

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

М. Я. Островерхов

**КОМП'ЮТЕРНІ ЗАСОБИ
АВТОМАТИЗАЦІЇ
ЕЛЕКТРОТЕХНОЛОГІЧНИХ
УСТАНОВОК
КОНСПЕКТ ЛЕКЦІЙ**

Навчальний посібник

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського
як навчальний посібник для здобувачів ступеня бакалавра за освітньою програмою
«Електротехнічні пристрої та електротехнологічні комплекси»
спеціальності 141 «Електроенергетика, електротехніка та електромеханіка»*

Електронне мережне навчальне видання

Київ
КПІ ім. Ігоря Сікорського
2022

Рецензент: *Чумак В.В.*, канд. техн. наук, завідувач кафедри електромеханіки факультету електроенерготехніки та автоматики

Відповідальний редактор *Лободзинський В.Ю.*, канд. техн. наук, доцент

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського (протокол № 5 від 26.05.2022 р.)
за поданням Вченої ради факультету електроенерготехніки та автоматики (протокол № 9 від 17.05.2022 р.)*

Комп'ютерні засоби автоматизації електротехнологічних установок: Конспект лекцій [Електронний ресурс]: навч. посіб. для здобувачів ступеня бакалавра за освітньою програмою «Електротехнічні пристрої та електротехнологічні комплекси» спеціальності 141 «Електроенергетика, електротехніка та електромеханіка» / М. Я. Островерхов; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 3,5 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2022. – 222 с.

У навчальному посібнику наведено теоретичні відомості з основ автоматизації електротехнологічних установок та процесів, основні характеристики промислових мереж та інтерфейсів, принципи керування координатами технологічного процесу, принципи та технологія розроблення проектів автоматизованих систем керування електротехнологічними комплексами на основі SCADA-системи.

Призначено для здобувачів ступеня бакалавра за освітньою програмою «Електротехнічні пристрої та електротехнологічні комплекси» спеціальності 141 «Електроенергетика, електротехніка та електромеханіка». Посібник буде також корисним для здобувачів вищої освіти за іншими освітніми програмами спеціальності 141 «Електроенергетика, електротехніка та електромеханіка».

Реєстр. № НП 21/22- 449. Обсяг 11,2 авт. арк.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
проспект Перемоги, 37, м. Київ, 03056
<https://kpi.ua>

Свідоцтво про внесення до Державного реєстру видавців, виготовлювачів і розповсюджувачів видавничої продукції ДК № 5354 від 25.05.2017 р.

©М. Я. Островерхов
© КПІ ім. Ігоря Сікорського, 2022

ЗМІСТ

ВСТУП	5
1. ОСНОВИ АВТОМАТИЗАЦІЇ ЕЛЕКТРОТЕХНОЛОГІЧНИХ УСТАНОВОК ТА ПРОЦЕСІВ	6
1.1. Основні поняття автоматизованих систем управління	6
1.2. Функції та склад АСУ ТП	9
1.3. Класифікація АСУ ТП	12
1.4. Структура АСУ ТП	14
1.5. Характеристики сучасних SCADA-систем.....	16
1.6. Взаємодія програмного забезпечення АСУ ТП з фізичними пристроями.....	20
1.7. Контрольні запитання до розділу 1	27
2. ПРОМИСЛОВІ МЕРЕЖІ ТА ІНТЕРФЕЙСИ	28
2.1. Основні поняття промислових мереж	28
2.2. Мережева модель OSI	31
2.3. Основні інтерфейси.....	34
2.4. Промислові мережі рівня датчиків та виконавчих пристроїв	36
2.5. Промислові мережі рівня контролерів	44
2.6. Промислові мережі операторського рівня.....	47
2.7. Бездротові промислові мережі	50
2.8. Обладнання та кабелі мережі	56
2.9. Контрольні запитання до розділу 2	71
3. КЕРУВАННЯ КООРДИНАТАМИ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ	72
3.1. Ідентифікація моделей об'єктів керування технологічних процесів.....	72
3.2. Класичні закони керування та їх модифікації.....	99
3.3. Контрольні запитання до розділу 3	108

4. РОЗРОБКА ПРОЕКТІВ АВТОМАТИЗОВАНИХ СИСТЕМ УПРАВЛІННЯ ЕЛЕКТРОТЕХНОЛОГІЧНИМИ КОМПЛЕКСАМИ.....	109
4.1. Розробка функціональних схем автоматизації технологічних процесів	109
4.2. Особливості проектування АСУ ТП в SCADA Trace Mode	121
4.3. Контрольні запитання до розділу 4	131
СПИСОК ЛІТЕРАТУРИ.....	132
Додаток А. Приклад проекту графічного екрану.....	133
Додаток Б. Основні відомості про SCADA Trace Mode.....	134

ВСТУП

Дисципліна «Комп'ютерні засоби автоматизації електротехнологічних установок» є теоретичною та практичною основою для здобувачів ступеня бакалавра за освітньою програмою «Електротехнічні пристрої та електротехнологічні комплекси» спеціальності 141 «Електроенергетика, електротехніка та електромеханіка» в області побудови автоматизованих систем управління електротехнологічними установками та комплексами.

Метою вивчення дисципліни «Комп'ютерні засоби автоматизації електротехнологічних установок» є підготовка студентів в області основ теорії побудови систем автоматизації електротехнологічних установок та комплексів, а також особливостей використання, розробки й налагодження автоматизованих систем управління на основі сучасних програмних та мікропроцесорних засобів.

Основними завданнями дисципліни «Комп'ютерні засоби автоматизації електротехнологічних установок» є набуття конкретних знань, навичок та умінь для вирішення задач в області управління технологічними процесами за допомогою програмних засобів комп'ютерної техніки.

1. ОСНОВИ АВТОМАТИЗАЦІЇ ЕЛЕКТРОТЕХНОЛОГІЧНИХ УСТАНОВОК ТА ПРОЦЕСІВ

1.1. Основні поняття автоматизованих систем управління

В процесі своєї діяльності людина виконує певні дії – операції, які умовно розділяються на робочі операції та операції керування. На виконання робочих операцій витрачається енергія м'язів, а на операції керування – розумова енергія.

Заміна машинами праці людини в робочих операціях називається механізацією. Заміна ж операцій керування технічними пристроями називається автоматизацією. Якщо всі операції керування виконуються цими пристрої, то система називається автоматичною. Якщо ж замінюється лише її частина, то система називається автоматизованою.

Схема побудови автоматизованої системи управління (АСУ) виробництвом представляється відомою пірамідою управління (рис 1.1).

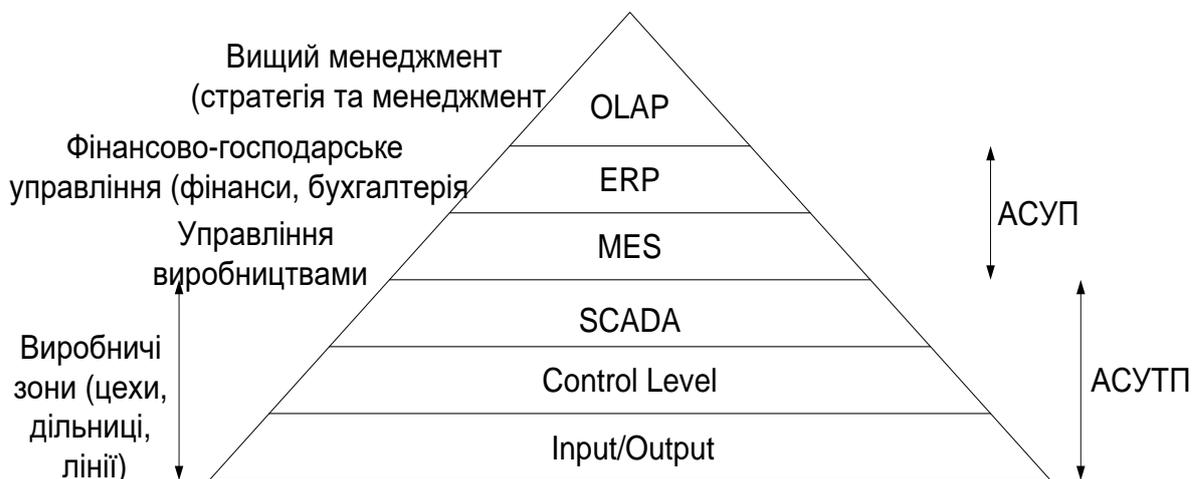


Рисунок 1.1 – Схема АСУ виробництвом

На рис. 1.1. позначено:

АСУ ТП – автоматизована система управління технологічним процесом;

АСУП – автоматизована система управління підприємством;

Input/Output – рівень датчиків та виконавчих пристроїв;

Control Level – рівень систем автоматичного керування (САК) та програмованих логічних контролері (ПЛК) (на англ. Programmable Logic Controller (PLC));

SCADA (Supervisory Control And Data Acquisition) – диспетчерське керування та збір даних;

ERP (Enterprise Resources Planning) – планування ресурсів підприємства. Забезпечує фінансово-господарську діяльність, планування та облік виробництва;

MES (Manufacturing Execution Systems) – система виробничого виконання (або система управління технологією). Вирішує задачі синхронізації, координації, аналізу та оптимізації випуску продукції. MES оперує виключно виробничою інформацією. Якщо ERP здійснюють перепланування в основному один раз на добу, то MES працює в реальному часі.

OLAP (Online Analytical Processing) – аналітична обробка в реальному часі. Ця технологія запропонована в 1993 році у вигляді 12 законів. Це технологія обробки інформації, яка дозволяє створити динамічні звіти з метою аналізу інформації в базі даних і служить для підготовки бізнес звітів маркетингу та глобальним керуванням в цілому.

Автоматизована система управління технологічним процесом АСУТП (на англ. Process Control System або Industrial Control System; ICS) – це система, яка за участю операторського персоналу забезпечує в реальному часі виготовлення або переробку продукції за заданими технологічними та техніко-економічними критеріями.

АСУТП дозволяє вирішити наступні задачі:

- 1) Ввести процес з максимально-допустимою продуктивністю. Автоматично враховувати зміну технологічних параметрів, властивості вихідних матеріалів, зміну навколишнього середовища та помилок операторів;

- 2) Керування процесом з урахуванням плану та номенклатури продукції шляхом оперативного переналаштування технологічного обладнання;
- 3) Реалізувати статистичне керування процесом в реальному часі за екстремальними та адаптивними алгоритмами;
- 4) Здійснювати автоматичне керування в шкідливих та небезпечних для людини умовах.

На рівні Control Level та Input/Output реалізується локальне керування певними параметрами технологічного процесу за допомогою САК на основі контролерів. Швидкодія на цьому рівні складає десятки долі секунд – секунди. Ці рівні формують потік інформації на гору.

Основною функцією SCADA є збір інформації про хід технологічного процесу та оперативне керування. SCADA забезпечує основний інформаційний потік в реальному часі, який поступає в АСУП для прийняття глобальних рішень. Швидкодія процесів на порядок менша, ніж в Control Level.

Додаткова вартість, яка визначає прибутки підприємства формується у виробничій зоні, яку контролює АСУТП, тому достовірна і своєчасна інформація дозволяє реально змінити фінансові показники підприємства. Таким чином прибутки, які формуються у виробничій зоні можуть підсилюватися за допомогою MES. Конкурентна спроможність підприємства визначається виробничою зоною.

АСУТП призначено для вироблення та реалізації керуючої дії на технологічний об'єкт управління (ТОУ). ТОУ – це сукупність технологічного обладнання та реалізованого на ньому, відповідно до інструкцій та регламенту, технологічного процесу виробництва. В якості ТОУ можуть виступати окремі агрегати та установки, які ведуть відносно самостійний процес, або технологічна лінія виробництва всього підприємства або певних його ділянок. Функціонування ТОУ разом з АСУТП створюють автоматизований технологічний комплекс (АТК).

АСУТП – це людино-машинна система керування, яка забезпечує автоматизований збір та переробку інформації з метою оптимізації

технологічного процесу. Найчастіше критерієм керування виступають техніко-економічні показники: собівартість, продуктивність виробництва, мінімізація витрат, підвищення якості продукції.

1.2. Функції та склад АСУ ТП

Функції АСУТП визначаються її задачами:

- 1) Економія палива, сировини та інших виробничих ресурсів;
- 2) Забезпечення безпечного функціонування;
- 3) Підвищення показників якості продукції;
- 4) Зниження витрат на утримання персоналу;
- 5) Оптимізація технологічного процесу.

Основні функції АСУТП:

- 1) Керуюча функція. Регулювання та стабілізація окремих технологічних змінних. Логічне керування окремими апаратами. Програмно-логічне керування групою обладнання. Оптимальне керування технологічним процесом. Адаптивне керування.
- 2) Інформаційна функція. Збір, обробка та представлення інформації оперативному персоналу або передача її на вищий рівень автоматизації:
 - централізований контроль та вимір технологічних параметрів, посереднє (непряме) обчислення технологічних показників або змінних;
 - формування та видача даних оперативному персоналу;
 - узагальнена оцінка та прогноз стану АТК;
 - підготовка та передача інформації у суміжні системи.
- 3) Допоміжна функція. Не має користувача за межами АСУТП.

В залежності від ступеня участі людини в процесі керування є дві функції або два режими роботи АСУТП:

- 1) Автоматизований – реалізує керування за участю оператора:

- *ручний режим*, при якому комплекс технічних засобів (КТЗ) представляє оперативному персоналу інформацію про ТОУ, а вибір та реалізацію керування здійснює оператор;
 - *режим порадики*, при якому КТЗ виробляє рекомендації для керування, а рішення про його використання приймає оператор;
 - *діалоговий режим* (супервізорний), при якому КТЗ виробляє та реалізує керування, а оператор має важливість його корегування;
- 2) Автоматичний режим – реалізує керування без участі оперативного персоналу:
- *режим непрямого (посереднього) керування*, коли КТЗ змінює завдання (уставки) локальним САК або здійснює налаштування параметрів САК;
 - *режим прямого цифрового керування*, коли КТЗ безпосередньо керує виконавчими пристроями.

Різниця між САК та АСУ полягають у функціях, які вони виконують (рис. 1.2). САК призначена для оптимального відпрацювання завдання, а АСУ для оптимізації режиму роботи об'єкта керування (табл.1.1).

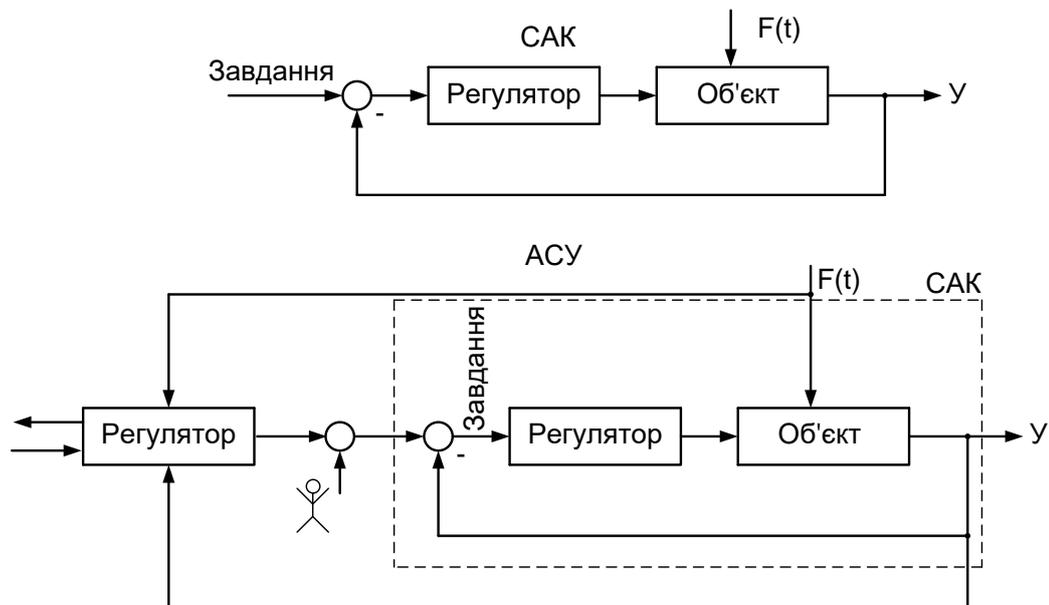


Рисунок 1.2 – Структурні схеми САК та АСУ

Таблиця 1.1 – Різниця між САК та АСУ

САК	АСУ
<i>Мета функціонування:</i>	
Оптимальне відпрацювання завдання	Оптимальний режим роботи
<i>Критерії якості функціонування:</i>	
Показники якості регулювання	Показники ефективності роботи
<i>Спосіб досягнення мети:</i>	
Вибір алгоритму та параметрів регулятора	Вибір оптимального сигналу завдання

Для виконання функцій потрібна взаємодія наступних складових АСУТП:

- 1) Технічне забезпечення;
- 2) Програмне забезпечення;
- 3) Інформаційне забезпечення;
- 4) Організаційне забезпечення;
- 5) Оперативний персонал.

Технічне забезпечення представляє собою повний склад КТЗ, достатніх для функціонування АСУТП:

- засоби отримання інформації (датчики);
- засоби перетворення, зберігання, відображення та реєстрації інформації;
- пристрої передачі сигналів;
- пристрої вироблення керуючих сигналів (контролери).

Програмне забезпечення – сукупність програм, необхідних для реалізації АСУТП та заданого функціонування технічних засобів. Буває загальне (системне) та спеціальне. Загальне поставляється з технічними засобами та забезпечує їх функціонування. Спеціальне реалізує керування технологічним процесом.

Інформаційне забезпечення включає інформацію про стан процесу та технічні засоби, системи класифікації і кодування технологічної та техніко-економічної інформації, масиви даних і документів, а також нормативно-довідкова інформація.

Організаційне забезпечення представляє собою сукупність інструкцій та регламентів для оперативного персоналу, а також опис функціональної, технічної та організаційної структури.

Оперативний персонал складається з:

- операторів-технологів (диспетчерів), які здійснюють контроль та керування технологічним об'єктом;
- експлуатаційного персоналу, що підтримує програмно-технічні засоби АСУТП. *Ремонтний персонал не входить до складу оперативного персоналу АСУТП.*

АСУТП може розбиватися на підсистеми за функціональною ознакою (наприклад, інформаційна, керуюча підсистема) або за структурною ознакою, яка ділить на частини ТОУ.

1.3. Класифікація АСУТП

Класифікація, що приводиться нижче, направлена на вирішення наступних задач:

- 1) Вибір систем-аналогів на ранніх етапах розробки АСУТП;
- 2) Попередня оцінка ресурсів та планування робіт по створенню АСУТП;
- 3) Визначення науково-технічного рівня АСУТП;
- 4) Визначення капіталоємності у відносних одиницях.

