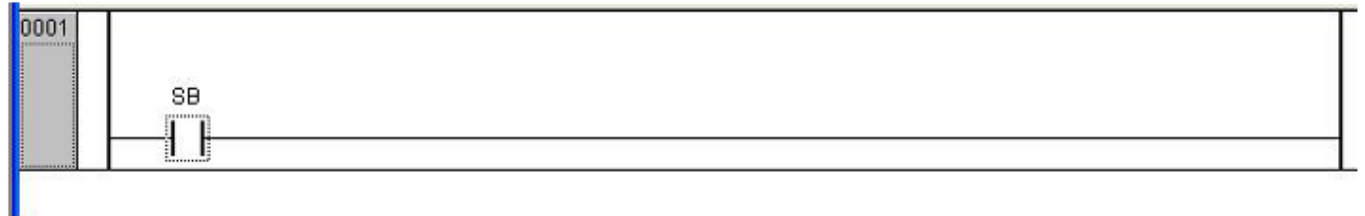




Рис. 10 – Оголошений замикаючий контакт.

Для завершення першого ланцюга необхідно ввести в нього котушку реле K1.

4 Наводимо курсор на , клацаємо 1ЛКМ і ... нічого не вийшло. Треба спочатку активувати ланцюг. Для цього наводимо курсор на лінію ланцюга, клацаємо 1ЛКМ, тобто активуємо її (з'явився пунктирний прямокутник!), переводимо курсор на , клацаємо 1ЛКМ і в ланцюзі з'являється котушка із трьома білими ??? у синьому прямокутнику.

Котушки та, як буде показано нижче, функціональні блоки FV з'являються завжди з такими ???. Якщо ім'я відразу не привласнити й перейти до інших дій, то ??? стануть червоними, як і у випадку включення контактів. Потім буде потрібна зайва операція при ідентифікації котушки: знову навести курсор на червоні ???, клацнути 1ЛКМ, ??? стають білими на синьому фоні і т.д. Але це не принципово.

5 Привласнюємо котушці ім'я: **K1**. Натискаємо **Enter**, відкривається вікно **Оголошення змінної**, натискаємо 1ЛКМ на **ОК**.

Можна спочатку включити в ланцюг котушку, потім контакт. Можна включити ці елементи в будь-якій послідовності й лише потім привласнити їм імена. Результат буде тим же.

Перший ланцюг завершений (Рис. 11). Як видно на моніторі, зона ланцюга обмежена двома горизонтальними лініями. При активації ланцюга необов'язково наводити курсор точно на ланцюг. Досить потрапити в цю зону за виключенням смуги, пов'язаною з верхньою лінією, що і є як би продовженням рядка, що починається з номера ланцюга. У нашому випадку з 0001. Ця смуга може бути використана для завдань англійською мовою адрес міток переходів.

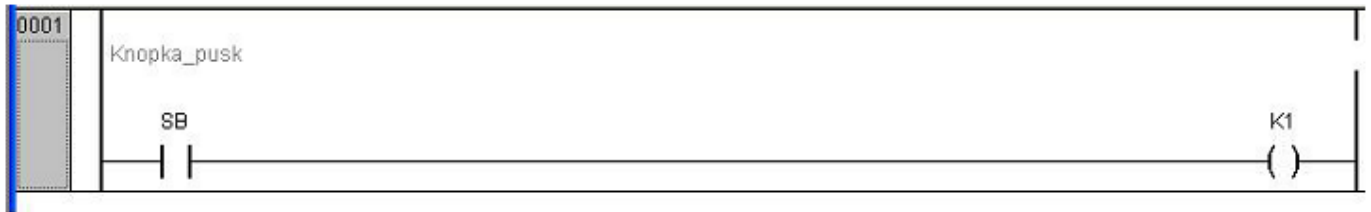


Рис. 11 – Завершений перший ланцюг.

Для позначення функціонального призначення ланцюги в LD –діаграмах часто задають коментарі на будь-якій мові, зрозумілій обслуговуючому персоналу.

Із цією метою наводимо вказівник на смужку, розташовану нижче смужки міток і клацаємо ІЛКМ, з'являється миготливий курсор, із клавіатури задаємо, наприклад, «Knopka\_puska» і натискаємо Enter. Коментарі можна виконати й російською мовою. Із цією метою наводимо курсор на поле ланцюга, клацаємо ІПКМ. Відкривається контекстне меню (Рис.

б). Клацаємо ІЛКМ на рядку **Коментар**. Ліворуч над ланцюгом з'явиться слово «**Comment**», у рядку з яким можна написати на будь-якій мові необхідні пояснення. Слово «**Comment**» потім можна видалити.

Перед тем як перейти до наступного ланцюга, обговоримо ще деякі важливі моменти.

Якщо реальне реле має обмежену кількість замикаючих, розмикальних і перемикаючих контактів, то в LD таких обмежень немає, і віртуальні контакти можуть застосовуватися в будь-якому ланцюзі в будь-якій кількості.

У всіх ланцюгах однієї схеми імена логічної змінної контактів одного і того ж реле повинні зберігатися. Ім'я може бути однобуквеним (X, Y, Z і т. д.), мати цифрові індекси (X1, X2 і т.д.), що вписуються без пробілу. Якщо є необхідність у такому пробілі, наприклад, при написанні двох або більше слів, то в пробіл ставиться символ підкреслення. Цей символ є значимим, тобто імен X\_1, X1, \_X1 і \_X\_1 сприймаються програмою як самостійні. Але цифру на перше місце ставити не можна: 1X, 2X - неправильно!