Є п'ять класифікаційних ознак:

- 1) За рівнем, який займає АСУТП в структурі підприємства:
 - АСУТП нижнього рівня для ТОУ: агрегати, установки, ділянки (код 1);

- АСУТП верхнього рівня для ТОУ: групи установок, цехи виробництва. При цьому АСУТП не має в своєму складі АСУТП нижнього рівня (код 2);
 - АСУТП багаторівневі: групи установок, цехи виробництва. При цьому АСУТП має в своєму складі АСУТП нижнього рівня (код 3).
- 2) За характером протікання технологічного процесу (ТП) у часі:
- АСУ з перервним ТП. Процес характеризується перервною подачею сировини та реагентів з процесом, близьким до усталеного (код Н);
 - АСУ з перервно-дискретним ТП. Процес характеризується поєднанням неперервних та перервних режимів функціонування, як технологічних апаратів так і стадій процесу (код П);
 - АСУ з дискретним ТП. Процес характеризується перервними технологічними процесами з несуттєвою для керування тривалістю технологічних операцій (код Д).
- 3) За умовною інформаційною потужністю ТОУ:
- найменша – число контрольованих технологічних змінних (тегів) $10 \div 40$ (код 1);
 - мала – число контрольованих технологічних змінних $41 \div 160$ (код 2);
 - середня – число контрольованих технологічних змінних $161 \div 650$ (код 3);
 - підвищена – число контрольованих технологічних змінних $651 \div 2500$ (код 4);
 - велика – число контрольованих технологічних змінних >2500 (код 5);
- 4) За рівнем функціональної надійності:
- мінімальний рівень надійності – рівень надійності практично не регламентується, не потребує спеціальних засобів (код 1);
 - середній рівень надійності – рівень надійності регламентується, проте відмови АСУТП не приводить до зупинки ТОУ (код 2);
 - високий рівень надійності – рівень надійності жорстко регламентується, бо відмовлення можуть привести до зупинки ТОУ (код 3).
- 5) За типом функціонування АСУТП:

- інформаційна АСУТП – автоматично виконуються лише інформаційні функції, а керування реалізує оператор (код І);
- локально-автоматична АСУТП – автоматично виконуються інформаційні функції та функції локального керування, а керування процесом в цілому реалізує оператор (код Л);
- порадна АСУТП – автоматично виконуються інформаційні функції та функції локального керування, а також формуються поради по керуванню процесом в цілому. Керування реалізує оператор (код С);
- автоматична АСУТП – автоматично виконуються всі функції. Керування контролює оператор (код А).

Наприклад, АСУТП з кодом **ЗН43С** означає:

З – багаторівнева;

Н – з неперервним технологічним процесом;

4 – з підвищеною інформаційною потужністю;

3 – з високим рівнем надійності;

С – з функціонуванням в режимі «порадника».

1.4. Структура АСУТП

Найчастіше використовується трирівнева структура АСУТП.



Рисунок 1.3 – Структура АСУТП

Рівень датчиків та виконавчих пристроїв (актуаторів або актюаторів).

Датчики фізичних величин виробляють сигнали аналогового рівня. Найбільш поширені стандарти за напругою ± 15 В, ± 10 В та за струмом $0 \div 20$ мА, $4 \div 20$ мА.

Є дискретні датчики наявності об'єктів, їх положень тощо. Ці датчики можуть виробляти сигнали типу «сухий контакт», забезпечуючи гальванічну розв'язку: Вкл. – $R \approx 0$, Викл. – $R \approx \infty$. Інший тип таких датчиків найчастіше реалізовано за електронною схемою з відкритим колектором типу «мокрый контакт» без гальванічної розв'язки: Вкл – R_{\min} , Викл. – R_{\max} ,

Первинні давачі датчиків (елементи датчиків) можуть мати різну природу та різний рівень сигналу. Для передачі цих сигналів на технологічний рівень використовуються пристрої зв'язку з об'єктом (ПЗО). До задач ПЗО входять:

- 1) Нормалізація сигналів до стандартизованих рівнів;
- 2) Гальванічна розв'язка;
- 3) Попередня фільтрація сигналів;
- 4) Перетворення до стандартів промислових інтерфейсів.

ПЗО можуть виконуватися у вигляді окремих пристроїв або вбудовуватись в датчики, що характерно для сучасних «інтелектуальних» датчиків, або виконуються у вигляді інтерфейсних плат або у вигляді окремих елементів контролерів технологічного рівня.

Найбільш поширеними виконавчими пристроями є електроприводи (сервоприводи). Використовуються також електромагнітні, електрогідравлічні та електропневматичні виконавчі пристрої.

Технологічний рівень реалізується на мікропроцесорних пристроях:

- 1) Контролери на базі ПК, в тому числі для промислового застосування випускають відомі фірми: Octagon, Advantech, Analog Devices.
- 2) Локальні ПЛК. Можуть бути вбудовані в обладнання або у вигляді автономних пристроїв. Вони мають порти для з'єднання з датчиками по системі «точка-точка» та інтерфейси для підключення в комп'ютерну мережу. Випускаються також локальні ПЛК спеціального типу (наприклад, для аварійного захисту). Найчастіше вони мають близько 10 входів/виходів.

Найвідоміші фірми: General Electric, Fanuc Automation, RockWell Automation, Schneider Automation, Siemens.

- 3) Мережевий комплект контролерів. Найбільш широкий клас, який впроваджується в АСУТП. Контролери мають декілька локальних пультав операторів, набір інтерфейсів для різних стандартів комп'ютерних мереж та дозволяють реалізувати різні топологічні схеми мереж.
- 4) Повномасштабні серії контролерів. Одна така система дозволяє автоматизувати виробничу діяльність великого підприємства. Найбільш відомі фірми: ABB, Honeywell, Valmet, Yokogawa .

Диспетчерський рівень представляється операторськими станціями автоматизованих робочих місць (АРМ) спеціалістів, серверами баз даних. Найчастіше вони реалізуються на ПК. Основою програмного забезпечення цього рівня є SCADA-системи.

1.5. Характеристики сучасних SCADA-систем

SCADA-система – це спеціалізоване програмне забезпечення, яке реалізує інтерфейс між людиною та системою керування, а також комунікацію з навколишніми системами (рис. 1.4).

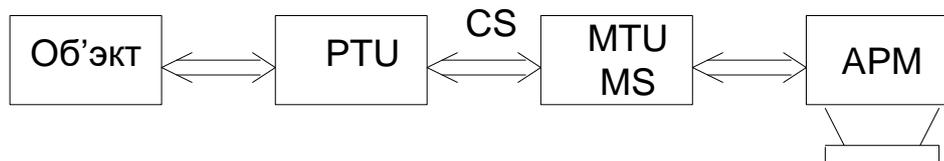


Рисунок 1.4 – Інтерфейс на основі SCADA- системи

АРМ – автоматизоване робоче місце;

MTU (Master Terminal Unit) – диспетчерський пункт керування;

MS (Master Station) – головний термінал, який здійснює обробку даних та керування високого рівня в режимі квазі-реального часу;

CS (Communication System) – канал зв'язку, необхідний для передачі даних між об'єктом на центральному пункті оператора;

RTU (Remote Terminal Unit) – віддалений термінал, який здійснює обробку задач в режимі реального часу.

Задачами SCADA-системи є:

- 1) Збір інформації про контрольовані технологічні параметри;
- 2) Зберігання інформації в архівах;
- 3) Обробка прийнятої інформації;
- 4) Графічне представлення ходу технологічного процесу;
- 5) Прийом команд від оператора;
- 6) Реєстрація подій;
- 7) Інформування персоналу про аварійні ситуації;
- 8) Створення звітів;
- 9) Автоматизоване керування технологічним процесом.

Для вирішення задач SCADA-система має наступний набір елементів:

- 1) Набір кнопок, поворотних регуляторів та інших перемикачів для керування процесом;
- 2) Набір індикаторів та графіків для індикації процесу;
- 3) Шаблони різного виду звітів;
- 4) Спрощена мова створення алгоритмів керування;
- 5) Драйвери обладнання;
- 6) Різні функції мережі.

Основні існуючі SCADA-системи наведено у табл. 1.2

Таблиця 1.2 – Основні SCADA-системи

SCADA система	Фірма	Країна
In Touch	Wonder ware	США
i FIX	Intellution	США

Factory	Unated States Data Co	США
Genesis	Iconic	США
Real Flex	BySoftWare Systems	США
RS View	RockWell	США
Simplicity	GE Famie Automatic	США
Bridge View		США
Monitor Pro	Schneider Electric	Франція
Win Cc	Siemens	Німеччина
Sitix	Jade Software	Великобританія
Citect	CiTechnologies	Австралія
IGSS	Seven Technologies	Данія
Advantech Studio	Advantech	Тайвань
Trace Mode	AdAstra	Росія
MIK Sys	МИФИ	Росія

Найчастіше SCADA-система складається з двох частин:

- 1) Середовище для розробки проекту;
- 2) Інструментальне середовище для запуску готового проекту в роботу.

Більшість SCADA-систем реалізовано на Windows-платформах, що забезпечує легку модернізацію людино-машинного інтерфейсу (НМІ – Human Machine Interface) та легке навчання персоналу.

Функції середовища для розробки проекту:

- 1) *Графічне середовище.* Всі системи мають графічний інтерфейс для створення діалогу оператора з АСУТП. Зокрема, Trace Mode та Genie запускають стандартне графічне вікно з можливістю розташування на ньому бібліотечних графічних елементів для створення мнемосхем технологічного процесу, а In Touch Citect мають набір готових шаблонів. Графічні зображення на екрані з мнемосхемою технологічного процесу можуть бути статичними та динамічними.

2) *Подія* в SCADA-системі – це видача повідомлення про зміну стану системи, яка не потребує термінової уваги оператора.

3) *Тривога (Alarm)* – це вихід якоїсь змінної за критичні значення, порушення роботи обладнання та інше, що може викликати аварійну ситуацію. Сигнал тривоги може бути дискретним або аналоговим.

Дискретні тривоги приймають два значення типу: True/False; 1/0; On/Off.

Характер сигналу аналогової тривоги представлено на рис. 1.5.

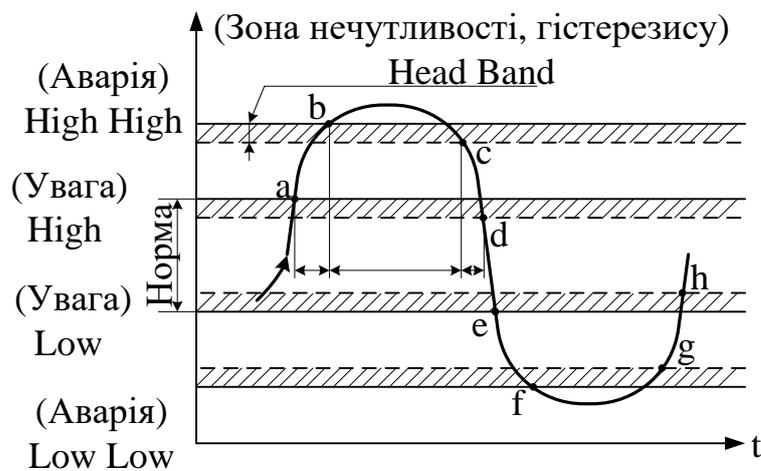


Рисунок 1.5 – Аналоговий сигнал тривоги

Аналогова тривога може бути за швидкістю зміни параметру ROC (Rate of Change). Тривога може також бути представлена як відхилення від норми – Deviation.

4) *Тренди*. Динаміку зміни технологічних параметрів зручно представляти залежностями у часі. Тренди – це графіки зміни параметрів. Вони бувають поточні (реального часу) та архівні (історичні). Тренди стають архівними після запису на диск поточних трендів.

Деякі SCADA-системи крім залежностей у часі дозволяють побудувати гістограми, кругові діаграми та залежності між змінними.

5) *Програмування*. Всі SCADA-системи мають вбудовані мови програмування, які можна розділити на дві групи:

- мови, що направлені на технологів;
- мови для системних інтеграторів.

Зокрема, в In Touch програмування побудоване на скриптах. Скрипт – це частина програми, що прив’язана до конкретної події. Мова наближається до спрощеного варіанту Visual Basic. Citect має мову програмування CiCode, яка схожа на Basic та Ci. Genie має графічну мову програмування, а також спрощений Basic. Trace Mode використовує модифіковані мови, які застосовуються для програмування ПЛК (PLC):

Техно ST – мова схожа на Pascal та Ci;

Техно IL – мова схожа на Asembler;

Техно FBD – графічна мова;

Техно LD – мова релейних діаграм;

Техно SFC – мова схожа на алгоритм.

б) *Звіти*. Кожна SCADA-система дозволяє формувати звіти про хід технологічного процесу, про події та тривоги. Формат звітів – це файли з розширенням .pdf, .html, .xls. Звіти можуть формуватися по команді оператора, періодично у відповідності з програмою роботи, за подією, при тривозі, за результатами обчислень.

1.6. Взаємодія програмного забезпечення АСУТП з фізичними пристроями

Взаємодія здійснюється за допомогою методів: DDE, OLE, COM, DCOM, OPC.

DDE (Dynamical Date Exchange) – динамічний обмін даними. З’явився в 1987 році разом з Windows 2.0.

OLE (Object Linking and Embedding) – зв’язування та впровадження об’єктів.

COM (Component Object Model) – модель багатокomпонентних об’єктів.

DCOM (Distributed Component Object Model) – модель багатокомпонентних об'єктів для розподілених систем. Дозволяє взаємодіяти програмам, які виконуються на різних комп'ютерах локальної мережі. Вона стала універсальною технологією взаємодії SCADA-системи, як клієнта із сервером, який забезпечує зв'язок з апаратами і засобами. DCOM стало базою для створення OPC.

OPC (OLE for Process Control) – найчастіше працюють за схемою «клієнт-сервер».

Метод OPC

Стандарт OPC створено в 1998 році. В даний момент створена організація OPC-Foundation. Вже в 2008 році в цю організацію входило 400 фірм.

OPC – це набір специфікацій стандартів:

1) OPC DA (OPC Data Access) – це специфікація для обміну даними між клієнтом (SCADA наприклад) та апаратурою (ПЛК, пристрої зв'язку з об'єктом ПЗО, інтелектуальні датчики) в реальному часі. Найбільш поширена версія 2.05.

2) OPC Alarm&Events (A&E) – специфікація для повідомлень клієнта про події та сигнали тривоги.

3) OPC HDA (Historical Data Access) – специфікація для доступу до архівних даних.

4) OPC Batch – специфікація для особливих фізико-хімічних технологічних процесів, які не є неперервними.

5) OPC Data Exchange – специфікація для обміну даними між двома OPC DA серверами через мережу Ethernet (найбільш поширена офісна мережа).

6) OPC Security – специфікація, яка визначає методи доступу клієнтів до сервера для забезпечення захисту від несанкціонованих змін.

7) OPC XML DA – це правила для представлення первинних даних за допомогою Web-технологій.

8) OPC Complex DATA – додаткова специфікація до OPC DA і OPC XML DA, яка дозволяє працювати серверам зі змішаними та складними типами даних.

9) OPC Commands – набір програмних інтерфейсів, які дозволяють клієнтам та серверам ідентифікувати, посилати та контролювати команди, які виконуються в технічних пристроях.

10) OPC Unified Architecture (UA) – останній набір специфікацій, який не прив'язаний до Windows-платформи, тому може застосовуватися в інших операційних системах.

OPC DA-сервер

Найбільш широко використовуваний сервер.

Дані складаються з 3-х полів:

- 1) Значення;
- 2) Якість (контроль похибок передачі даних);
- 3) Мітка часу.

Є 4 способи читання даних з сервера:

- 1) Синхронний – клієнт посилає запит і чекає відповіді. Недоліком є затримка і завантаженість мережі.
- 2) Асинхронний – клієнт посилає запит і продовжує поточну роботу. Сервер виконує запит і посилає підтвердження.
- 3) Режим підписки. Клієнт повідомляє сервер про список тегів – змінних за якими потрібно прислати дані в разі їх зміни.
- 4) Режим оновлення даних – клієнт одночасно читає всі активні теги.

В кожному режимові дані можуть бути прочитані з активного пристрою.

Запис даних до фізичного пристрою виконується лише двома методами: синхронним або асинхронним, причому прямим способом без буферизації.

OPC сервер встановлюється як звичайна програма, тому вона заноситься в реєстр Windows і може викликатися будь-яким чином, тобто від клієнтської SCADA чи іншої програми.

OPC сервер встановлюється поверх драйверів пристроїв, не вносячи змін в їх роботу.

Приклад взаємодії програм та фізичних пристроїв через OPC-сервер на одному комп'ютері показано на рис. 1.6.

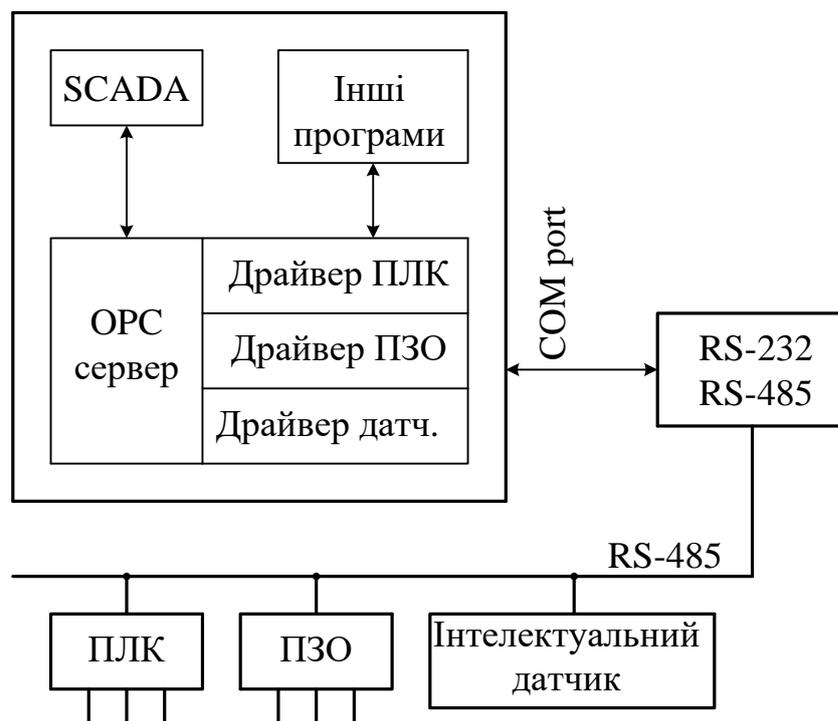


Рисунок 1.6 – Схема взаємодії програм та фізичних пристроїв через OPC-сервер

Приклад використання OPC-технологій для мережевого доступу до даних представлено на рис. 1.7.

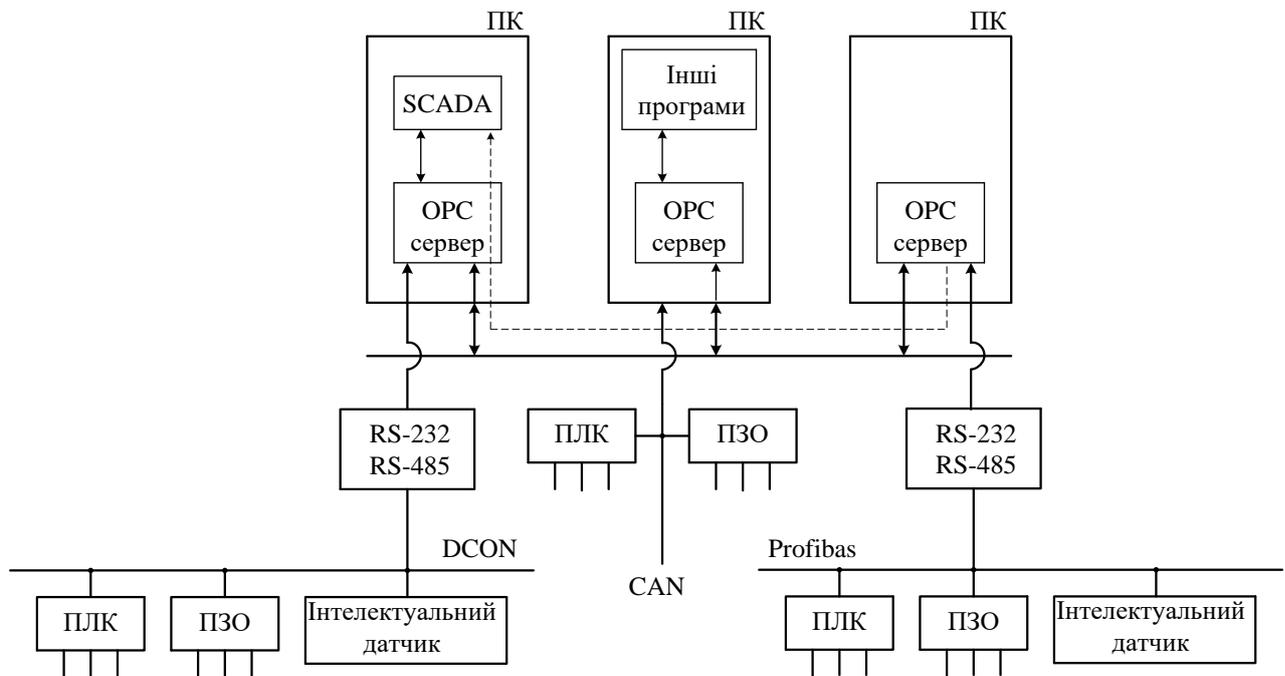


Рисунок 1.7 – OPC-технологія для мережевого доступу до даних

OPC HDA-сервер

OPC HDA використовується для створення архівів. Дає клієнту єдиний інтерфейс для обміну даними, в якості яких можуть виступати стандартні файли, стандартні файли управління базами даних та інші OPC DA та OPC HDA сервери. Структура та методи використання не такі як для OPC DA.

Існує 2 види HDA серверів:

- 1) Простий сервер даних передісторії для побудови графіків;
- 2) Сервер для зберігання даних в упакованому виді з можливістю їх обробки та аналізу.

При керуванні повільними процесами, коли інформація поступає не частіше мілісекунд, архівний OPC HDA можна створити в програмі Excel.

OPC UA-сервер

Сервер, робота якого заснована на Web-технологіях. Популярні OPC сервери побудовані на Windows-платформах, що є недоліком. Крім того

програми (DCOM-технології) є закритими. В 2006 році запропонована специфікація OPC UA. Основа OPC UA побудована на архітектурі SOA (Service Oriented Architecture) – архітектура орієнтована на сервіси. OPC повідомлення передаються за допомогою протоколів HTTP з використанням мови XML та транспортного механізму SOAP.

Обмін інформацією може здійснюватися між клієнтом і декількома серверами, а також між клієнтами. Всі елементи, які представляють собою дані, створюють адресне поле у вигляді вузлів. Причому сервер може зробити їх видимими та невидимими клієнту. Обмін може здійснюватися за схемою «запит – відповідь» або «за підпискою» (рис. 1.8).

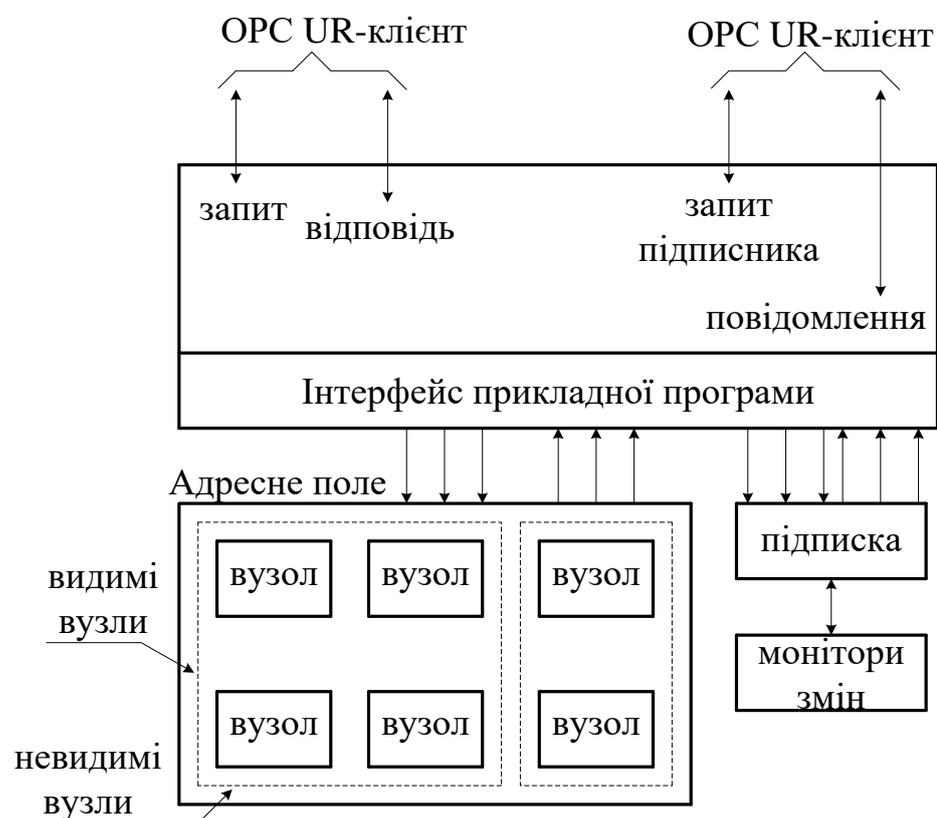


Рисунок 1.8 – Схема обміну інформацією на основі OPC UA

ОС реального часу

Швидкодія ПЛК та ПК впливає на величину динамічної похибки та стійкість системи.

Відповідно до стандарту POSIX IEEE 100s_1. Реальний час в ОС – це здатність ОС забезпечити необхідний рівень сервісу за певний проміжок часу. Базовими вимогами для забезпечення реального часу: високопріоритетні задачі повинні виконуватися в першу чергу; повинна бути виключена інверсія пріоритетів; процеси та потоки, час виконання яких не можливо запланувати, ніколи не повинні повністю займати ресурси системи. В системах реального часу найчастіше відсутня віртуальна пам'ять.

Windows CE NET – операційна система жорсткого часу. Властивості:

- 1) Одночасне виконання до 32 процесів;
- 2) 256 рівнів пріоритетів;
- 3) Карусельне виконання ланок з однаковим пріоритетом. Середній час обробки переривання 2,8 мсек.
- 4) Мінімальна конфігурація може бути встановлена при об'ємі пам'яті 200 кбайт.

QNX Neutrino – це UNIX-подібна операційна система жорсткого часу забезпечує багаторівневий режим з пріоритетами. Складається із ядра та планувальника процесів і серверів (невеликі задачі, які виконують основні функції ОС).

Microware System – це багатозадачна та багатокористувацька операційна система, що працює в режимі реального часу. Найчастіше застосовується у вбудованих системах.

1.7. Контрольні запитання до розділу 1

1. Визначення понять «механізація», «автоматизація», «автоматизована система», «автоматична система», «АСУТП».
2. Піраміда управління виробництвом та опис її складових.
3. Функції та задачі АСУТП. Різниця між САК та АСУ.
4. Склад та класифікація АСУТП.
5. Структура АСУТП та опис її складових.
6. Основні характеристики сучасних SCADA-систем.
7. Взаємодія програмного забезпечення АСУТП з фізичними пристроями.
8. Метод OPC. Приклади взаємодії через OPC-сервер.

2. ПРОМИСЛОВІ МЕРЕЖІ ТА ІНТЕРФЕЙСИ

2.1. Основні поняття промислових мереж

Обмін інформацією в АСУТП здійснюється за допомогою промислових мереж (на англ. FieldBus – польова шина).

Переваги підключення пристроїв за допомогою мережі:

- 1) В декілька разів зменшуються витрати на кабель;
- 2) Збільшується відстань між датчиками та виконавчими пристроями;
- 3) Спрощується керування і модифікація датчиками та виконавчими пристроями;
- 4) Є можливість дистанційно налаштовувати та вести діагностику обладнання.

Недолік використання промислових мереж – при розриві кабелю зв'язку втрачається можливість керування зразу декількома пристроями, тому для підвищення надійності необхідне резервування.

Промислова мережа відрізняється від офісної наступними властивостями:

- 1) Спеціальне виконання, яке забезпечує захист від вібрації, пилу;
- 2) Більш широкий діапазон температур ($-40 \div +60^{\circ}\text{C}$);
- 3) Підвищена міцність кабелю, затискачів, ізоляції;
- 4) Підвищена надійність передачі даних.

Промислова мережа має ту ж саму класифікацію, що і офісна:

- 1) LAN (Local Area Network) – мережа на обмеженій території;
- 2) MAN (Metropolitan Area Network) – мережа міст;
- 3) WAN (Wide Area Network) – глобальна мережа, що реалізується на основі інтернет-технологій.

Найвідоміші промислові мережі: Modbus, Profibus, DeviceNet, CANopen, Lonworks, ControlNet, ASI, Ethernet, InterBus, Foundation, Fieldbus.

З'єднання промислової мережі з її компонентами здійснює за допомогою інтерфейсів. Найбільш поширені послідовні інтерфейси: RS-485, RS-232, RS-422, Ethernet, CAN, HART, ASI.

Для обміну інформацією пристрої повинні мати однаковий протокол обміну, який реалізується апаратно, програмно та програмно-апаратно.

Мережа може реалізувати декілька протоколів, які називаються стеком протоколів.

Взаємодія пристроїв в промисловій мережі здійснюється за моделями:

- 1) Клієнт-сервер;
- 2) Видавець-передплатник (виробник-споживач).

В моделі «клієнт-сервер» може бути декілька клієнтів та декілька серверів. Клієнт посилає запит, а сервер відповідає. Ця модель зручна для передачі даних, які змінюються періодично. Якщо ж події виникають випадково, то ця модель перевантажує трафік.

В моделі «видавець-передплатник» є один виробник і багато споживачів. Споживачі спочатку підписуються на розсилку певних тегів, а виробник розсилає їх за графіком або при їх зміні.

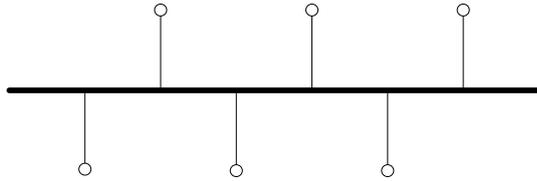
В будь-якій моделі є пристрій, який називається головним або Master або ведучим та пристрій, який відповідає на запити майстра – ведомий, підпорядкований або Slave. Мережа може мати одного або декілька майстрів. При наявності декількох майстрів виникає проблема вирішення конфліктів, які вирішуються в кожній мережі по своєму. В Profibus – це метод передачі маркера. В CAN – це побітне порівняння ідентифікаторів. В Ethernet – це метод прослуховування мережі. У безпроводникових мережах – це метод запобігання колізій.

У всіх мережах застосовується ширококомовна розсилка даних без вказування конкретного адресу.

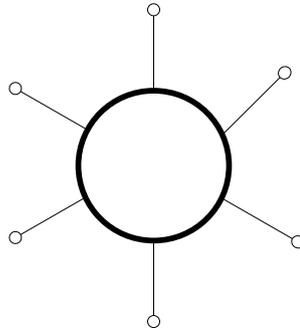
Передача повідомлень здійснюється через канал зв'язку за допомогою передавача та приймача. В якості каналу зв'язку виступає середовище передачі (кабель – вита пара, кабель – оптоволокло, коаксіальний кабель, ефір, плоский кабель).

Мережі можуть мати різну топологію:

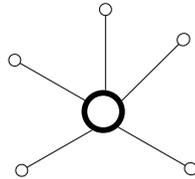
1. Шинна (найбільш поширена).



2. Кільцева.



3. Зірка.



4. Деревовидна.

В мережах може бути 5 типів даних:

- 1) Сигнали – це результати вимірювань. Їх життя дуже коротке. Потрібно їх отримати в максимально-короткий термін.
- 2) Команди – це повідомлення, які викликають дію (наприклад, закрити клапан). Висока надійність передачі, повторно не передаються.
- 3) Стан – показує поточний та майбутній стан системи. Повідомлення може посилатися повторно. Час доставки не жорсткий.
- 4) Подія – вихід змінної за межі, гарантований час доставки.
- 5) Запит – це команда на яку необхідно отримати відповідь.

Дані в мережі передаються порціями: фрейм, кадр, сегмент, дейтаграма.

Основними параметрами промислової мережі є продуктивність та надійність. Продуктивність характеризується часом реакції (запит-відповідь) та пропускнуою здатністю (біт за секунду). Надійність характеризується коефіцієнтом готовності, вірогідністю доставки даних, ймовірний час доставки, безпечність та відмова стійкість. В останній час для опису надійності

використовується термін «якість обслуговування», яка визначає вірогідність передачі заданого потоку даних між двома вузлами.

2.2. Мережева модель OSI

OSI (Open Systems Interconnection) (1978 р). Абстрактна мережева модель для комунікацій та розробки мережевих протоколів (рис. 9.1).

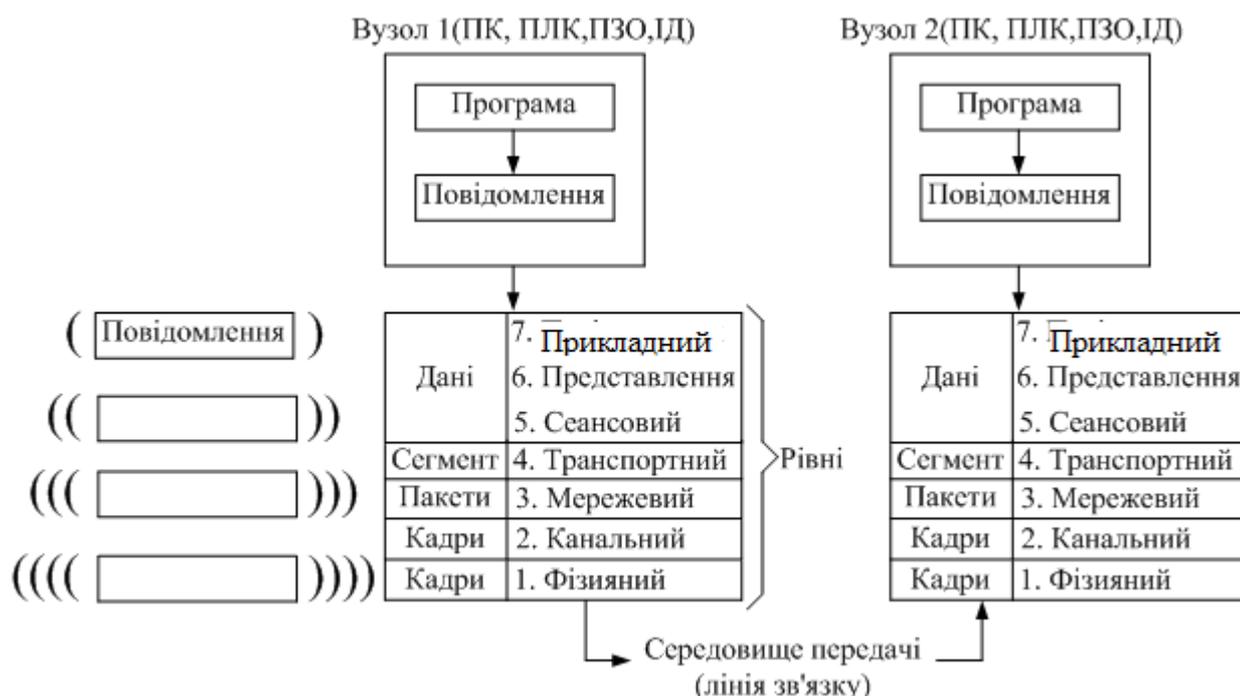


Рисунок 2.1 – Мережева модель OSI

Кожен рівень може взаємодіяти тільки із сусідніми рівнями.

7. Прикладний рівень (Application Layer) забезпечує взаємодію програми користувача з мережею. Він дозволяє програмі користувача використовувати мережеві служби (удалений доступ до файлів та баз даних, пересилка електронної пошти). Відповідає за пересилку службової інформації, повідомляє програму про помилки, формує запити та рівень представлення. HTTP, POP3, SMTP, FTP, XMPP, OSCAR, ModBus, SIP, TELNET.

6. Рівень представлення відповідає за перетворення протоколів і кодування/декодування даних. Запити, що прийшли з прикладного рівня, перетворюються в формат для передачі по мережі. На цьому рівні може здійснюватися стискання, розпакування даних або перенаправлення на інший мережевий ресурс. Перетворення здійснюється з метою обміну інформацією між вузлами на різних платформах.

Перетворюються дані наступних типів:

- Текстова інформація: ASCII, EBCDIS;
- Графічне зображення: PICT;
- Тегований формат зображень: TIFF;
- Для фотографій: JPEG;
- Звук та мікрофрагменти: MIDI, MPEG.

5. Сеансовий рівень відповідає за підтримку сеансу зв'язку, керує створенням/завершенням сеансу, обміном інформацією, визначенням прав на передачу, підтримкою сеансу в період неактивності, відновлення процесу при порушенні взаємодії. ADSP; ASP; H,245; ISO-SP; NetBIOS; PPTP; RTCP.

4. Транспортний рівень призначений для доставки даних без помилок. Тут дані поділяються на кадри. Transport Layer: ATP, CUDP, DCCP, FCP.

3. Мережевий рівень призначений для визначення шляху передачі даних. Відповідає за трансляцію логічних адрес та імен у фізичні, моніторинг неполадок та заторів мережі. На цьому рівні працюють маршрутизатори (роутери). IP/IPv4/IPv6, IPx, x.25, I.Psec, RIP

2. Канальний рівень відповідає за взаємодію мереж на фізичному рівні та за контролем помилок. Отримані з фізичного рівня дані упаковуються в кадри, перевіряються, виправляються. На цьому рівні працюють комунікатори та мости. Програмно цей рівень представляється драйверами мережевих плат. В операційних системах є спеціальний програмний інтерфейс для взаємодії каналного та мережевого рівня. ATM, Ethernet, CAN.

1. Фізичний рівень здійснює передачу електричних або оптичних сигналів по кабелю або в радіоефірі, а також прийом та перетворення бітів даних

відповідно з методами цифрового кодування. На цьому рівні працюють концентратори (хаби) та повторювачі. Стандартними типами мережевих інтерфейсів фізичного рівня є: V.35, RS-232c, RS-485, Rg-11, Rg-45.

Деякі рівні в мережах можуть бути відсутні крім фізичного.

Найбільш поширеним протоколом обміну інформації в даний час є сімейство протоколів TCP/IP.