Не можна застосовувати в якості імен оператори інших мов. Наприклад, оператори мови інструкцій IL: LD, ST, S, R, AND, MUL, JMP і ін. З індексами, символами підкреслення або іншими буквами можна. Наприклад S1, RU, AND\_, MULTI і т.д. Навіть не знаючи весь список операторів, легко встановити помилку. Після натискання на клавішу Enter у процесі присвоєння імені змінної заборонений ідентифікатор висвітлиться синім кольором. Його необхідно видалити й вписати інше ім'я.

Регістр букв не впливає на роботу ПЛК. Так, імена «SET» і «Set» сприймаються однаково.



У складних схемах важко запам'ятати призначення того або іншого елемента при спрощеній (однобуквенній) системі ідентифікації. Тому ім'я змінної (тобто ідентифікатор) можна записати в розгорнутому виді, не використовуючи букви російської мови.

Наприклад, якщо є труднощі з англійською мовою, можна привласнити імена російською мовою «**Dvigatel**», «**pusk**», «**BLOKIROVKA**» і т.д.

Кожний ланцюг закінчується обмоткою (котушкою) реле. Послідовно з'єднувати котушки не можна. Паралельно - можна.

Створюємо другий ланцюг.



Є три способи: за допомогою меню «Вставка» контекстного меню (Рис. 6) і кнопками  та . Скористаємося третім способом як найбільш простим.

1 Наводимо курсор на , клацаємо 1ЛКМ. Відразу з'являється другий ланцюг (Рис. 12).

2 За вищеписаною методикою вносимо в ланцюг замикаючий контакт датчика **SL** і котушку **K2**. Якщо є необхідність, то можна вписати коментар: наприклад, «**Datchik\_urovnia**».

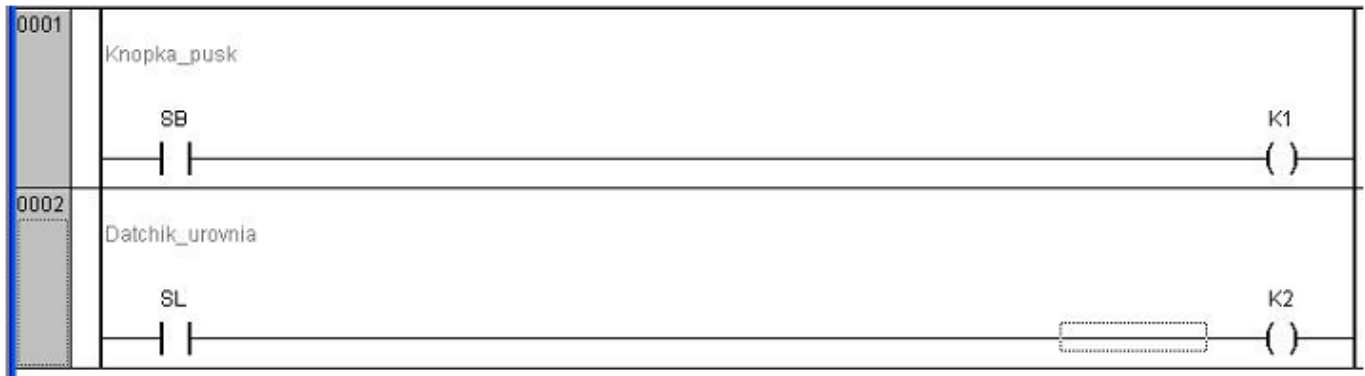






Рис. 12 – Другий ланцюг.

Якщо помилково або навмисно натиснули не , а , тобто **Ланцюг (вперед)**, то новий ланцюг стане поперед усіх створених і автоматично привласнить собі номер 0001, а наступним ланцюгам нумерацію побільшає на одиницю. Якщо ж проектувальник вирішив вставити новий ланцюг між двома вже створеними, то можна це виконати різними прийомами:

- Клацнути 1ЛКМ по ланцюгу 0001, потім по , тобто **Ланцюг (після)**. Новий ланцюг стане після ланцюга 0001, отримає номер 0002, а другий раніше створений ланцюг стане під номером 0003.


- Клацнути 1ЛКМ по ланцюгу 0002, потім по  й новий ланцюг з'явиться поперед ланцюга 0002, привласнить собі його номер, а раніше створеному ланцюгу надасть номер 0003.

Створюємо третій ланцюг, включаючи до його складу замикаючий контакт датчика температури **SK** та обмотки **K3**.

Трохи складніше буде програмування четвертого та п'ятого ланцюгів.

За відомою методикою створюємо четвертий ланцюг, послідовно вносячи в ланцюг замикаючий контакт **K1**, що розмикає **K2**. Тепер необхідно створити паралельний ланцюжок з розмикаючого контакту **K2** і розмикаючого **K3**. Для цього:

- Наводимо курсор на контакт **K1**, клацаємо 1ЛКМ. Контакт активований, про що свідчить пунктирний квадратик, що охоплює контакт.

- Наводимо курсор на , клацаємо 1ЛКМ і з'єднання виконане. Привласнюємо ім'я: **K3**.

Якщо активувати сам ланцюг, а не контакт K2, то замикаючий контакт K3 охопив би весь ланцюжок з контактів **K1** і **K2**, як показано на рис. 13.