Сімейство протоколів TCP/IP має 3 транспортні протоколи:

1. TCP, який повністю відповідає OSI.
2. UDP – частково відповідає транспортному рівню.
3. SCTP – розроблений додатково для усунення деяких недоліків TCP.

ICMP – використовується для внутрішніх потреб (службовий).

Обмін інформацією у мережі визначається топологією, апаратно-фізичними засобами (інтерфейси RS-485...), протоколами обміну даними.

Нижній рівень АСУТП (датчики, виконавчі пристрої) об'єднуються за допомогою «сенсорної мережі».

На рівні датчиків:

- SensorBus;
- ASI;
- 1-Wire;
- InterBus-s;
- ProfiBus-DP;
- Sercos interface.

На рівні ПЛК:

- HART;
- CAN;
- FIP;
- LON.

Найчастіше на цьому рівні час циклу складає від 1мс до 1 с. Об'єм даних за один цикл складає $1 \div 8$ байтів.

2.3. Основні інтерфейси

Інтерфейси RS-485, RS-422, RS-232. RS (Recommended Standart).

Першим з'явився RS-232 і він дозволяв з'єднувати між собою два комп'ютери.

RS-485 введений в дію в 1983 році. Є найбільш поширеним в промисловій автоматизації та використовується в Modbus, Profibus, DP, ARCNET, Bitbus, WorldFip, LON, Interbus.

Перевагами RS-485 є:

- 1) Двосторонній обмін даними по одній витій парі;
- 2) Велика довжина лінії;
- 3) Робота з декількома трансіверами (прийомо-передатчиками).

В основі інтерфейсу лежить диференціальний спосіб передачі сигналу, коли напруга, що відповідає «1» відраховується не від рівня «землі», а вимірюється як різниця потенціалів між двома проводами.

Різниця напруг $0,2 \div 12$ В відповідає «1», $-0,2 \div -7$ В – «0».

Завада може мати достатньо високий рівень до 1,3 В. Завдяки симетрії лінії відносно землі завади в обох провідниках близькі за формою та величиною. В приймачі сигнал виділяється шляхом віднімання напруг двох провідників, тому напруга завад практично рівна нулю.

Для мінімізації електромагнітних завад використовуються провідники типу «вита пара». Ступінь компенсації визначається кількістю витків на одиницю довжини, якістю та типом кабелю.

RS-485 дозволяє мати до 32 передавачів (Master) та до 32-х приймачів (Slave). Середня швидкість $90 \div 500$ кбод. Максимальна швидкість 30 Мбод.

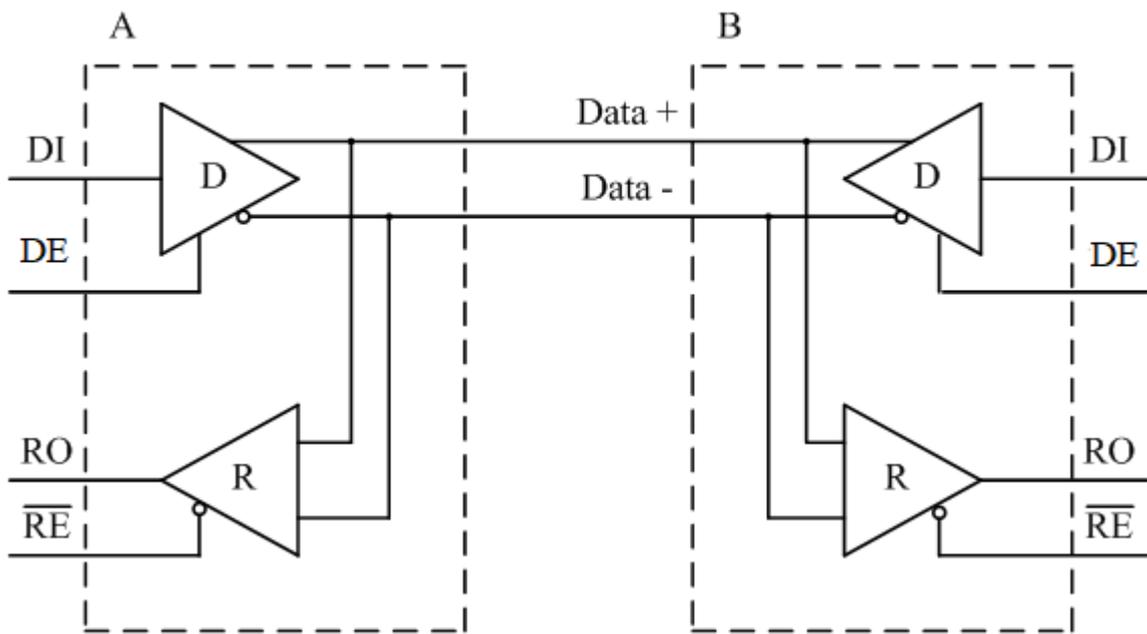


Рисунок 2.2 – З'єднання двох пристроїв з інтерфейсом RS-485 за двопровідною схемою

На рис. 2.2. позначено:

D – передавач;

R – приймач;

DI – вхідний інтерфейс, дані для передачі;

DE – переводить передавач в 3-й високоомний стан;

Двопровідна схема забезпечує напівдуплексний режим. Існує також чотирипровідна схема з повним дуплексом.

RS-232 введено в 1962 році, нова версія з'явилася в 1969 р. RS-232C, а у 1987 р. – RS-232D. Інтерфейс призначений для обміну інформацією між двома пристроями на відстань до 15 м за схемою «точка-точка». Спосіб передачі однофазний. Максимальна швидкість передачі 460 кбод.

Для передавача:

«0» - 5÷15 В

«1» - -5÷-15 В

Для приймача:

«0» - 3÷25 В

«1» - -3÷-25 В

Для зв'язку використовується кабель з 9-ма або 25-ма контактами (DB-9 або DB-25).

Для підвищення дальності передачі створено інтерфейс RS-422. Спосіб передачі диференціальний.

Рівень напруги $\pm 7\text{В}$;

Максимальна довжина кабелю 1200 м;

Схема «точка-точка» з можливістю багатоабонентної доставки (один передавач та до 10 приймачів).

2.4. Промислові мережі рівня датчиків та виконавчих пристроїв

Мережа Activator Sensor Interface (ASI) створена в 1989 році. Активно почав впроваджуватися з 1992 року. Найчастіше використовується в машинобудуванні. Він призначений для підключення масиву простих пристроїв вводу/виводу (наприклад, дискретних пристроїв, таких як приводи, датчики, декодери, аналогові входи і виходи, кнопки та датчики положення клапана) у дискретному виробництві та збору інформації з використанням одного двожильного кабелю.

Є «відкритою» технологією. Специфікація розроблена і підтримується провідними виробниками систем автоматизації. Топологія мережі – довільна. Версія AS-і 2.0 дозволяє передавати аналогові сигнали. Найновішою версією специфікації є AS-і 3.0.

До мережі AS-інтерфейсу можна підключати датчики і виконавчі пристрої як із вбудованим AS-інтерфейсом, так і без нього – через спеціальні модулі вводу-виводу. Послідовне опитування датчиків і видача команд на виконавчі механізми здійснюються одним ведучий пристрій, наприклад ПЛК, який має AS-інтерфейс. Крім того, ведучий пристрій має інтерфейс Fieldbus для підключення до промислової шини верхнього рівня. Для цієї шини мережа з ASI-інтерфейсом є одним з підключених до шини пристроїв вводу-виводу.

Варіант **ASI-Safe** використовується для систем з підвищеною безпекою.

Для під'єднання датчиків розроблено спеціальний плоский кабель з підключенням під прокол ізоляції (ножові клєми, передбачені конструкцією модулів вводу-виводу), як показано на рис. 2.3.

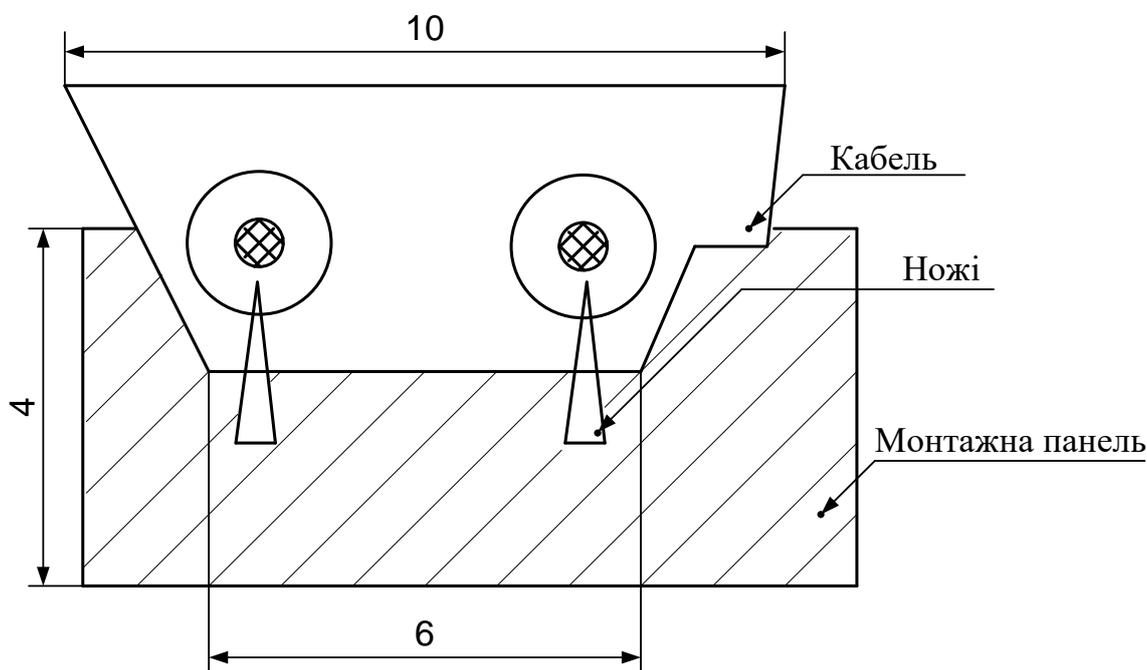


Рисунок 2.3 – Під'єднання датчиків спеціальним плоским кабелем під прокол ізоляції

По жилам кабелю одночасно здійснюється живлення датчиків та передача сигналів. Напруга живлення становить 24 В постійного струму. Споживання становить 150 мА на одну одиницю. Час циклу складає 5 мс, швидкість передачі 166 кбіт/с. Довжина кабелю – до 100 м, з підсилювачем – до 200 м. Число введених одиниць – до 31. Технологія передачі даних – Master-Slave. Кодування (технологія передачі) – манчестерський код (0-1). Логічна «1» →↑ – перепад сигналу з низького на високий. Логічний «0» – перепад з високого на низький. Максимальний об'єм даних з одного ASI-вузла – 4 біти.

Еволюцію підключення датчиків показано на рис. 2.4. Спочатку при розробці систем комп'ютерної автоматизації датчики і виконавчі пристрої підключали до ПЛК через вбудовані в контролер пристрої вводу-виводу. На наступному етапі розвитку пристрої вводу-виводу перенесли до датчиків і

виконавчих пристроїв і підключили до шини промислової мережі, до якої приєднували і ПЛК. Для живлення датчиків і виконавчих пристроїв використовували окремі провідники. З появою ASI датчики, виконавчі пристрої і контролер із вбудованим AS-інтерфейсом стали об'єднувати без проміжних пристроїв.

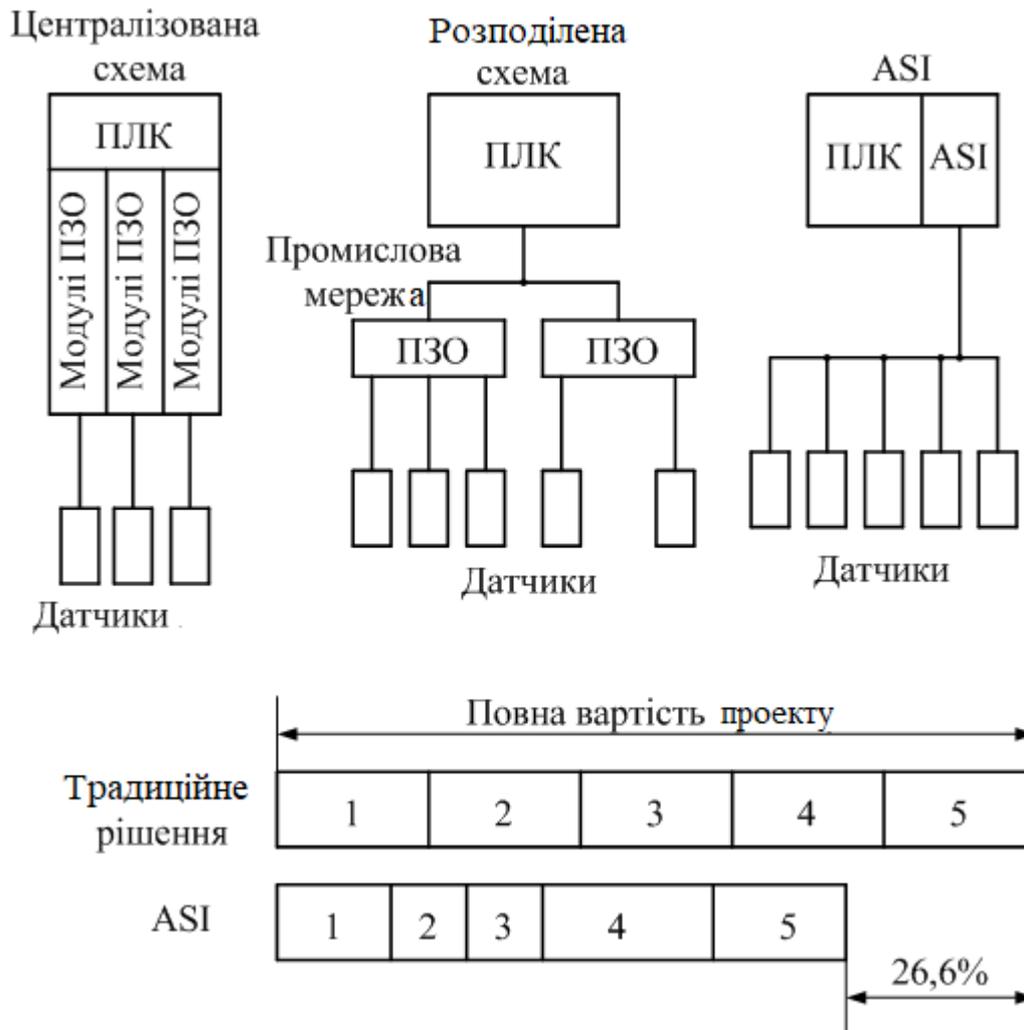


Рисунок 2.4 – Еволюція підключення датчиків

На рис. 2.4 позначено:

1. Контролери;
2. Кабель;
3. Прокладка кабелю;
4. Підключення пристроїв;
5. Периферійні пристрої.

Для збільшення довжини інформаційної мережі застосовуються повторювачі. Варіанти топології ASI-мережі з двома повторювачами показана на рис. 2.5, де позначено: ДЖ – джерело живлення.

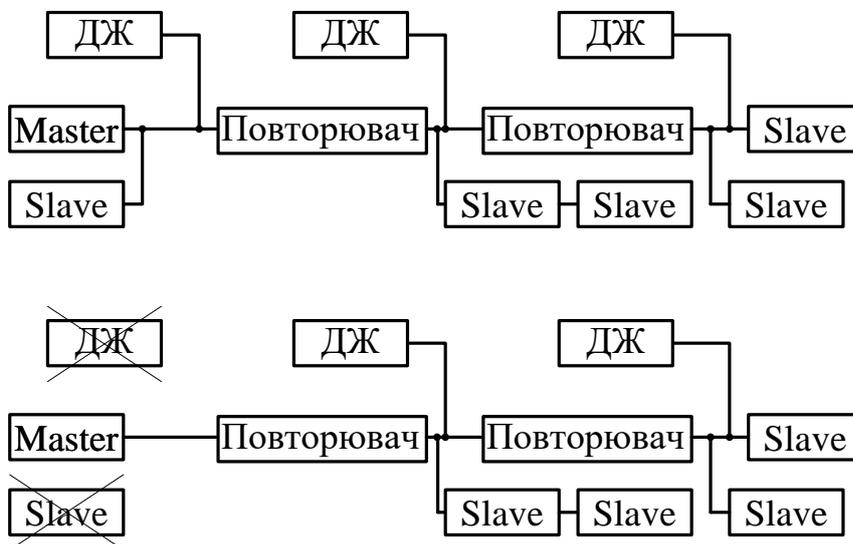


Рисунок 2.5 – Схема ASI-мережі з двома повторювачами

Структура кадру представлена на рис. 2.6. Швидкодія становить 1 біт за 6 мкс.

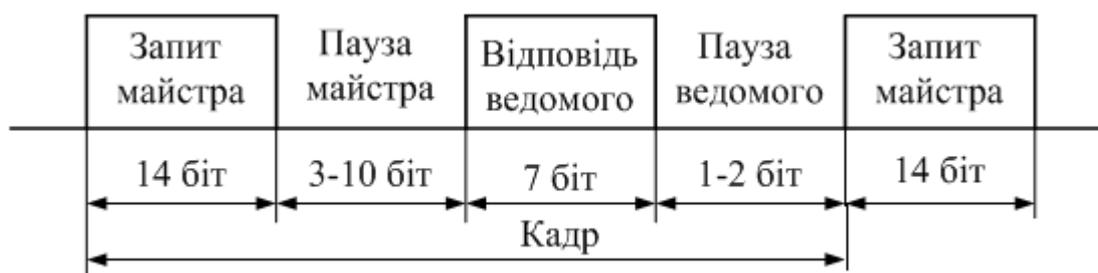


Рисунок 2.6 – Структура кадру ASI-мережі

Структура ASI-протоколу передачі інформації показана на рис. 2.7, на якому позначено:

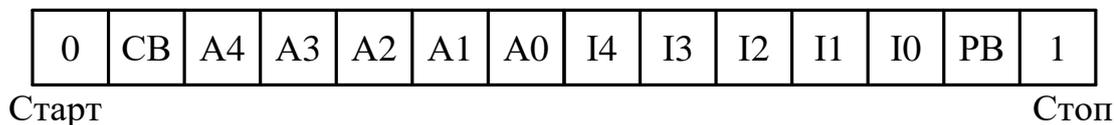
СВ (Control Bit): 0 – обмін параметрами, даними, адресами; 1 – команда;

A0÷A4 – адреса введеного (Slave);

I0÷I4 – інформація для запиту або відповіді;

PB – перевірка правильності передачі по біту парності (Parity Check).

Запит майстра



Відповідь введеного

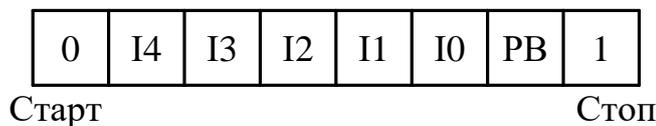


Рисунок 2.7 – Структура ASI-протоколу передачі інформації

Мережа 1-Wire (один провід) розроблена компанією Dallas Semiconductor Corp. Близько 60 % систем автоматизації не потребують високої швидкодії, на які розраховано 1-Wire. Інтерфейс низькошвидкісний – 16,4 кбіт/с, в режимі Overdrive – 125 кбіт/с. Довжина кабелю типу «вита пара» до 300 м. Швидкість обробки команд 75 вузлів/с. Живлення від зовнішнього джерела 2,8÷6 В або без джерела живлення на основі схеми «паразитне живлення» (паралельно підключається конденсатор на 800 пФ). Топологія – шина.

1-Wire в автоматизації найчастіше використовується для багатоточкового контролю температури в теплицях, елеваторах, інкубаторах тощо, а також для сигналізації в охоронних системах.

Мережа пристроїв 1-Wire із назначеним пристроєм на роль майстра називається MicroLAN. Відмінною особливістю цієї шини є можливість використання лише двох проводів: для даних і «землі». Аби забезпечити це, пристрої 1-Wire мають конденсатор у 800 пФ для збереження заряду і живлення пристрою під час періоду активності лінії даних. Для ПЛК на борту необхідно

мати відповідні інтерфейси. Для підключення мережі до ПК необхідно мати перетворювач інтерфейсів на Com, USB або LPT.

В залежності від своєї функції пристрої 1-Wire доступні у вигляді окремих компонентів для інтегральних мікросхем або у портативній формі, що називається *iButton* у вигляді акумулятора для годинника. Кожен чип 1-Wire має унікальний код ідентифікатора. Ця особливість робить їх корисними для використання як ключа для замків, сигналізації, автентифікації користувачів в комп'ютерних системах тощо.

Інтерфейс «струмова петля» використовується ще з 1950 року. Струм у петлі спочатку був $0\div 60$ мА. З 1962 року перейшли на $0\div 20$ мА, а з 1980-х років на $4\div 20$ мА.

Переваги:

1. Є можливість перевірки роботи мережі (цілісність електричного кола);
2. Можливість подачі живлення до датчика.

Функціональна схема інтерфейсу «струмова петля» показана на рис. 2.8.

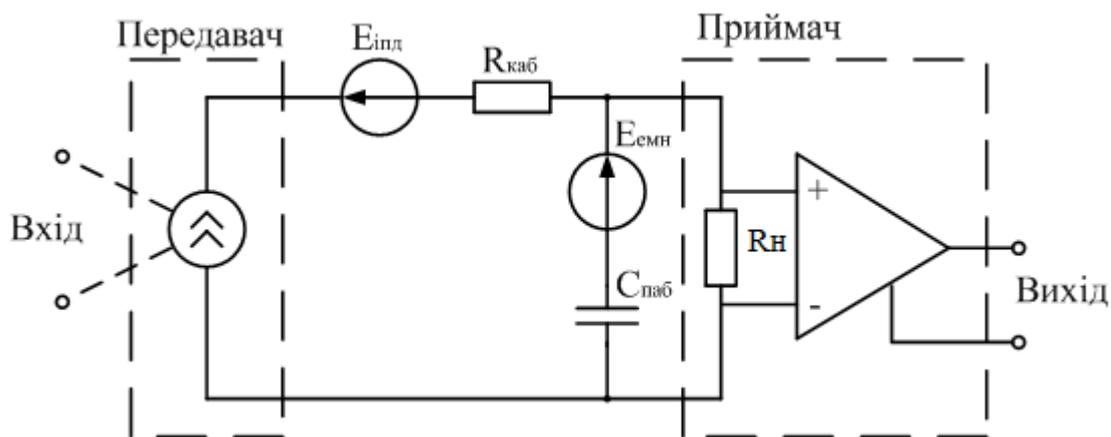


Рисунок 2.8 – Функціональна схема «струмова петля»

Перевага: На корисний струмовий сигнал не впливає ЕРС індуктивних завод $E_{інд}$, електричний опір кабелю $R_{каб}$ та електричний опір навантаження $R_{н}$.

Недолік: Корисний струмовий сигнал може зменшуватися за рахунок ємнісною завади $E_{\text{емн}}$, яка прикладається паралельно. Для її послаблення потрібно використовувати екрановану виту пару. Основним недоліком є невисока швидкодія, обумовлена наявністю ємності кабелю $C_{\text{каб}}$. При типовій погонній ємності кабелю 75 пФ/м та при довжині кабелю 1 км для її заряду до 5 В необхідно час у 19 мкс, що визначає швидкість передачі 9 кбт.

На стороні приймача струмовий сигнал перетворюється в сигнал напруги за допомогою каліброваного резистора $R_{\text{н}}$, який дозволяє отримати стандартний ряд напруг: 2.5, 5, 10 В при струмі 20 мА відповідно при $R_{\text{н}}=125, 250$ та 500 Ом.

Інтерфейс «струмова петля» використовується у двох версіях:

- 1) Аналогова петля – для передачі сигналів між датчиками та контролерами з високою точністю.
- 2) Цифрова петля.

Популярність петлі почала падати з появою RS-485 у 1983 р. Друге життя «струмова петля» знайшла у протоколі «HART».

HART-протокол (Highway Addressable Remote Transducer Protocol) створений у 1975 році з метою додати можливість керування інтелектуальними пристроями на основі відомої струмової петлі. Для цього на аналоговий струмовий сигнал накладається високочастотний синусоїдний сигнал для кодування «0» та «1»:

«0» кодується частотою $f=2200$ Гц.

«1» кодується частотою $f=1200$ Гц.

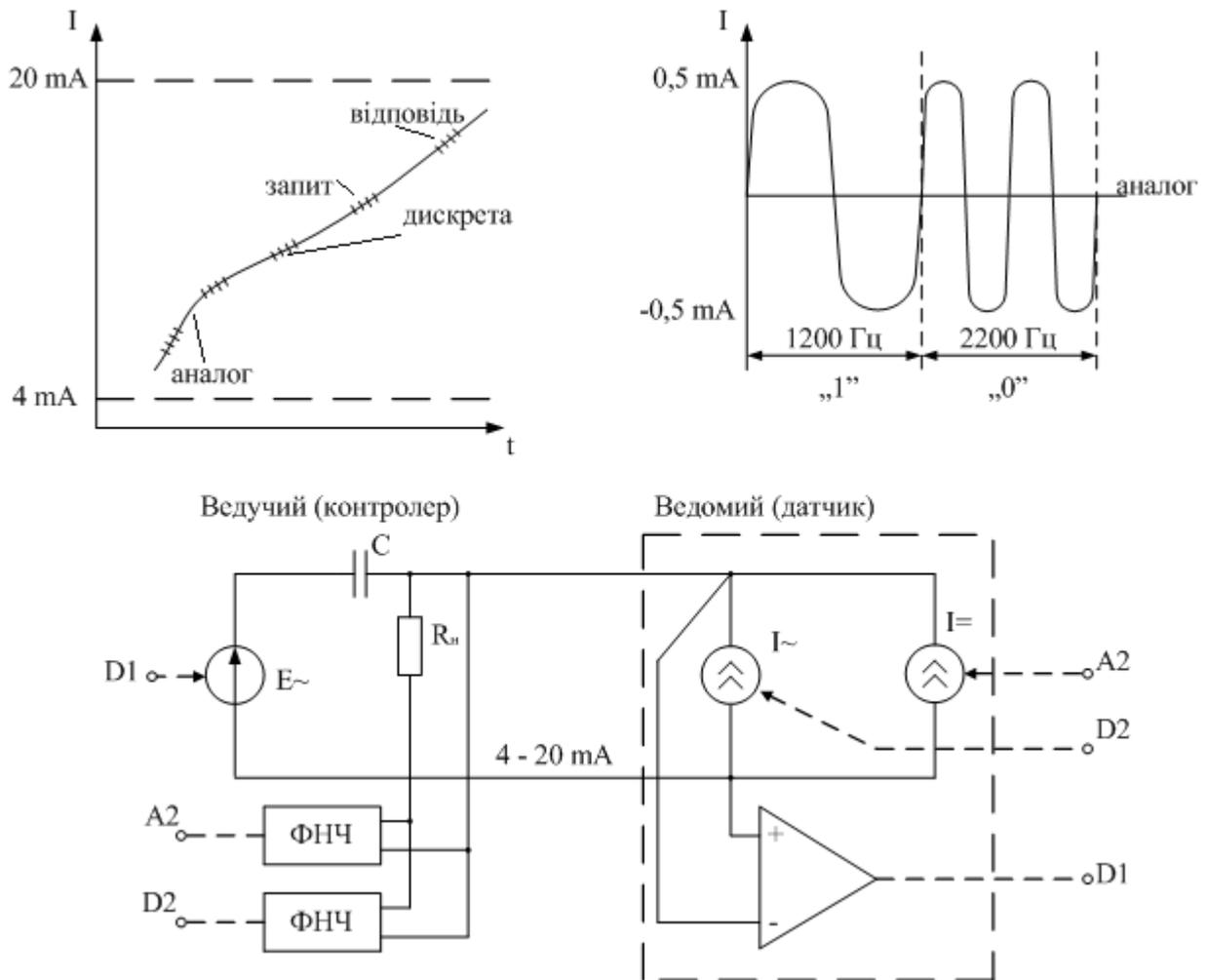


Рисунок 2.9 – Принцип роботи HART-протоколу

ФНЧ – фільтр низьких частот з $f_{\text{зрізу}}=10$ Гц, виділяє аналоговий сигнал з мережі.

ФВЧ – фільтр високих частот з $f_{\text{зрізу}}=400\div 800$ Гц, виділяє дискретний сигнал з мережі.

Перевага:

- 1) Висота завадостійкість;
- 2) Простота та низька вартість монтажу;
- 3) Дешевизна.

Недолік:

- 1) Невисока швидкість передачі;

- 2) Необхідність наявності на борту контролера HART-мікросхеми та відповідного програмного забезпечення.

Типова область застосування – дорогі інтелектуальні пристрої: електричні магнітні канали, датчики потоку рідини, датчики вимірювання рівня рідини.

Схема з'єднання: точка-точка або мережа з багатьма вузлами, в якій передаються тільки дискретні сигнали.

Може бути два майстра.

Довжина лінії: до 3 км.

Кількість вузлів: до 15.

Топологія: зірка, шина (для дискретної).

Кабель: вита пара або виділений телефонний канал.

В останніх версіях може використовуватися не тільки струмова петля, а і RS-485.

2.5. Промислові мережі рівня контролерів

CAN-мережа (Control Area Network) призначена для організації високонадійних та недорогих каналів зв'язку у розподілених системах керування. Спочатку створювалася для автомобільної промисловості фірмою Robert Bosch GmbH в 1983 році. Зараз використовується для промислової автоматизації та в технологіях «розумний будинок».

Перевага:

- 1) Робота в режимі жорсткого реального часу;
- 2) Висока стійкість до завад;
- 3) Надійна передача даних.

Недолік:

- 1) Довжина мережі зворотна до швидкості передачі;
- 2) Дуже великий обсяг службової інформації в пакеті даних;
- 3) Немає єдиного стандарту на протоколи високого рівня.

Кабель витой пари повинен мати третій загальний провід. Топологія шина. При передачі даних застосовується метод широкого мовлення, жоден вузол не має адреси, проте вузли мають конкретні ідентифікатори.

Найбільша довжина мережі при швидкості:

40 м – 1 Мбіт/с;

100 м – 500 Мбіт/с;

500 м – 125 Мбіт/с;

5000 м – 10 кбіт/с.

Profibus (Process Field Bus) – незалежний від виробника відкритий стандарт польової шини для широкого спектра застосувань у виробничій автоматизації. Profibus об'єднує технологічні і функціональні особливості послідовного зв'язку рівня польової шини. Вона дозволяє об'єднувати розрізнені пристрої автоматизації в єдину систему на рівні датчиків і приводів. Мережа Profibus – це комплексне поняття, вона базується на декількох стандартах і протоколах. Спочатку стандарт було прийнято у 1987 році у Німеччині.

При побудові систем багаторівневої автоматизації гостро стоїть задача обміну інформації між рівнями в залежності від різних вимог:

- на рівні датчиків – швидкий рівень обміну даними;
- для ПЛК – середня швидкість зі значними об'ємами;
- для небезпечних виробництв – вимога до безпечної передачі інформації.

Під іменем Profibus є три окремі протоколи:

- Profibus DP;
- Profibus FMS;
- Profibus PA.

Для обміну інформацією використовується вита пара від 100 до 1200 м у залежності від потрібної швидкості передачі даних або оптокабель до 15 км,

застосовуються модулі для організації інфрачервоного каналу до 15 м.
Топологія: зірка або шина.

Доля Profibus становить близько 40 % промислових мереж Європи.

Profibus DP – спроектовано для організації швидкого каналу зв'язку з рівнем датчиків. Швидкість обміну залежить від довжини мережі:

100 кбіт/с – 1200 м;

12 Мбіт/с – до 100 м.

В основі алгоритму лежить модель циклічного опитування каналів. В DP-протоколі є 3 типи пристроїв:

- 1) Майстер класу 2 – може виконувати функції конфігурування та діагностики пристроїв мережі;
- 2) Майстер класу 1 – виконує функції ведучого вузла;
- 3) Slave – ведомий пристрій.

Сам протокол може організовувати мономайстрову структуру до 126 Slave та багато майстрову конфігурацію. Інтерфейс RS-485.

Profibus FMS – призначений для роботи на рівні цеху. Висока ступінь функціональності важливіша за швидкодію. Протокол допускає гібридну структуру з'єднання вузлів, які мають назву віртуальний пристрій. Інтерфейс RS-485.

Profibus PA – застосовується для автоматизації небезпечних виробництв. Використовується система схожа на струмову петлю із забезпеченням передачі дискретних та аналогових сигналів.

Мережа Modbus – найбільш розповсюджена мережа та протокол у світі, хоча з'явився ще у 1979 році. Остання версія від 2006 р. Система повністю відкрита з безкоштовним програмним забезпеченням. На теренах СНД конкурує тільки з Profibus.

Перевага: Не потребує спеціальних інтерфейсних контролерів. Простота організації та дешевизна.

Недолік: Обмін за методом «ведучий-ведомий» не дозволяє ведомим передавати інформацію при її зміні, що обумовлює інтенсивний обмін інформацією.

Фізичний інтерфейс: RS 485 – двопровідний, RS 422, RS 232. Топологія шина або зірка. Один Master і до 247 Slave.

Топологія шина: один магістральний кабель, який складається з трьох провідників в екрані (2 провідники – це вита пара, по яким передаються дані, а 1 провідник – «земля»). Екран та земля повинні бути заземлені в одній точці.

Пристрої до шини можуть підключатися трьома способами:

- безпосередньо до кабелю;
- через пасивний T-конектор;
- через активний повторювач.

Стандартні швидкості обміну:

- 19200 біт/с (за згодою);
- 9600 біт/с;
- 4800 біт/с;
- 2400 біт/с;
- 1200 біт/с.

Якщо швидкість 9600 біт/с, то довжина кабелю до 1 км. При підключенні пристрою до магістрального кабелю відвід не повинен бути довше 20 м.

Варіанти розширення протоколу:

Modbus Plus – дозволяє мати декілька майстрів;

Modbus TCP – для узгодження та застосування з мережею Ethernet та Internet.

2.6. Промислові мережі операторського рівня

Промисловий Ethernet. Основи офісного Ethernet з'явилися ще у 1973 році. В теперішній час під Ethernet розуміють сімейство продуктів, які відповідають стандарту IEEE 802.3. В даний час промисловий Ethernet росте на 51% за рік.

Перевагами, що обумовили впровадження Ethernet в промисловість є:

- 1) Висока швидкість передачі до 10 Гбіт/с;
- 2) Простота інтеграції з інтернетом та інтранетом;
- 3) Простота інтеграції з офісними мережами;
- 4) Відкрита система;
- 5) Можлива організація багатомайстрових мереж.

Недолік: Відносно висока ціна, яка приблизно вдвічі більша від Modbus.

Є 4 стандартні швидкості передачі даних по витій парі та оптоволокну:
10 Мбіт/с, 100 Мбіт/с, 1 Гбіт/с, 10 Мбіт/с.

Ethernet має декілька специфікацій з наступною структурою:

<Швидкість передачі>BASE-<Додаткові позначення>

«Швидкість передачі» вказується цифрами, а при швидкості в Гбіт/с додається буква G.

«Додаткові позначення» мають на ступні значення:

T (Twisted pair) – вита пара;

F (Fiber optic) – оптоволоконна;

S (Short wavelength optic) – короткохвильовий оптокабель;

L (Long wavelength optic) – довгохвильовий оптокабель;

C (short Copper cable) – короткий мідний кабель;

X – наявність блоку кодування на фізичному рівні.

Приклади специфікацій:

10BASE-T – швидкість 10 Мбіт/с по витій парі;

100BASE-FX – швидкість 100Мбіт/с по оптоволокну з наявністю кодування.

Час реакції офісного (стандартного) Ethernet не перевищує 100 мс, проте він не є детермінованим (наперед відомим). В локальних мережах з невеликою кількістю пристроїв час може бути 20 мс при протоколі TCP/IP. Використовуючи протокол UDP можна зменшити час до 10 мс, а при використанні MAC – до 1 мс.

Для промислового використання Ethernet час реакції повинен бути детермінованим, тобто наперед відомим. Для вирішення цієї проблеми застосовуються комутатори, які ділять мережу на маленькі сегменти, а в сегменті дозволяють однозначно встановлювати зв'язок між двома вузлами.

Вдосконалення мережних засобів дало змогу широко застосовувати виту пару як середовище передавання. В рамках стандарту Ethernet створені специфікації 10BASE-T, що використовує дві неекрановані виті пари UTP (Unshielded Twisted Pair) 3,4 або 5 категорій, та 100BASE-T4, що ґрунтується на чотирьох витих парах UTP 5 категорії або екранованій витій парі STP (Shielded Twisted Pair). Для зв'язку між вузлами мережі необхідними є дві виті пари провідників: одна – для передавання, інша – для приймання інформації. Звичайно, замість двох кабелів по одній парі витих провідників у кожній використовують один кабель з чотирма парами провідників. Окрім економії та технічних переваг, це створює можливість переходу на більш швидкісні мережні архітектури без заміни самого кабелю. Фізична топологія – зірка: кожен вузол мережі з'єднується зі своїм портом кабельного центру кабельним променем, що не повинен перевищувати довжини 100 м. На кінцях кабелю 8-контактні роз'єкти RJ-45. Найбільш поширеними є 8-ми та 16-ти портів кабельні центри, що комплектуються зовнішніми адаптерами електромережі. Один з портів призначається для з'єднання з наступним кабельним центром (перехрещеними парами провідників). Більшість кабельних центрів мають також роз'єми для під'єднання тонкого коаксіального кабелю, що дає змогу гнучко комбінувати фізичну топологію мережі Ethernet та обидва найбільш поширених типи кабелю. Найбільш вразливе місце Ethernet на витій парі – кабельний центр, вихід з ладу якого паралізує всі вузли мережі. Слід відмітити основні характеристики та переваги витої пари:

- 1) Фізична топологія – зірка;
- 2) Максимальна довжина променя – 100 м;
- 3) До кожного вузла під'єднується лише один кабель;
- 4) Пошкодження кабелю виводить з ладу лише один мережний вузол;

5) Несанкціоноване прослуховування пакетів у мережі ускладнюється.