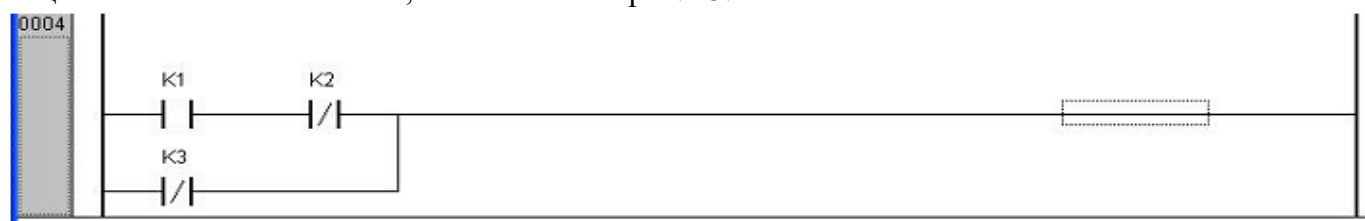



Рис. 13 – Фрагменти програми після невдалого введення контакту K3.

Вносимо в ланцюг котушку **КМ**. Четвертий ланцюг завершений.

Програмуємо п'ятий ланцюг LD -Програми.

За відомою методикою вносимо в ланцюг розмикаючі контакти **K3** і **K1**, замикаючий **K2** і обмотку (катушку) **HL**. Необхідності повідомляти змінні **K1**, **K2** і **K3** немає, тому що вони вже внесені в список булевих змінних. Залишилося створити паралельний ланцюжок із замикаючого контакту **K1** і розмикального **K2**. Для цього:

- Активуємо розмикаючий контакт **K1**, натискаємо і не відпускаємо «**Shift**», активуємо замикаючий контакт **K2**, відпускаємо «**Shift**», обидва контакти виявилися охопленими пунктирним прямокутником.

- Наводимо курсор на , клацаємо ЛКМ, і паралельно із цими контактами з'явиться —| / |-, якому привласнимо ім'я K1. Фрагмент отриманої схеми в п'ятому ланцюгу показано на рис. 14.

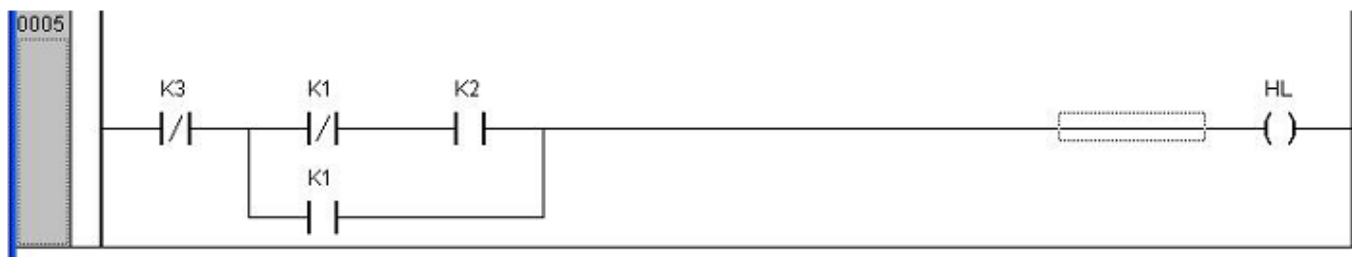




Рис. 14 – П'ятий ланцюг після введення замикаючого контакту K1 та його активації.

- Активуємо новий контакт, наводимо курсор на , клацаємо ЛКМ і послідовно із з'являється —| / |-. Привласнюємо ім'я **K2** новому елементу.

Паралельний ланцюжок виконаний, і завершена вся багатоступінчаста схема у вигляді LD -Програми (Рис. 15).

## 2.4 Додаткові приймання при розробці LD – програм.

Якщо помилково введений замість замикаючого контакту розмикаючий або навпаки, то можна навести на нього курсор, клацнути ПКМ і в контекстному меню (Рис. 6) клацнути ЛКМ на четвертому знизу рядку **Інверсія (Negate)**. Відбудеться інверсія цього елемента. Ім'я, якщо воно вже було привласнено, залишається колишнім.

Цю операцію можна виконати простіше, якщо є кнопка . (В деяких версіях CoDeSys вона може бути відсутньою). Тоді наводимо курсор на підлягаючий заміні контакт, клацаємо ЛКМ і натискаємо на зазначену кнопку.

Якщо необхідно вилучити якийсь компонент ланцюга або весь ланцюг з усіма елементами, то наводимо курсор на цей компонент (контакт, катушку, FB) або на сам ланцюг, клацаємо ПКМ і в контекстному меню натискаємо ЛКМ на самому верхньому рядку **Вирізати (Cut)**. Зникне елемент або весь ланцюг відповідно.

Якщо потрібно змінити ідентифікатор якогось компонента ланцюга, наводимо курсор на його ім'я, клацаємо ЛКМ, натисканням на клавішу **Backspace** видаляємо ім'я, задаємо нове, натискаємо **Enter**.

Якщо виникла необхідність перемістити якийсь компонент схеми по одному ланцюгу або навіть перекинути його в інший ланцюг, то захоплюємо цей елемент курсором і при натиснутій ЛКМ переставляємо його в нове місце. В момент переміщення на вході й виході кожного елемента відображаються невеликі прямокутники. Як тільки курсор підійде до одного з них, прямокутник стає зеленим. Відпускаємо ЛКМ і переміщуваний елемент займе нове місце.