Особливість промислового Ethernet:

- 1) Індустріальне кліматичне виконання елементів IP67;
- 2) Відсутність вентиляторів в обладнанні;
- 3) Стійкість до вібрацій;
- 4) Захист від електростатики , електромагнітних імпульсів;
- 5) Кріплення на DIN-рейку;
- 6) Можливість живлення від напруги 10-30 В;
- 7) Зручне підключення кабелів.

Мережа Profinet (Profinet =>Profibus + Ethernet) – це сучасний стандарт, який забезпечує просту інтеграцію пристроїв Profibus. ASI, Interbus с промисловим Ethernet.

Profinet як стандарт промислового Ethernet за стандартами IEC 61158 та IEC 61784, розроблений і опублікований організацією PROFIBUS International і розвивається компанією Siemens AG і Profibus User Organization. Він регламентує вимоги до устаткування і побудови мережі для передачі даних із швидкістю 100 Мбіт/с в умовах підвищених температурних, механічних і електромагнітних впливів. Взаємодію мереж Profibus. ASI, Interbus, Devicenet з Profinet забезпечують спеціальні пристрої шлюзування (проху). Для конфігурування і діагностики в Profinet використовують протоколи TCP/IP та UDP. Стандарт Profinet визначає топологію мережі, вимоги до з'єднувачів і прокладки кабелю. Виробники устаткування отримали єдині вимоги для інтерфейсів, а кінцеві користувачі – простий монтаж мережі.

2.7. Бездротові промислові мережі

Існує багато об'єктів автоматизації, де бажане застосування бездротових мереж:

- 1) Датчики та виконавчі пристрої на нерухомих частинках конвеєрів, млинів, міксерів, підшипники двигунів, на роботах, на тілі людини та тварин;
- 2) На контейнерах для перевозки вантажів;
- 3) Об'єкти, в яких не бажано погіршувати дизайн (свердління стін, пожежна охоронна сигналізація, датчики систем кондиціонування та обігріву, система «розумний будинок»);
- 4) Епізодичне програмування та діагностика, дистанційне зчитування показників лічильників;
- 5) Об'єкти з агресивним середовищем та під високою напругою, відстеження траєкторій транспорту, моніторинг (ліс, море, екологія);
- 6) Об'єкти, для яких вартість провідних систем перевищує вартість безпроводних при умові відсутності жорстких вимог до надійності та доставки інформації в реальному часі;
- 7) Об'єкти в вибухонебезпечних зонах.

Перевага бездротових мереж:

- 1) Зниження вартості встановлення датчиків;
- 2) Виключення профілактики та обслуговування кабельних мереж;
- 3) Зменшення витрат на прокладку мереж;
- 4) Зручна модернізація системи.

Недоліки в порівнянні з дротовими мережами:

- 1) Не гарантований час доставки інформації;
- 2) Гірша завадостійкість;
- 3) Менша надійність;
- 4) Обмежена дальність;
- 5) Різде зменшення пропускної здатності із збільшенням одночасно працюючих станцій;
- 6) Нижча безпечність передачі інформації.

Бездротові мережі діляться на 3 класи:

- 1) Сотові;

- 2) Бездротові LAN;
- 3) Бездротові мережі датчиків.

Проблеми при створенні бездротових мереж:

- 1) Залежність щільності потужності від відстані d

$$P(t) \approx P_t \left(\frac{d_0}{d} \right)^\gamma$$

де $\gamma = 2 \div 6$ – для бездротових систем в цілому;

$\gamma = 2 \div 3$ – для систем промислової автоматизації;

$P_t \rightarrow d_0$ – потужність заміряна на відстані d_0 .

- 2) Інтерференція хвиль;
- 3) Джерела завад;
- 4) Паразитний вплив сусідніх каналів;
- 5) Ефект Доплера;
- 6) Розряди статичної електрики;
- 7) Робота двигунів постійного струму.

При організації бездротових мереж існує також проблема електроживлення датчиків. Крім акумуляторів та батарейок розглядаються методи на основі сонячних панелей, повітряних трансформаторів, живлення електромагнітними хвилями, отримання енергії від продуктів згоряння.

В промисловій автоматизації найбільше розповсюдження знайшли три типи бездротових мереж Bluetooth, Zig Bee та Wi-Fi. Вони використовують не ліцензований ISM (Industrial, Scientific, Medical) діапазон 2,4 ГГц.

Для зменшення впливу завад застосовується ширококомвна передача. В бездротових мережах застосовуються два методи широкосмугової модуляції для передачі інформації:

- 1) З прямим розширенням спектра DSSS;
- 2) З перескоком з однієї несучої частоти на іншу FMSS.

DSSS (Direct Sequence Spread Spectrum). Один прямокутний імпульс замінюється послідовністю одинадцяти коротких імпульсів. Для полегшення знаходження сигналу в ефірі застосовується код Баркера для кодування «1» та «0»:

«1» 11100010010

«0» 00011101101

Приймач має копію кода Баркера, тому сигнал з ефіру легко виділяється навіть при пропуску імпульсів.

Перевагою DSSS є висока стійкість до вузькосмугових завад.

FHSS (Frequency Hopping Spectrum Spreading) використовує діапазон 2,4 ГГц, шириною 83,5 МГц, в якому виділяється 79 частотних смуг по 1 МГц, що не перехрещуються. Частота переходу з однієї несучої на іншу повинна бути не менша 4 Гц для Wi-Fi та 1,6 кГц для Bluetooth. FHSS забезпечує швидкість передачі 1 та 2 Мбіт/с.

Для покращення якості передачі застосовується адаптивний FHSS, який називається AFH. В ньому запам'ятовується частота, на якій були помилки передачі, і вона виключається із таблиці частот.

Bluetooth було спроектовано для бездротового з'єднання побутової техніки та офісного обладнання. Специфікація Bluetooth підтримується організацією SIG з 1998 р. Застосовується для запису програм ПЛК, зчитування інформації з накопичувачів та датчиків.

Bluetooth організовано у вигляді піко-мережі, в якій один ведучий пристрій Master взаємодіє не більше ніж з сімома ведомими. Ведомі можуть взаємодіяти між собою через Master. Декілька піко-мереж створюють розподілену мережу.

Трафік мережі організовано із часовим розділенням каналів та дуплексною передачею. Інтервал складає 625 мкс. Ведучі пристрої ведуть передачу на не парних інтервалах, а ведомі пристрої відповідають в парні інтервали. За один інтервал можна передати 366 біт.

В Bluetooth застосовується метод FHSS з переходом за випадковим законом 1600 разів за секунду. Швидкість передачі становить 433,9 кбіт/с. Кожен Bluetooth-пристрій має 48-бітну адресу. В основному потужність передавача складає 1 мВт. Хоча по стандарту є три класи:

- 1) до 100 мВт (до 100 м);
- 2) до 25 мВт (до 15 м);
- 3) до 1 мВт (до 5 метрів).

Zig Bee анонсовано у 2004 р. Мережа створювалась за критеріями малої дальності, низької ціни, низької швидкості і малих габаритів. Ідеально підходить для промислових датчиків. Діапазон 2,4 ГГц розділено на 11..26 каналів по 5 МГц кожний. Максимальна швидкість передачі становить 250 кбіт/с. Потужність передавача стандартом не встановлена. При потужності 1 мВт відстань у приміщенні становить 10 м, а при 10 мВт – 80 метрів у приміщенні та до 1 км в умовах прямої видимості без перешкод.

В Zig Bee мережі є три типи пристроїв:

- 1) Координатор – формує топологію. Може бути тільки один. Він відповідає за зв'язок з іншими мережами;
- 2) Маршрутизатор – проміжна ланка для передачі інформації між пристроями;
- 3) Кінцевий пристрій – передає дані маршрутизатору чи координатору.

Топологія – зірка, дерево, стільникова мережа. Адрес пристроїв складається із 16 біт.

Wi-Fi створено у 1997 р. як бездротове розширення мереж Ethernet. Швидкість передачі становить до 54 Мбіт/с. Стандартні значення 11 Мбіт/с та 54 Мбіт/с. Передача сигналів здійснюється двома методами FHSS та DSSS. Дальність зв'язку сильно залежить від зовнішніх умов, типу антени та потужності передавача. Типові значення в приміщенні становлять 100..200 м та декілька кілометрів на відкритій місцевості при потужності передавача 50..100 мВт. Ширина виділеного для Wi-Fi частотного діапазону становить 83,5 МГц, що дозволяє в ньому помістити три канали, які не перехрещуються.

Стандарт встановлює 3 варіанти топології мереж:

- 1) Незалежні базові зони обслуговування IBSS;
- 2) Базові зони обслуговування BSS;
- 3) Розширені зони обслуговування ESS.

Зона обслуговування – це логічно згруповані пристрої, які мають свій ідентифікатор.

IBSS - станції зв'язуються безпосередньо між собою без використання точки доступу і без можливості підключення до провідникової ланки. Всі пристрої працюють автономно.

BSS - станції спілкуються між собою через центральний вузол, який називається точка доступу.

ESS - об'єднання декількох BSS в єдину систему за допомогою провідникової мережі Ethernet.

В табл. 2.1 показано порівняння основних параметрів бездротових мереж.

Таблиця 2.1 – Порівняння трьох бездротових технологій

Параметр	Bluetooth	Zig Bee	Wi-Fi
Довжина, м	~ 10(50-100)	10	~100
Швидкість, Мбіт/с	0,723	0,25	до 54
Кількість вузлів	8	245	не обмежена
Потужність споживання енергії, мВт	10	1	50
Ціна/складність	10	1	20
Повторна передача	+	+	-/+
Основне призначення	зв'язок з периферійним обладнанням	мережа датчиків	бездротове розширення Ethernet

2.8. Обладнання та кабелі мережі

При проектуванні мережі виникають наступні проблеми:

- 1) Довжина відводів від шини перевищує допустиму;
- 2) Довжина самої мережі більша норми ;
- 3) Кількість пристроїв більше допустимого;
- 4) До мережі потрібно підключати пристрій, яких не має відповідного порту;
- 5) Необхідно об'єднати декілька різних мереж (з різними протоколами);
- 6) Електричний сигнал в мережі із збільшенням довжини кабелю послаблюється, що обмежую максимально-допустиму довжину мережі.

Вказані проблеми вирішуються за допомогою встановлення додаткового обладнання.

Перетворювачі інтерфейсів

Використовуються для роботи пристроїв з різними інтерфейсами або для зміни фізичного способу передачі інформації.

Перетворювач RS-232 на RS-485/422. Найчастіше із 10 клем інтерфейсу RS -232 при перетвореннях використовуються тільки 3:

- 1) TD – передача даних;
- 2) RD – прийом даних;
- 3) SG – сигнальне заземлення.

Типова структура перетворювача показана на рис. 2.10.

Перетворювач RS-232 в оптоволоконний інтерфейс. Перевагою даного перетворювача є:

- 1) Велика дальність передачі в багатомодовому каналі;
- 2) Нечутливість до електромагнітних завад;
- 3) Відсутність аварійних ситуацій при обривах/закорочення;
- 4) Вища пропускна здатність та менша кількість помилок;
- 5) Гальванічна розв'язка;
- 6) Захищеність від несанкціонованого доступу.

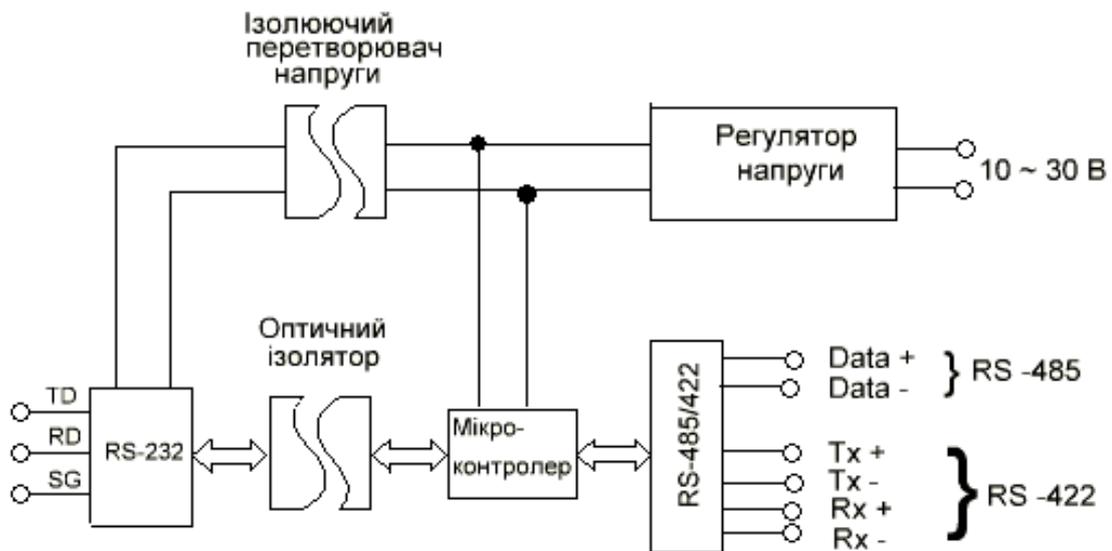


Рисунок 2.10 – Типова структура перетворювача RS-232 на RS-485/422

Перетворювачі USB в RS-232/485/422. Ці перетворювачі набагато складніші за попередні, бо не дозволяють побітову ретрансляцію. Для реалізації перетворення необхідно встановити необхідний драйвер, після чого з'являється віртуальний COM-порт.

Мережеві адаптери (Network Interface Card, NIC)

Це периферійні пристрої, що безпосередньо взаємодіють з середовищем передачі даних. Мережевий адаптер виконує функції фізичного і канального рівнів моделі OSI. Основні функції мережевих адаптерів:

- 1) Реалізація методу доступу до середовища передачі даних;
- 2) Кодування і декодування даних, – впізнавання кадрів, що приймаються;
- 3) Буферизація даних;
- 4) Гальванічна розв'язка комп'ютера і кабелю мережі.

В залежності від технології побудови мережі адаптери поділяються на адаптери Ethernet, Token Ring, FDDI тощо. Мережеві адаптери виконуються у вигляді окремої плати, що вставляється в системні слоти. Часто мережеві адаптери вбудовують в системні плати, що спрощує процес підключення комп'ютера до мережі.

Мережевий адаптер розраховується на один тип середовища передачі (наприклад – вита пара), але може підтримувати і кілька середовищ. Найбільш універсальними є так звані адаптери «Combo», які мають повний набір роз'ємів (BNC, RJ-45 і AUI).

Головним завданням мережевого адаптеру є прийом та передача даних. Ця функція поділяється між адаптером і його драйвером. В одних моделях адаптерів (як правило, що встановлюються на клієнтські машини) більша частина роботи з даними передається драйверу, а сам адаптер спрощується і здешевлюється, проте зростає завантаженість центрального процесора. В інших, які зазвичай встановлюються на серверні машини, використовується власний процесор, який самостійно виконує основну обчислювальну роботу.

Повторювач або репітер (Repeater)

Це комунікаційний пристрій, що відновлює послаблені сигнали (їх амплітуду і форму) і ретранслює їх в інший сегмент мережі з метою збільшення довжини мережі. Повторювачі не здійснюють ніякої інформаційної обробки сигналів, що надходять до них. Повторювачі використовували в мережах Ethernet на основі коаксіального кабелю. Повторювачі об'єднували в спільну шину кілька відрізків кабелю, до яких підключались абонентські станції. Повторювачі працюють на фізичному рівні моделі OSI. Кілька повторювачів (репітерів) можуть конструктивно об'єднуватися в спільному корпусі, утворюючи репітерний концентратор. Перевага такого об'єднання полягає в тому, що всі точки виявляються зібраними в одному місці, що спрощує реконфігурацію мережі, контроль і пошук несправності.

Концентратор або хаб (Hub)

Це спеціальний багатопортовий пристрій, основна функція якого полягає у повторі кадру з одного з портів на інші. Іноді концентратори називають також репітерними концентраторами, щоб відрізнити їх від комутуючих концентраторів – комутаторів. Тому, репітери можна розглядати як двопортові концентратори. Концентратори виконують ті ж функції, що й репітери, та працюють на фізичному рівні моделі OSI. Для кожної мережевої технології

(Ethernet, Token Ring, FDDI) використовуються свої концентратори. За конструкцією розрізняють концентратори наступних видів:

- 1) Концентратори з фіксованою кількістю портів – виконуються у вигляді окремого корпусу з фіксованою кількістю портів; (наприклад, на 8, 16, 24 порти).
- 2) Стекові або нарощувані концентратори виконуються у вигляді окремого корпусу з фіксованою кількістю портів, які мають спеціальні порти для об'єднання їх внутрішніх шин між собою. Внаслідок цього швидкість взаємодії між об'єднаними концентраторами є значно вищою, ніж при з'єднанні через звичайний порт;
- 3) Модульні концентратори на основі шасі мають спільне шасі з внутрішньою шиною, до якої підключаються модулі з фіксованою кількістю портів.

Міст (Bridge)

При досягненні деякого граничного значення числа вузлів мережі, тривалість затримок перед отриманням ними доступу до поділюваного середовища зростає і пропускна здатність мережі стрімко починає падати. Для вирішення цієї проблеми мережа розбивається на кілька сегментів, що об'єднуються за допомогою мостів. Міст – це комунікаційний пристрій, призначений для об'єднання мереж з різними стандартами обміну (наприклад, Ethernet і Token Ring), або кількох сегментів однієї мережі (наприклад, Ethernet). Міст ретранслює кадри з однієї мережі до іншої або з одного сегмента до іншого (як повторювач), але аналізує адресу їх призначення. Тобто, кадр транслюється в іншу мережу, або сегмент, лише в тому випадку, коли в цій мережі або сегменті, знаходиться адресат. В результаті за допомогою мостів мережа поділяється на кілька підмереж, внаслідок чого зменшується завантаженість середовища передачі даних. В основі роботи мостів лежить принцип прозорості. Він означає, що мережеві адаптери не здійснюють будь-яких додаткових зусиль для пересилки своїх кадрів через міст. Досягається така прозорість за рахунок того, що міст будує особливу адресну таблицю, на основі

якої і приймає рішення про необхідність ретрансляції кадрів. Механізм реалізації принципу прозорості моста полягає у наступному. Міст приймає всі кадри, що передаються по мережі, і записує їх до свого буферу, з якого вони надходять на подальшу обробку. Обробка кадрів здійснюється послідовно по мірі їх надходження. При цьому аналізується адреса відправника і адреса одержувача і якщо вони:

- 1) Містяться в адресній таблиці і належать різним сегментам – здійснюється ретрансляція кадру;
 - 2) Містяться в адресній таблиці і знаходяться в одному сегменті – кадр видаляється з буферу і нікуди не ретранслюється;
 - 3) Не містяться в адресній таблиці – кадр ретранслюється до всіх сегментів.
- Крім того, незнайомі адреси додаються до адресної таблиці.

Таким чином мости «самонавчаються», дізнаючись про розташування вузлів в сегментах. Потім міст передає кадри лише в місце призначення, зменшуючи загальний обсяг даних, що передаються через мережу. Мости працюють на каналному рівні моделі OSI. Як правило, мости підтримують не більше 4 портів.