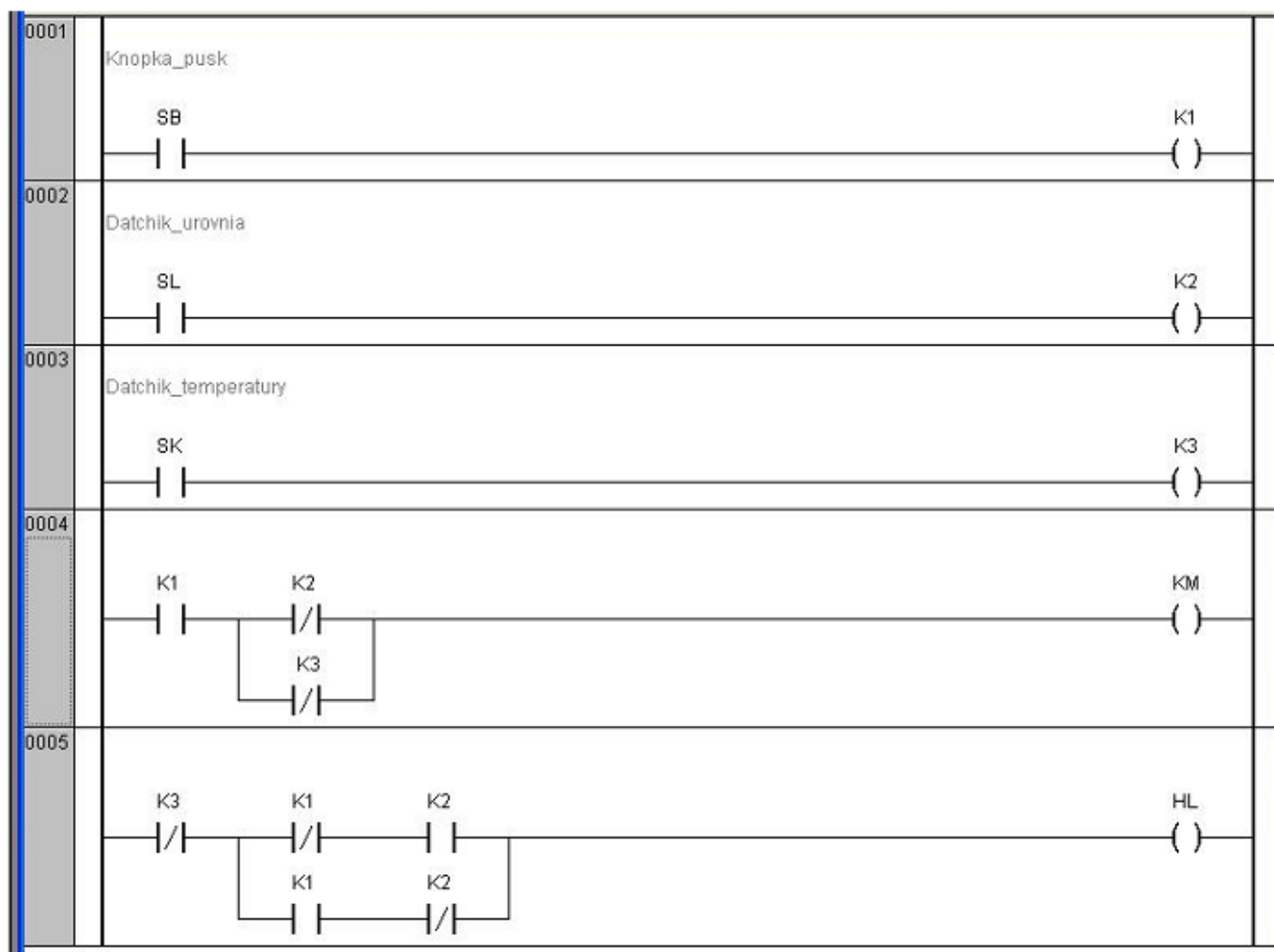


Рис. 15 – Система логіко-програмного керування дискретним процесом у вигляді програми мовою LD.

Якщо в міру збільшення кількості елементів у ланцюзі їм вже не вистачає місця в межах межує робочої області, то користуючись горизонтальним «прокручуванням» зміщаємо ланцюг вліво. Якщо в міру додавань ланцюгів їм уже ніде розміститися, то за допомогою вертикального «прокручування» піднімаємо всю багатоступінчасту схему або зменшуємо масштаб.

## 2.5 Завдання в програмі вихідних дискретних сигналів.

Крім «звичайних» реле —( )- можна застосовувати «інверсний» аналог, позначений у програмі —( / )-.

Це реле може мати скільки завгодно замикаючих і розмикаючих контактів, але логіка їх дії протилежна поведінці контактів звичайного реле: при відсутності струму в —( / )- замикаючий контакт —| | - замкнутий контакт, розмикаючий, —| / | - розімкнутий. При подачі живлення в котушку -( / )- стан її контактів змінюється на протилежне.

У наборі програмних компонентів є також спеціальні обмотки **SET** і **RESET**, позначені в лінійці кнопок як —(S)- і —(R)-. З їхньою допомогою можна фіксувати умови керування виконавчим механізмом.

Якщо обмотка S «спрацює», тобто прийме значення ІСТИНА (TRUE), то змінити цей стан на протилежний, тобто НЕПРАВДА (FALSE), можна лише за допомогою обмотки R (Рис. 16).

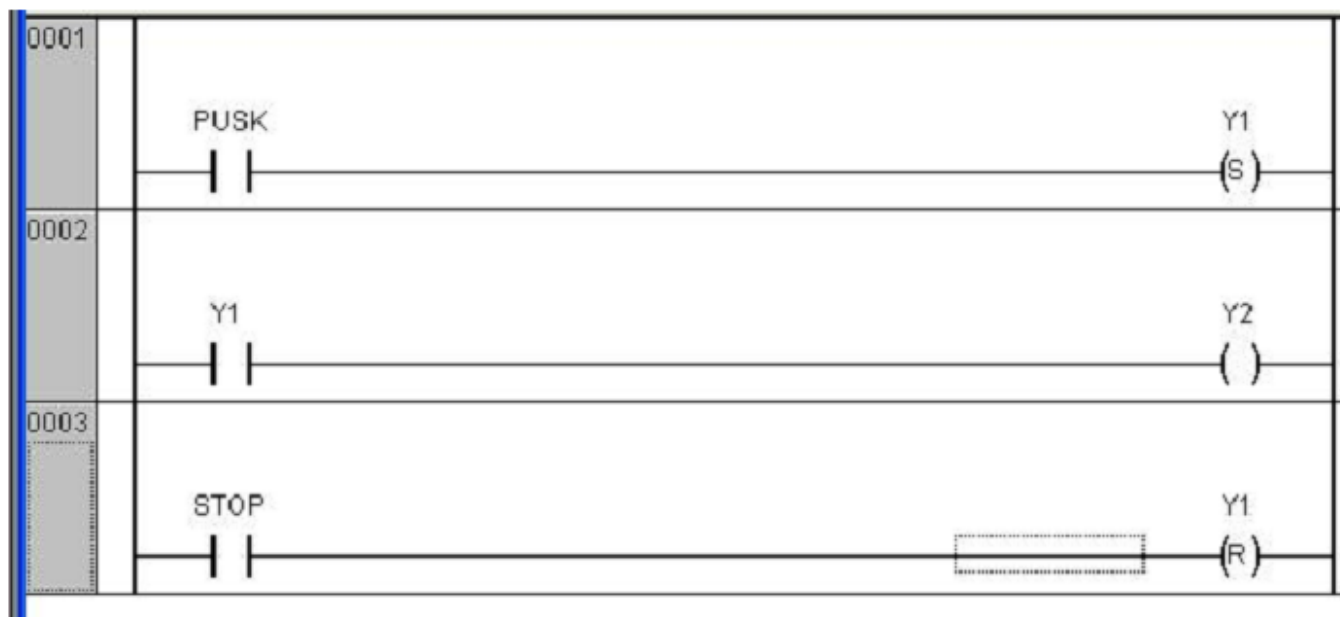


Рис. 16 – Ld-Діаграма фіксації котушки реле Y2 за допомогою обмоток S і R.

Ця схема працює як класичний Rs-тригер: при короткочасному натисканні кнопки **PUSK** спрацьовує котушка **S**, якій привласнимо ім'я **Y1**, і своїми контактами **Y1** включає авантаження – котушку реле **Y2**. Виключити реле **Y2** можна тільки натисканням кнопки **STOP**. Одночасне натискання на **PUSK** і **STOP**, як і в класичному Rs-тригері, неприпустимо. Слід помітити, що котушкам R і S привласнене те саме ім'я! В нашому прикладах Y1.

Це ж завдання самофіксації можна виконати й на звичайному реле (Рис. 17).

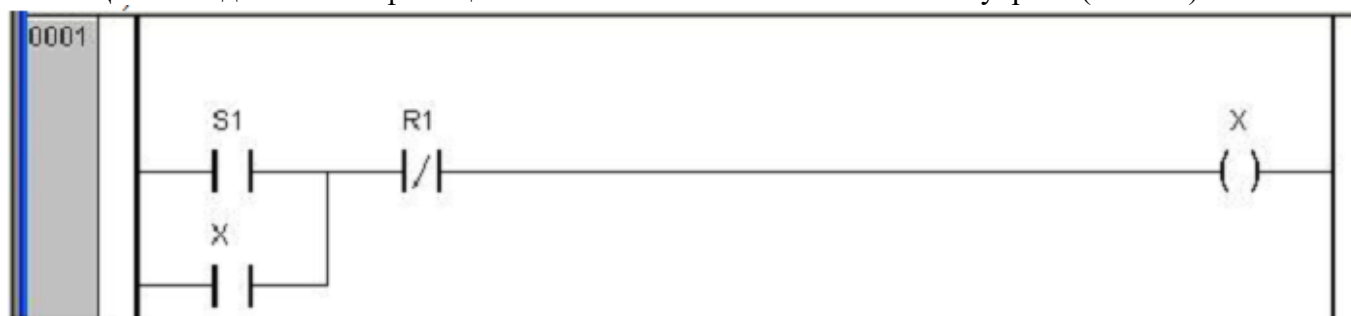


Рис. 17 – Схема керування котушкою X з самофіксацією стану.

При короткочасному натисканні на кнопку **S1** відбувається спрацьовування реле **X**, яке своїм контактом **X** фіксує цей стан. Відключення реле **X** можливо тільки натисканням на кнопку **R1**.

Зазначений прийом самофіксації (саможивлення) широко використовується на практиці при організації логіко-програмного керування дискретними виконавчими пристроями.

## 2.6 Дослідження LD -програми в режимі емуляції.

Отже, вихідну РКС (Рис. 7) запрограмували в CoDeSys мовою LD і представили її у вигляді багатоступінчастої схеми (Рис. 15). Тепер необхідно перевірити виконання запланованих умов спрацьовування й відсутність неправильних включень виконавчих елементів.

Усе це можна виконати в режимі емуляції, не використовуючи реальні апаратні засоби й сам ПЛК. Для цього:

- Наводимо курсор на **Онлайн (Online)** (Рис. 3), клацаємо 1ЛКМ. Відкривається меню (Рис. 18).

- Зміщаємо курсор до рядка **Режим емуляції (Simulation Mode)**, клацаємо 1ЛКМ. Перед цим рядком з'явиться «галочка» ✓, що свідчить про готовність програми до «діалогу». І надалі при роботі із цієї програмою навіть у випадку внесення в неї змін, доповнень повторювати цю операцію не треба.

- Знову відкриваємо це меню. Клацаємо 1ЛКМ по рядку **Підключення (Login)**. Ліва вертикальна «шина живлення» реле, що розмикають контакти, ділянки ланцюгів від цієї шини до якогось «непровідного» у цім положенні елемента ланцюга (наприклад, замикаючого контакту) зафарблюються в синій колір.