Маршрутизатор (Router)

Маршрутизатор – це мережевий пристрій, що пересилає пакети даних між комп'ютерними мережами. Маршрутизатори виконують функції керування трафіком в Інтернеті. Пакет даних, як правило, пересилається з одного маршрутизатора на інший маршрутизатор через мережі, які складають мережу Інтернет, доки він не досягне свого кінцевого вузла. Маршрутизатор підключається до двох, або більше, ліній зв'язку з різних мереж. Коли пакет даних надходить на маршрутизатор, він зчитує інформацію про IP-адресу в пакеті, щоб визначити кінцевий пункт призначення. Потім, використовуючи інформацію у таблиці маршрутизації, або політику маршрутизації, направляє пакет до наступної мережі. Більш складні маршрутизатори, наприклад корпоративні, підключають великі мережі або мережі Інтернет до потужних маршрутизаторів, які передають дані на високій швидкості до волоконно-

оптичних магістралей. Хоча маршрутизатори, як правило, призначені для апаратних пристроїв, існують також маршрутизатори на базі програмного забезпечення.

Комутатор або свіч (Switch)

Комутатор (комутуючий концентратор) – багатопортовий комунікаційний пристрій, який дозволяє об'єднувати кілька сегментів в одну мережу, забезпечуючи її високу продуктивність і пропускну здатність. Комутатор може розглядатись як дуже швидкісний міст. Він дозволяє розділити мережу на кілька підмереж для збільшення допустимої довжини мережі і зниження навантаження в її частинах. На відміну від мостів, комутатори здійснюють не послідовну, а паралельну обробку кадрів, направляючи їх одночасно між усіма парами своїх портів. Ще однією відмінністю комутаторів, на відміну від мостів, є те, що вони не приймають, а в реальному часі розпізнають адресу приймача і перенаправляють кадри з одної частини мережі до іншої. При цьому ніякої обробки кадрів не виконується, тому комутатори практично не зменшують швидкість обміну в мережі. Проте вони не можуть перетворювати формат кадрів. Комутатори не ретранслюють колізії, на відміну від концентраторів. Також комутатори ведуть таблицю комутації, що ставить у відповідність порт комутатора та адресу відправника. Таблиця комутації формується в асоціативній пам'яті комутатора, коли приходить пакет, що містить адресу відправника. В початковий момент таблиця комутації порожня, а розсилка пакета дублюється всім пристроям в мережі. Якщо комутатор знає з таблиці адресу, то відсилає пакет за адресом і це скорочує час доставки пакетів. Найчастіше комутатори випускаються на 6, 8, 12, 16 і 24 порти. Для ефективної роботи при розбиванні мережі на частини за допомогою комутатора рекомендується дотримуватись правила «80/20»: щоб 80% передач здійснювалось в межах однієї частини (сегмента) і 20% передач має здійснюватися між різними частинами (проходили через комутатор).

Міжмережеві шлюзи (Gateways)

Дозволяють здійснювати обмін даними між різними мережами. Вони мають два спеціалізовані мережеві мікроконтролери, а також буферну пам'ять.

Мультиплексори

Дозволяють передавати по одному каналу декілька різних сигналів. Їх ефективно застосовувати, якщо пропускна здатність каналу на багато більша за продуктивність джерел даних.

Міжмережевий екран

Комплекс апаратних та програмних засобів для фільтрації інформації за певними критеріями.

Модем

Використовується для передачі даних по телефонним лініям, кабелям колективного телебачення, проводам передачі електроенергії, через супутник, радіоканал, сотовий канал.

Кабелі для промислових мереж

Кабелі визначають пропускну здатність та полосу пропускання сигналів. Полоса пропускання визначається відношенням частот, при яких потужність сигналу зменшується вдвічі. У більшості випадків для кодування сигналів застосовується зміна якогось параметру періодичного сигналу (частота, амплітуда). Полоси пропускання ліній зв'язку для популярних частотних діапазонів показано на рис. 2.11.

Типи кабелів: коаксіальний кабель, вита (скручена) пара, волоконно-оптичний кабель.

Коаксіальний кабель – це не найкращий вибір для сучасних мереж. Коаксіальний кабель (coaxial) має будову, зображену на рис. 2.12. Сигнал даних передається по центральній мідній жилі кабелю та зовнішньому мідному екрану, які відділені діелектричною оболонкою. Екран захищає також сигнал від електромагнітних полів. Навколо екрана є ізоляційна оболонка. Найпоширенішими у LAN коаксіальними кабелями є Thicknet (RG-8 або товстий Ethernet) з максимальною довжиною сегмента 500 м та Thinnet (RG-58

або тонкий Ethernet) з максимальною довжиною сегмента 185 м. Для приєднання до коаксіального кабелю використовують такі роз'єднувачі:

- 1) AUI (Attachment Unit Interface) – товстий Ethernet;
- 2) BNC (Barrel Network Connector) – тонкий Ethernet.

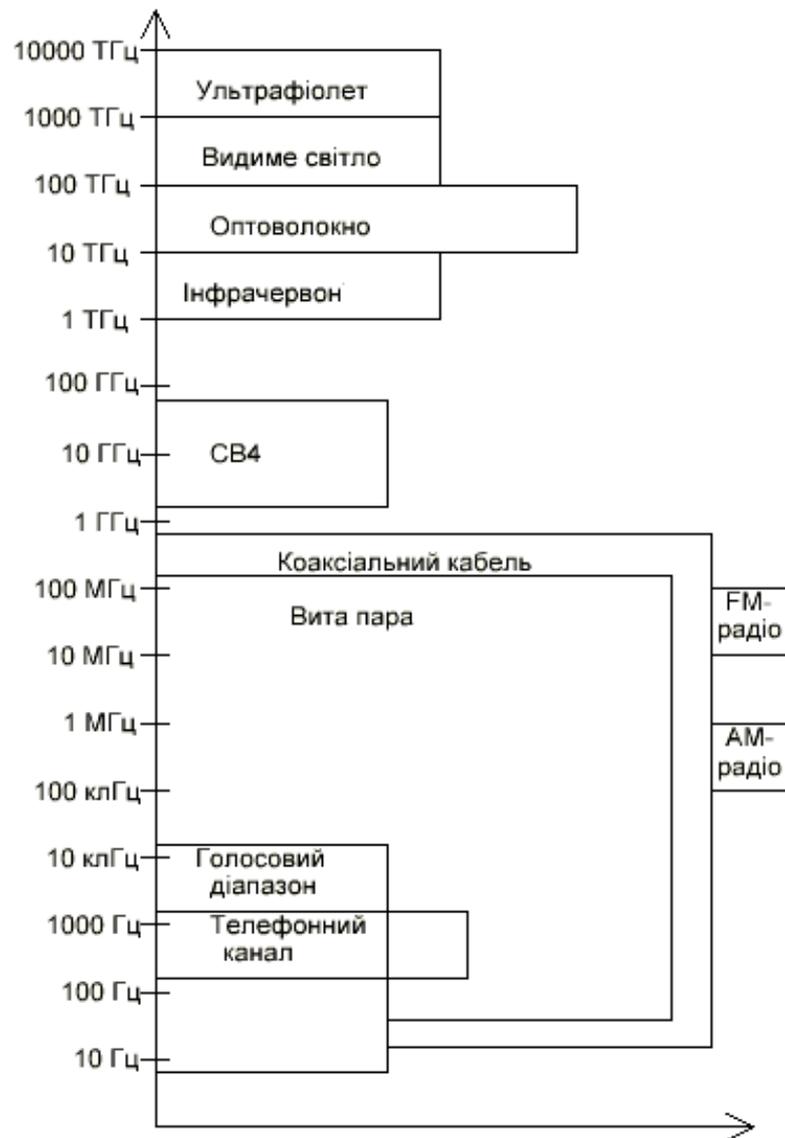


Рисунок 2.11 – Частотний діапазон

«Товстий» кабель створено для мережі Ethernet 10BASE-5, діаметр кабелю 12 мм, хвильовий опір 50 Ом, діаметр внутрішньої мідною жили 2,17 мм.

«Тонкий» кабель створено для мережі Ethernet 10BASE-2, діаметр кабелю 5 мм, хвильовий опір 50 Ом, діаметр внутрішньої мідною жили 0,89 мм.

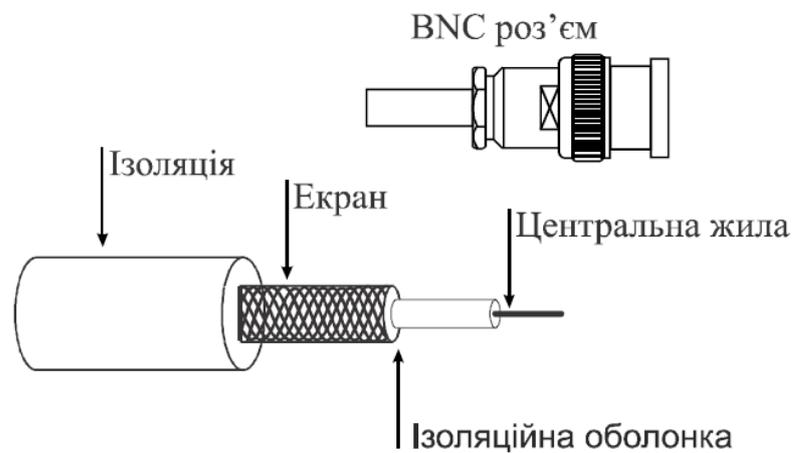


Рисунок 2.12 – Будова коаксіального кабелю

За техніко-експлуатаційними характеристиками розрізняють широко- та вузькосмугові коаксіальні кабелі. Широко- та вузькосмугові кабелі використовують для аналогового широко- та вузькосмугового передавання. Смуга перепускання такого кабелю, зазвичай, розділена на декілька аналогових каналів з різними частотами-носіями. Вона залежить від марки кабелю і може сягати 2-3 ГГц. Згасання сигналу на частоті 100 МГц не більше 7 Дб на 100 м. Термін придатності становить 10..12 років. Вузькосмугові кабелі застосовують для цифрового передавання. Вони мають швидкість передавання не більше 80 Мбіт/с, загасання сигналів на частоті 10 МГц становить 4 Дб на 100 км. Решта параметрів збігається з аналогічними в широко- та вузькосмугових кабелях. Сфера застосування коаксіальних кабелів невпинно звужується. У сфері магістральних сполучень їх витіснили волоконно-оптичні кабелі, які мають більшу смугу перепускання, менші втрати сигналу, а у локальних підсистемах – дешевша та простіша у прокладанні й експлуатації вита пара. Водночас широко- та вузькосмугові коаксіальні кабелі мають ширшу смугу перепускання, ніж вита пара, краще придатні для передавання широко- та вузькосмугового відеосигналу.

Вита (скручена) пара (twisted pair) – це найпоширеніше фізичне середовище передавання сигналів для локальних мереж. У порівнянні з коаксіальним кабелем стійкість до завад у пари нижча. Час поширення сигналу 8..12 нс/м. Загасання сигналу -28 дБ на 100 м при частоті 10 МГц, що більше, ніж у коаксіальному кабелі. Вита пара – це кабель, що складається з чотирьох витих пар. Кожен провідник має свій колір, кабель є симетричним. Канал, що використовує даний тип кабелю, найдешевший для прокладання.

Розрізняють декілька типів скручених пар (рис. 2.13):

- 1) UTP (Unshielded Twisted Pair) – незахищена скручена пара;
- 2) FTP (Foiled Twisted Pair) – фольгована скручена пара;
- 3) STP (Shielded Twisted Pair) – екранована скручена пара.

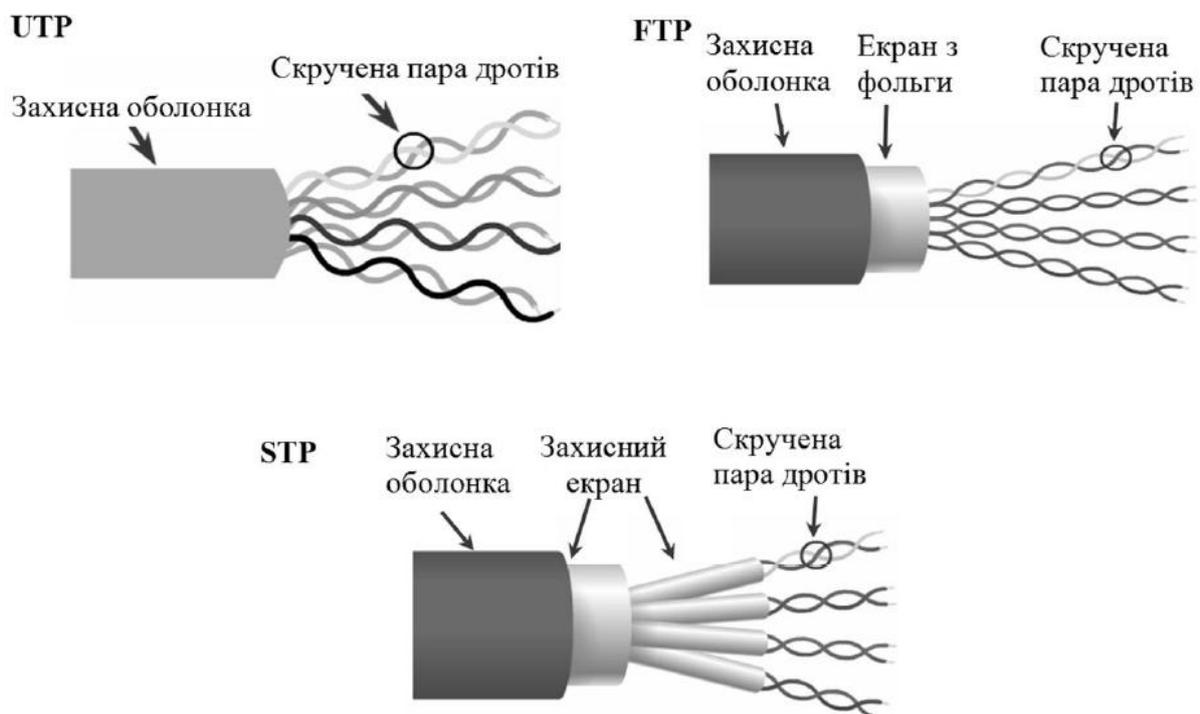


Рисунок 2.13 – Будова та типи виті пари

Вита (кручена) пара UTP – це вісім мідних дротів, скручених попарно в спільній ізоляції. Вона є найпоширенішою та найдешевшою скрученою парою,

проте при експлуатації виникають проблеми з електромагнітною сумісністю. У FTP та STP кабелях пари дротів мають спільний екран для захисту від електромагнітного випромінювання. У STP, окрім того, кожна пара дротів має окремий екран. Скручені пари STP та FTP мають ширший частотний діапазон передавання, менше електромагнітне випромінювання порівняно з UTP, однак вони дорожчі та складніші у прокладанні і монтажі.

За стандартами кабель поділяється на сім категорій:

- 1) *Категорія 1*: до 1983 р. це був основний тип кабелю. Застосовувався при швидкостях до 20 Кбіт/с;
- 2) *Категорія 2*: вперше застосована фірмою ІВМ для побудову власної мережі. Передає сигнали зі спектром до одного 1 МГц;
- 3) *Категорія 3*: стандартизована у 1991 р. Частотний діапазон до 16 МГц. Є основою мереж більшості будівель;
- 4) *Категорія 4*: покращений варіант категорії 3. Частота до 20 мГц. Має підвищену завадостійкість, на практиці застосовують рідко;
- 5) *Категорія 5*: розроблена для високошвидкісних протоколів, діапазон до 100 МГц, орієнтована на заміну категорії 3. Забезпечує Fast Ethernet;
- 6) *Категорія 6 та 7*. Категорія 6 до 250 МГц, 7 до 600 МГц. Кабель 6 може бути як з екраном, так і без нього, а кабель 7 має екран кожної пари та кабелю в цілому. Застосовується для реалізації високошвидкісних протоколів.

Крім кабелю «вита пара» є екранований провід «вита пара», який застосовується в середині будівель. Стандартом визначаються типи від 1 до 9. Тип 1 за характеристиками приблизно відповідає кабелю категорії 5.

Прокладка кабелів скрученої пари тиу UTP і STP закінчуються роз'ємом RJ-45 (рис. 2.14). Якщо дивитися на роз'єм спереду, направивши металеві контакти вгору, контакт номер 1 буде ліворуч, а номер 8 – праворуч.

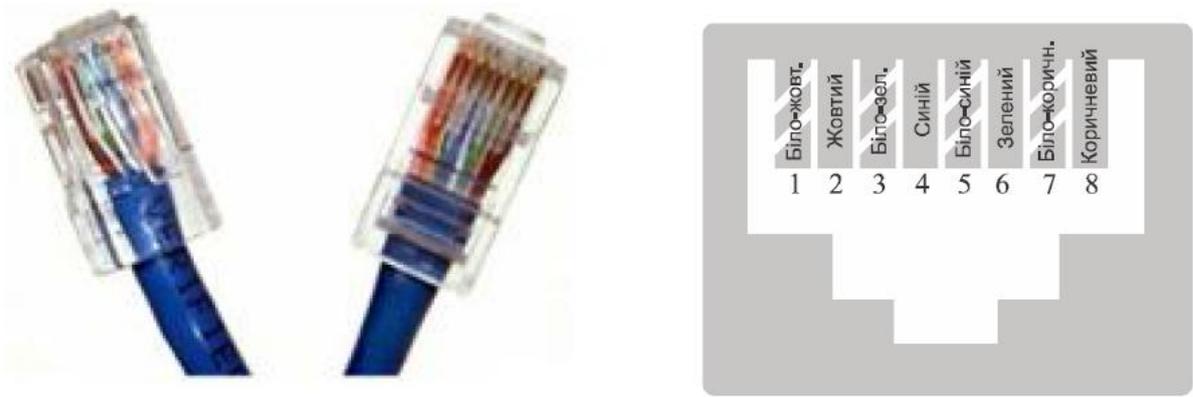


Рисунок 2.14 – Монтаж витії пари роз'ємом RJ-45

Волоконно-оптичний кабель (Optical fiber) (ВОК) складається із тонких, гнучких, скляних волокон діаметром від 5 до 60 мікрон. Будову ВОК показано на рис. 2.15. У центрі розташована серцевина – світлопередаюче середовище, що виготовлене з прозорого матеріалу. Матеріали серцевини у порядку зниження якості: одномодове кварцове скло, градієнтне скло, силікатне скло з пластиковим покриттям, пластик. У даний час широко розповсюджені пластикові оптичні волокна. Серцевину такого волокна виготовляють з поліметилметакрилату (РММА), а оболонку з фторованих РММА (фторполімерів). ВОК позначають як відношення діаметрів серцевини та оболонки (у мкм), наприклад, 9/125, 50/125, 62.5/125.