Можлива ситуація, коли після клацання 1ЛКМ по **Підключення** буде видане повідомлення про допущену помилку в програмі. Можливо, забули привласнити ім'я якомусь компоненту або зняти ??? на вході того або іншого функціонального блоку FB. У кожному разі, не зробивши необхідні корективи, рухатися далі не можна. Будемо вважати, що в нашій схемі немає помилок або ми їх уже усунули.

Подключение	Alt+F8
Отключение	Ctrl+F8
Загрузка	
Старт	F5
Стоп	Shift+F8
Сброс	
Сброс (холодный)	
Сброс (заводской)	
Переключить точку останова	F9
Диалог точек останова	
Шаг поверху	F10
Шаг детальный	F8
Один цикл	Ctrl+F5
Записать значения	Ctrl+F7
Фиксировать значения	F7
Освободить фиксацию	Shift+F7
Диалог Запись/Фиксация	Ctrl+Shift+F7
Показать стек вызовов...	
Отображать поток выполнения	
✓ Режим эмуляции	
Параметры связи...	
Загрузка исходных текстов	
Создание загрузочного проекта	
Записать файл в ПЛК	
Читать файл из ПЛК	

Рис. 18 – Меню **Онлайн**.

- Знову відкриваємо меню **Онлайн** і клацаємо 1ЛКМ по рядкові **Старт (Run)**. Програма готова до приймання вхідних сигналів. Можна приступитися до її дослідження.

- Почнемо з комбінації вхідних сигналів, рівних логічному нулю **SB** = 0, **SL** = 0, **SK** = 0, тобто **SB**, **SL** і **SK** не спрацювали. Відкриваємо меню **Онлайн** і клацаємо 1ЛКМ по рядкові **Записати значення (Write Values)**. Переконаємося, що K1, K2, K3, KM й HL не спрацювали, про що свідчить їх незмінний вигляд.

- Переходимо до другої комбінації. Повинен замкнутися контакт кнопки керування **SB** (**SB** = 1). Наводимо курсор на контакт **SB**, клацаємо 2ЛКМ. В порожнині цього контакту з'явиться синій квадратик. Відкриваємо меню **Онлайн** і клацаємо 1ЛКМ по рядкові **Записати значення**.

Контакт **SB**, котушка **K1** і весь перший ланцюг заллються синім кольором. Контакти **K1** у четвертому та п'ятому ланцюгах замикаються й також зафарблюються в синій колір. Розмикаючі контакти **K3** і **K4** у цих ланцюгах залишаються замкненими, отже, спрацьовують **KM** і **HL**.

Аналогічно необхідно досліджувати роботу програми при всіх інших комбінаціях вхідних сигналів з метою перевірки правильності реалізації алгоритму керування й виявлення можливості неправильних спрацьовувань виконавчих елементів.

Наприклад, необхідно «розімкнути» контакт **SB** і «замкнути» **SK**. Для цього наводимо курсор на **SB**, клацаємо 1ЛКМ. У порожнині контакту з'явиться зелений квадратик, а частина залишиться синьою. Наводимо курсор на **SK**, клацаємо 1ЛКМ. Контакт заливається синім кольором. Відкриваємо меню **Онлайн**, клацаємо 1ЛКМ по рядкові **Записати значення**. Переконаємося, що виключилася котушка **KM**. І так проходимо послідовно по всіх рядках таблиці станів вхідних і вихідних пристроїв.

У випадку дослідження такої простої системи, як у нашому прикладі, можна відразу перевірити тільки умови спрацьовування виконавчих елементів, створюючи «замикання» і «розмикання» відповідних вхідних елементів **SB**, **SL** і **SK** за вищеописаною методикою.

Якщо, приступаючи до дослідження реакції LD -Програми на наступну комбінацію станів вхідних сигналів, необхідно повернути схему в вихідний стан (**SB** = 0, **SL** = 0 і **SK** = 0), то для цього досить клацнути 1ЛКМ по рядкові **Відключення (Logout)**, тобто відключити режим **Онлайн**. Потім по освоєній уже методиці запустити режим емуляції, створити необхідну комбінацію станів вхідних елементів і, нажавши 1ЛКМ по рядку **Записати значення**, зафіксувати поведінку виконавчих елементів.

При великій кількості ланцюгів багатоступінчастої схеми немає можливості бачити її на моніторі цілком навіть при найменшому масштабі. Тому перед натисканням 1ЛКМ на рядок **Записати значення** можна, користуючись прокручуваннями, перемістити ділянку схеми, що цікавить нас, в поле зору й лише потім запустити її.

## 2.7. Завантаження програми в ПЛК

Дотепер розглядалися різні аспекти розробки LD-Програм у припущенні, що їх дослідження буде завершуватися тільки в режимі емуляції, тобто без наступної прив'язки до конкретного ПЛК.

Заповнимо цей пробіл на прикладі розробленої програми (Рис. 15).

Перед завантаженням програми в ПЛК необхідно виконати конфігурацію вхідних і вихідних змінних, тобто ввести імена цих змінних певним чином у конфігуратор ПЛК. Розглянемо цей етап детальніше.

Для цього внизу вікна організатора об'єктів (Рис. 5) необхідно натиснути 1ЛКМ на вкладку **Ресурси (Resources)** і з дерева ресурсів вибрати **Конфігурація ПЛК (PLC Configuration)**. Відкриється вікно (Рис. 19), в якому слід повідомляти вхідні й вихідні змінні.

Нагадаємо, що в розробленій програмі три вхідні змінні (**SB**, **SL**, **SK**) і дві вихідні (**KM** і **HL**). Почнемо із вхідних елементів.

У лівій частині вікна конфігуратора слід розкрити вкладку **PLC150.I**, клацнувши 1ЛКМ по символу «+». Потім 2ЛКМ клацнути по **Discrete Input 6 bit**. Тому що в контролері ОВЕН ПЛК150 усього 6 дискретних входів, то в нашому розпорядженні буде 6 рядків: від **AT%IB0.0.0** до **AT%IB0.0.5**. Ми маємо право вибрати будь-які два входи з шести можливих. Наприклад, нажавши на «+», розкриваємо вкладку **AT % IB0.0: BYTE**. Далі клацаємо 1ЛКМ на самому початку рядка **AT % IB0.0.0: BOOL** і вводимо ім'я змінної **SB**. Завершуємо введення клавішею **Enter**. Аналогічним чином, але тільки вже в рядку **AT % IB0.0.1:BOOL** привласнимо значення змінної **SL**, а в рядку **AT % IB0.0.2: BOOL** – змінної **SK**.

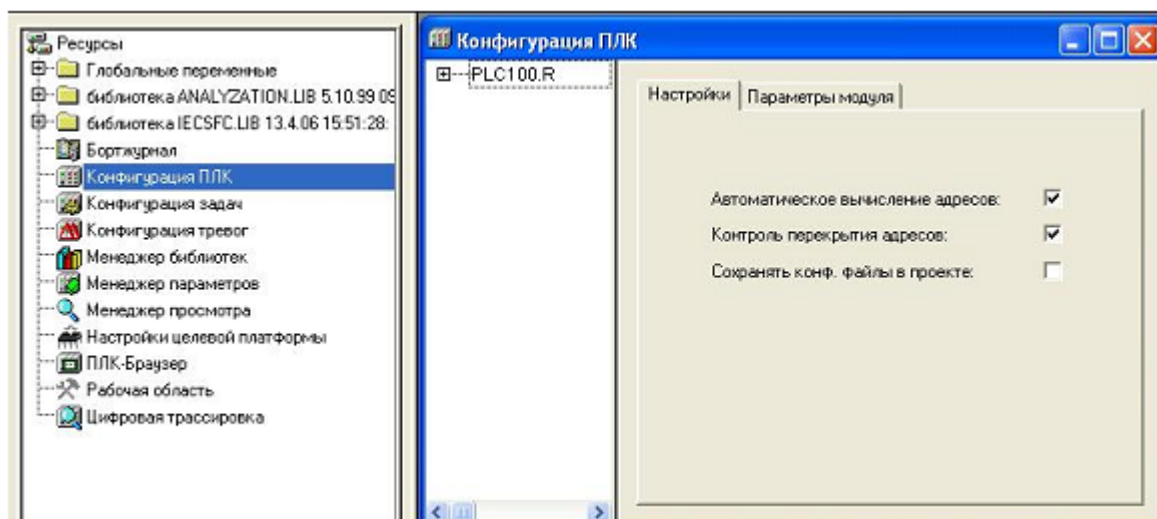


Рис. 19 – Відкриття вікна «Конфігурація ПЛК».

Для введення вихідних змінних необхідно натиснути на «+» біля рядка **Discrete Output-Relay**. У контролері ОВЕН ПЛК150 чотири дискретних виходи. Тому відкриються 4 рядки: від AT%QX1.0 до AT%QX4.0. Вибираємо будь-які дві із чотирьох. Наприклад, клацаємо 1ЛКМ на початку напису АТ у рядку AT%QX1.0: BOOL і вводимо ім'я першої вихідної змінної **КМ**. Аналогічним образом слід увести ім'я другої змінної виводу **НЛ** в рядок AT %QX2.0: BOOL.

Отже, змінні введені, і отриманий результат представлений у вікні конфігуратора (Рис. 20).

Виконані дії по конфігурації ПЛК припускають, що при апаратній реалізації даної системи керування, кнопка керування «Пуск» **SB**, дискретні датчики температури **SK** і рівня **SL**, обмотка магнітного пускача **КМ** і сигнальна лампа **НЛ** будуть підключені до відповідних входів та виходів контролера.

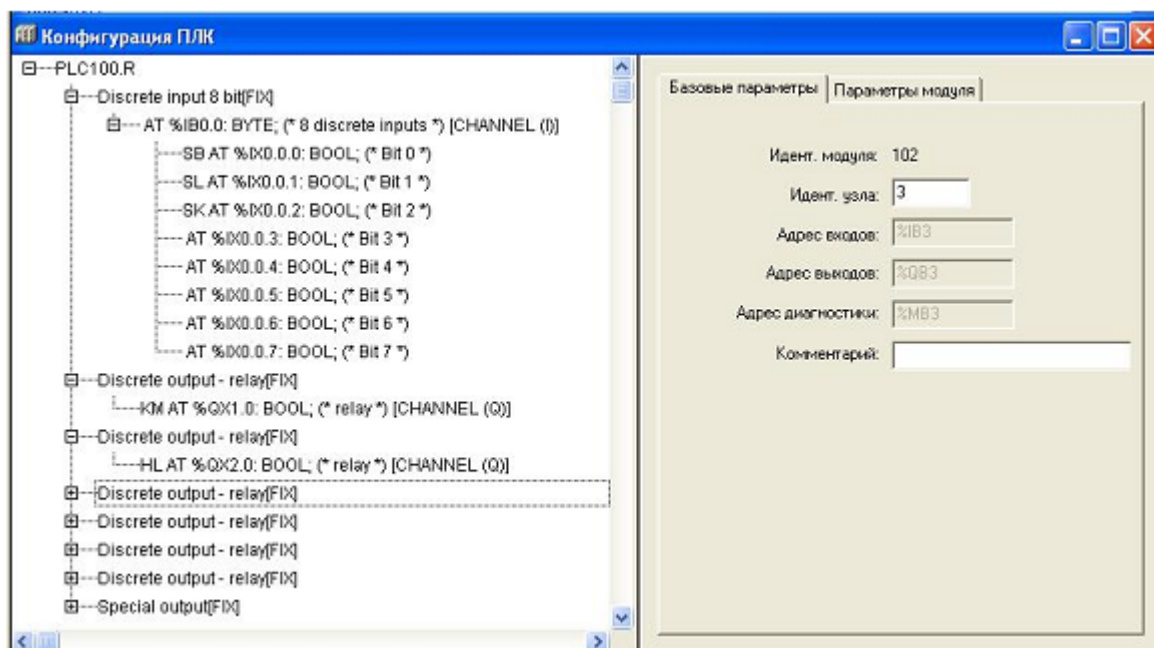


Рис. 20 – Вікно конфігуратора ПЛК.

Тепер створений проект можна відкомпілювати командою **Компілювати** в меню **Проект** або клавішею **F11**. Якщо все зроблене вірно, то у нижній частині вікна повинне з'явитися

повідомлення 0 помилок або **Проект актуальний**. А якщо ні, то необхідно виправити допущені помилки. В цьому допоможуть розгорнуті повідомлення про помилки.

Наступним етапом є підключення контролера до комп'ютера. Контролер ОВЕН ПЛК150 має наступні комунікаційні інтерфейси: Ethernet 10/100 Mbps, RS232 (два канали), RS485. Зв'язок з контролером може бути організований двома засобами: по інтерфейсах Ethernet, Debug RS-232.

Один з варіантів організації зв'язку ПЛК із комп'ютером забезпечується через порт Debug RS-232. Даний порт розташований на лицьовій панелі контролера ОВЕН ПЛК150 і призначений для зв'язку ПЛК із середовищем програмування, завантаження програми користувача її налагодження. Підключення до цього порту здійснюється кабелем, що входить у комплект поставки ОВЕН ПЛК150. Використаний протокол – Gateway (протокол системи Codesys).

Після організації зв'язку виконується налаштування каналу з'єднання з контролером у вікні **Параметри зв'язку (Communication parameters)**, викликуваному командою меню **Online**.

Далі в меню **Онлайн** клацаємо ІЛКМ по рядкові **Підключення**. При цьому **Режим емуляції** повинен бути попередньо відключений.

Установлюється зв'язок між ПЛК і комп'ютером (системою Codesys). Якщо компіляція проекту командою **Компілювати** ще не виконувалася, то вона буде виконана автоматично при з'єднанні контролера з комп'ютером по команді **Підключення**. При цьому виводяться виявлені помилки й попередження.

Команда **Завантаження** в меню **Онлайн** завантажує код проекту (програму) в контролер. Запускає програму в ПЛК команда **Старт** із меню **Онлайн**, а зупиняє – команда **Стоп**.

### 3 Порядок виконання лабораторної роботи.

Лабораторну роботу необхідно виконувати в наступній послідовності:

1. Вивчити основні відомості про склад і можливості програмного комплексу CoDeSys.
2. Запустити на комп'ютері програму CoDeSys.
3. Користуючись даними методичними вказівками, вивчити методику програмування ПЛК мовою LD і відпрацювати в середовищі CoDeSys усі приклади, наведені в розділі 2.
4. Отримати у викладача завдання на розробку програми керування дискретним технологічним об'єктом.
5. На основі аналізу алгоритму керування визначити склад вхідних і вихідних дискретних сигналів.
6. Написати Ld-Програму керування.
7. Дослідити розроблену програму в режимі емуляції.
8. Виконати конфігурацію вхідних і вихідних змінних
9. Компілювати отриману Ld-Програму.
10. Виконати підключення контролера до комп'ютера й подати напругу живлення на ПЛК.
11. Завантажити код розробленої програми в ПЛК.
12. Промодельовати на лабораторному стенді роботу системи логіко-програмного керування. Для цього запустити програму в ПЛК. Імітуючи за допомогою перемикачів і кнопок керування ввід у контролер вхідних дискретних сигналів, перевірити правильність реалізації заданого алгоритму, спостерігаючи за станом індикаторних ламп, що імітують дискретні виконавчі пристрої.
13. Зупинити виконання програми в ПЛК і відключити живлення.
14. Оформити звіт по лабораторній роботі.

#### **4. Зміст звіту.**

У звіті вказується мета лабораторної роботи й приводяться наступні результати:

1. Завдання на розробку програми керування дискретним технологічним об'єктом.
2. Таблиці вхідних і вихідних сигналів із вказівкою найменування сигналів, їх умовних позначень і джерел (датчиків) і виконавчих пристроїв сигналів.
3. Блок-схема алгоритму керування.
4. Текст програми мовою LD.
5. Виводи за результатами дослідження програми в режимі емуляції й моделювання роботи системи керування на лабораторному стенді.

#### **5. Контрольні питання.**

1. Яке призначення базових команд мови?
2. Яка структура Ld-Програми?
3. Як в Ld-Програмі задають перевірку стану вхідних дискретних сигналів?
4. Якими командами в Ld-Програмі формують вихідні дискретні сигнали?
5. Яким чином можна фіксувати вихідні дискретні сигнали?
6. Як виконати конфігурацію вхідних і вихідних змінних?
7. Як перевірити правильність Ld-Програми?
8. Як завантажити код програми в ПЛК?
9. Як запустити програму в контролері?