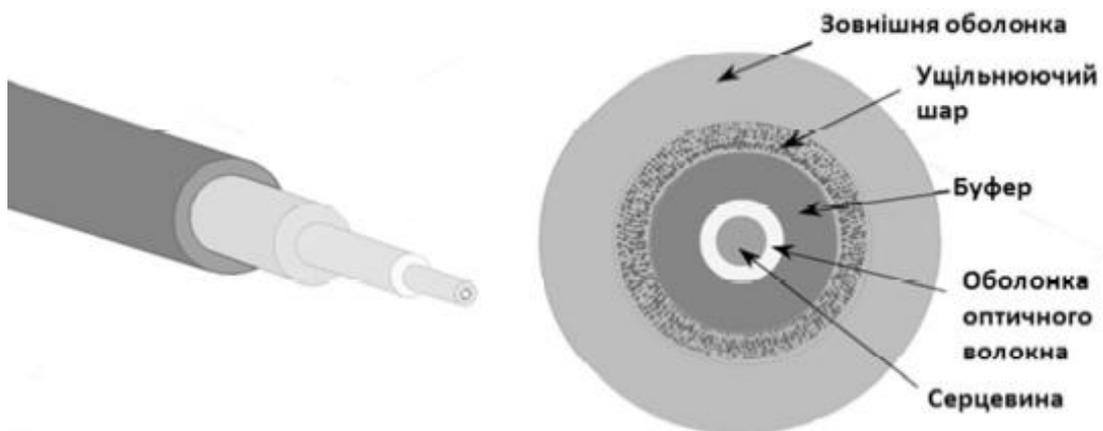


Рисунок 2.15 – Будова волоконно-оптичного кабелю

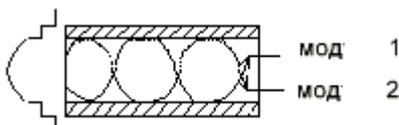
В залежності від діаметру серцевини та розподілу показника заломлення світла є наступні кабелі:

1) Багатомодове волокно із ступінчатою зміною показника заломлення.

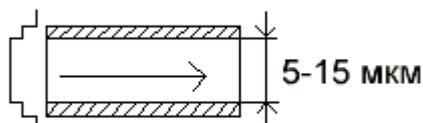
Кожна мода має свою частоту.



2) Багатомодове волокно з плавною зміною коефіцієнта заломлення.



3) Одномодове волокно.



Багатомодові кабелі застосовуються при швидкостях до 1 Гбіт/с на відстані до 2 км.

Одномодові – для передачі даних на надвисоких швидкостях. Технологія передачі даних AWDM забезпечує швидкість декілька Тбіт/с та відстані 10..100 км.

В якості джерела світла в багатомодових кабелях застосовуються світлодіоди, а в одномодових – лазерні діоди. Вартість безпосередньо волоконно-оптичних кабелів трохи більша за виту пару, проте значна частка коштів при реалізації мережі йде на монтажні роботи.

Для ВОК використовують різного типу роз'єднувачі (рис. 2.16). Роз'єднувачі для ВОК допускають невелику кількість вмикань/вимикань – до 1000.

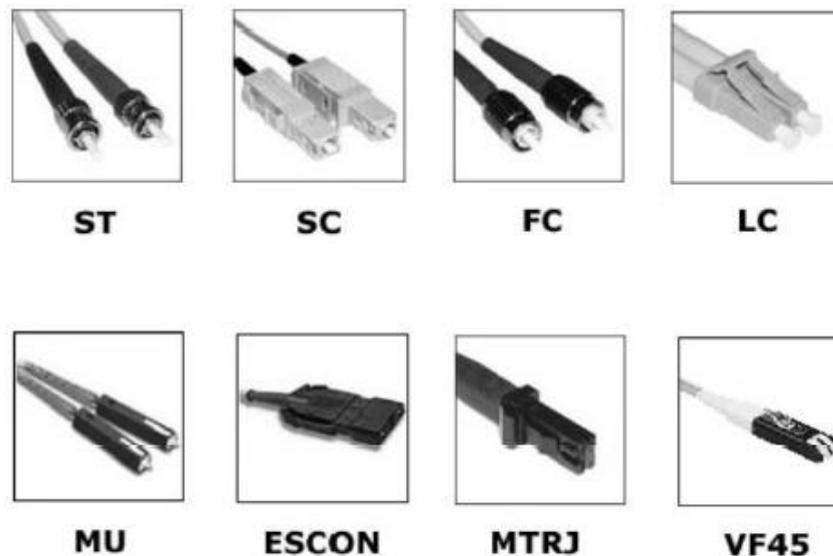


Рисунок 2.16 – Типи роз’ємів для волоконно-оптичного кабелю

Сьогодні волоконно-оптичні кабелі застосовуються для побудови магістралей, глобальних та локальних інформаційних мереж, міжповерхових з’єднань, в умовах сильних електромагнітних завад, у разі потреби забезпечити гальванічну розв’язку декількох мереж.

Структурована кабельна система будівель

Структурована кабельна система (СКС) (Structured Cabling System, SCS) – це набір комунікаційних елементів (кабелів, роз’ємів, кросових панелей, шаф, конекторів), а також методика їх використання, яка дозволяє створювати швидкорозширювані структури зав’язків в комп’ютерних мережах. Вартість кабельної системи визначається не вартістю кабелю, а вартістю робіт з його прокладання. В 90-х роках отримав розвиток новий вид організації промислового зв’язку, який ґрунтується на виготовленні, поставці, монтажі, сертифікації повністю комплектних, сумісних з усім мережевим обладнанням систем проводки і з’єднань для будівель і споруд – структуровані кабельні системи. Будівлі мають фіксовану архітектуру та найбільш забезпечені документацією, тому це і визначає структуру мережі. На основі стандартів СКС визначаються площі будівель (споруд), де можуть розташовуватись робочі

місця користувачів. На рис. 2,16 показано архітектуру СКС у комплексі з кількох споруд.

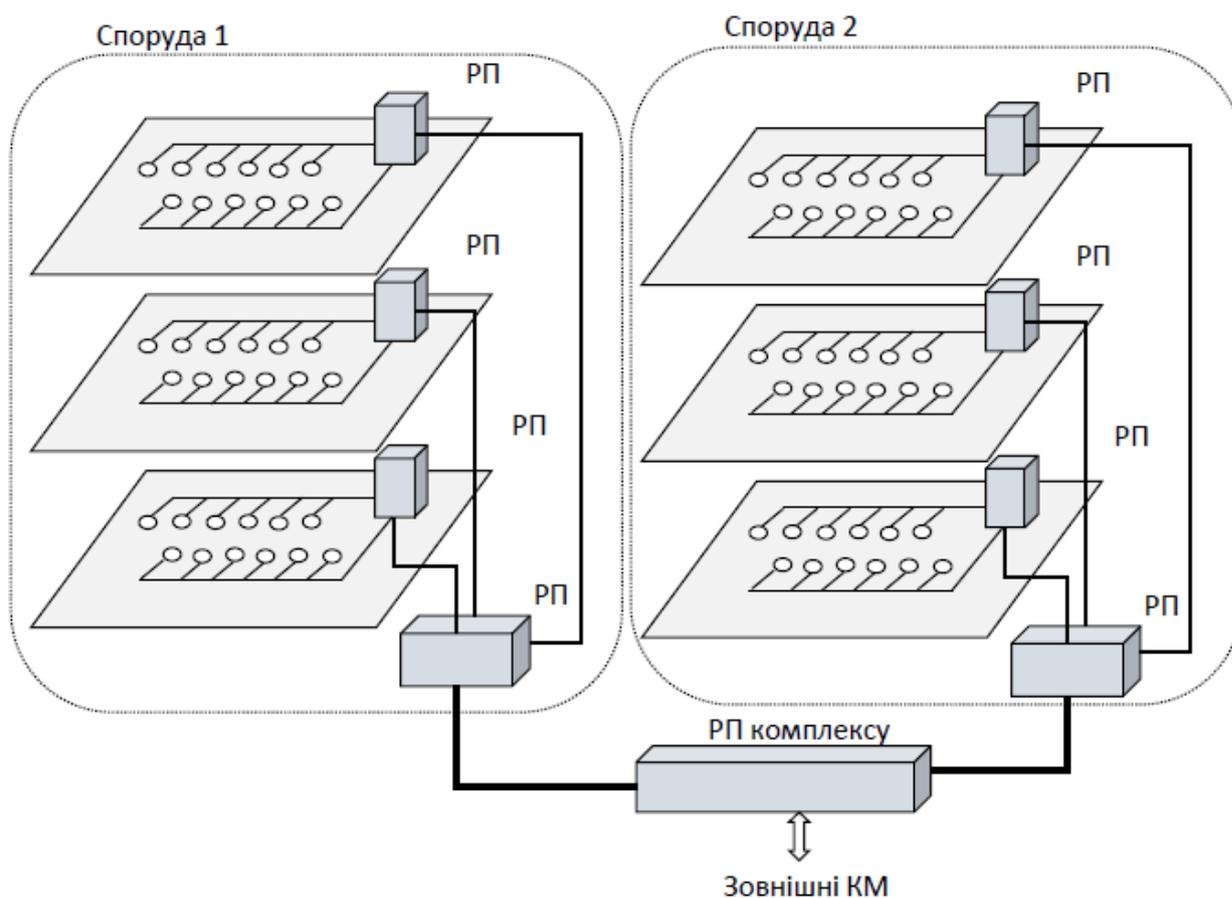


Рисунок 2,16 – Структурована кабельна система будівель

У відповідності зі стандартами проектування, СКС включає три підсистеми:

- 1) Магістральна кабельна система (КС) комплексу – з'єднує розподільчі пункти комплексів з розподільчими пунктами споруд;
- 2) Магістральна КС споруди (вертикальна КС) – з'єднує розподільчі пункти поверхів з розподільчими пунктами споруди;
- 3) Горизонтальна КС – з'єднує розподільчі пункти поверхів з точками підключення (розетками) користувачів.

Розподільчі пункти (РП) представляють собою кросові шафи, що забезпечують можливість створення мережі базової топології – шини, кільця, або зірки. У відповідності зі стандартами монтажу, рекомендується використовувати лише виту пару (кабель категорії 5 і вище) і волоконно-оптичний кабель, причому чим вищий рівень підсистеми, тим більша перевага надається оптоволокну.

2.9. Контрольні запитання до розділу 2

1. Загальна характеристика промислових мереж.
2. Мережева модель OSI.
3. Інтерфейси RS-485, RS-422, RS-232.
4. Промислові мережі 1-Wire, ASI.
5. Інтерфейс «струмова петля». HART-протокол.
6. Промислові мережі CAN, Profibus, Modbus.
7. Промисловий Ethernet.
8. Загальна характеристика безпроводникових мереж.
9. Безпроводникові мережі Bluetooth, ZigBee, Wi-Fi.
10. Обладнання мереж, в т.ч. перетворювачі інтерфейсів.
11. Кабелі для промислових мереж. Полоси пропускання ліній зв'язку.

3. КЕРУВАННЯ КООРДИНАТАМИ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ

3.1. Ідентифікація моделей об'єктів керування технологічних процесів

3.1.1 Структура та визначення електротехнологічного об'єкту

Процеси, що використовують електричну енергію, називаються електротехнологічними. Вони реалізуються за допомогою електротехнологічних систем (ЕТС). Електротехнологічні системи, як об'єкти дослідження, незалежно від призначення мають загальну структуру, що зображена на рис. 3.1.

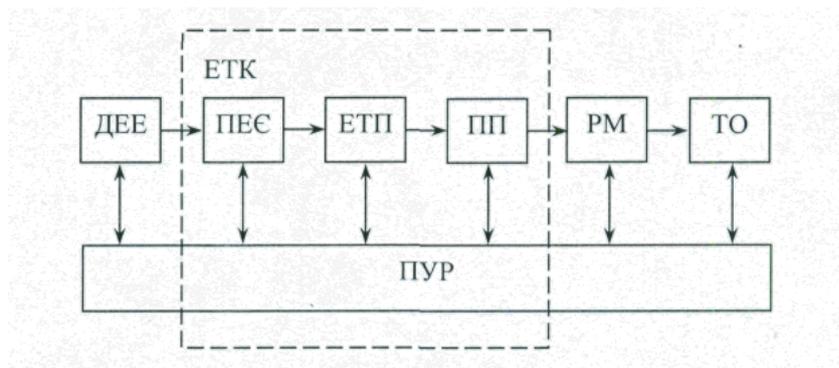


Рисунок 3.1 – Структура електротехнологічних систем

Електротехнологічні системи складаються з наступних елементів:

- 1) ДЕЕ – джерело електричної енергії, в якості якого може виступати перетворювач будь-якого виду енергії в електричну енергію (автономні джерела електроживлення, вторинні джерела електроживлення, акумуляторні і сонячні батареї, фотоелементи, термопари тощо). Окрім цього, ДЕЕ може бути промислова мережа, енергосистема, електромережі транспортних засобів і рухомих об'єктів;

- 2) ПЕЕ – перетворювач електричної енергії, пристрій, що перетворює електричну енергію з одними параметрами в електричну енергію з іншими параметрами (трансформатори, напівпровідникові перетворювачі, високочастотні електричні машини, керовані вентильні перетворювачі з автоматичним регулюванням зовнішньої характеристики, електронні гармати тощо) і призначений для узгодження параметрів джерела електричної енергії з параметрами електротехнологічного пристрою, і управління електротехнологічним пристроєм відповідно до сигналів пристрою управління.;
- 3) ЕТП – електротехнологічний пристрій, що перетворює електричну енергію в енергію іншого виду (механічну, теплову, хімічну, гідравлічну, акустичну, світлову), яка називається «технологічною». Він призначений для реалізації основного технологічного процесу і є основним елементом електротехнологічної системи в цілому;
- 4) ПП – перетворювальні пристрої виконують перетворення «технологічної» енергії з одними параметрами на «технологічну» енергію з іншими параметрами (редуктори, мультиплікатори, гідропідсилювачі, пневмопідсилювачі, пристрої каналізації теплової енергії, системи охолодження, футерування, теплоізоляція тощо) і призначені для узгодження параметрів електротехнологічного перетворювача енергії з параметрами робочої машини і управління робочою машиною відповідно до сигналів пристрою управління і регулювання.;
- 5) РМ – робочий механізм. До нього відноситься частина вищенаведеного електроустаткування, що безпосередньо забезпечує реалізацію необхідних робочих процесів у конкретному технологічному процесі (високотемпературні і низькотемпературні камери для термохімічної обробки виробів із чорних і кольорових металів, вузли, механізми і машини, приведені в рух електричними двигунами і так далі);

- 6) ТО – технологічний об'єкт. Це об'єкт, який піддається перетворенням у результаті виконання технологічних операцій. Саме для нього і створюється дана технологічна система.
- 7) ПУР – пристрій управління. Сюди відносяться інформаційні системи, що забезпечують управління і регулювання параметрів перетворення енергії в ході виконання технологічного процесу;
- 8) ЕТК – електротехнічний комплекс, підсистема електротехнологічної системи, що призначена для реалізації робочих процесів робочої машини в заданому технологічному процесі.

Сукупність різнорідних підсистем, що взаємодіють з ЕТК на основі закону загального взаємозв'язку, складають «довкілля».

Електротехнічний об'єкт (ЕТО) – може представляти собою як увесь ЕТК, так і окремі його частини і елементи. В залежності від виконуваних завдань електротехнічні об'єкти бувають:

- 1) Технологічні, де відбувається перетворення початкових матеріалів в готові вироби;
- 2) Транспортні, де відбувається переміщення сировини, напівфабрикатів або готових виробів;
- 3) Енергетичні, де відбуваються перетворення енергій;
- 4) Інформаційні, у яких здійснюються процеси, пов'язані з управлінням (керуванням) об'єктами.

Високоякісна продукція є продуктом високоякісних складових багатостадійного технологічного процесу (ТП), де кожна ділянка робить свій внесок у вихідний показник підсистеми верхнього рівня, у якість і собівартість продукції. Тому дуже важливо формалізувати та узгодити з головним показником показники якості процесу на кожній ділянці та побудувати адаптивні (до природної зміни параметрів) оптимальні системи автоматичного керування кожною ділянкою. Останнє можливо лише за наявності підсистем ідентифікації ділянок ТП, як об'єктів керування з використанням принципів побудови багаторівневих систем. Модель даного об'єкту це – зображення

істотних сторін реальної (чи конструйованої) системи в зручній формі, що відбиває необхідну для досягнення головної цілі інформацію про систему. Це визначення можна застосувати і до математичних моделей ЕТО, серед яких розрізняють наступні:

- 1) Непараметричні моделі ЕТО, які припускають використання деякого перетворення або функціонального простору, за допомогою якого можна представити вхідній і вихідній сигнали. Прикладами таких перетворень можуть слугувати розкладання в ряд Фур'є, розкладання по ортогональним сигналам прямокутної форми, розкладання в ряд по функціях Лаггера. Побудова моделей цього типу полягає у визначенні зв'язку між функціональними просторами вхідних і вихідних сигналів. У цих моделях використовується ідея «чорної скриньки» – яка не враховує фізичну природу досліджуваного ЕТО. Перевага цих моделей – вони не вимагають знання структури об'єкта. Недолік – функціональний простір є практично нескінченновимірним, що ускладнює його використання, зокрема, моделі непридатні для задач діагностики параметрів об'єкта.
- 2) Параметричні моделі ЕТО є системами з диференціальних або диференціально-алгебраїчних рівнянь кінцевої розмірності. Параметри цих рівнянь, що є параметрами моделей, виступають як координати деякого простору. Перевагою параметричних моделей є використання простору параметрів кінцевої розмірності, а також урахування апріорної інформації про головні складові структури і фізичні параметри ЕТО.

Саме теорія ідентифікації на основі попередньої (апріорної) і поточної (апостеріорної) інформації про причини (вхідні змінні ЕТО) і наслідки (вихідні змінні ЕТО) займається визначенням математичної моделі, що зв'язує ці змінні. Абсолютно точну ізоморфну модель ЕТО побудувати неможливо, оскільки за законами реального світу не існує ідеально ізольованого автономного об'єкта (все з усім взаємно пов'язане); не існує нічого стаціонарного (сталого), бо рух і матерія нероздільні, а виходячи з цього, як наслідок, не існує лінійних стаціонарних систем. Отже, будь-яка математична модель реального ЕТО буде

наближеною. Тому для отримання якісного наслідку в задачах керування, діагностики внутрішніх параметрів і змінних стану ЕТО, необхідно методами теорії ідентифікації побудувати таку математичну модель, за якої якнайкраще вирішується головне завдання (керування, прогноз, діагностика тощо). Теорія ідентифікації – це фактично формалізована математична теорія пізнання об’єктів реального світу, в тому числі і ЕТО.

Для кількісної і якісної оцінки процесів енергоперетворення, що протікають в ЕТО, використовуються параметричні математичні моделі, що припускають застосування систем диференціальних або диференціально-алгебраїчних рівнянь. Найбільш загальним підходом у математичному моделюванні цього типу, який дозволяє детальніше враховувати процеси перетворення енергії в ЕТО, є енергетичний формалізм Лагранжа.

Відповідно до сучасних уявлень енергія є мірою кількості руху матерії, незалежно від форми її прояву (механічною, електромагнітною, тепловою, хімічною, атомною). Енергія може акумулюватися на одних елементах ЕТО у вигляді потенціальної або кінетичної складової (індуктивні і ємнісні елементи електричних кіл, пружні елементи механічних пристроїв, теплопередавальні елементи теплових пристроїв тощо). На інших елементах ЕТО енергія розсіюється (резистивні елементи електричних пристроїв, дисипативні елементи механічних пристроїв, теплопередавальні елементи теплових пристроїв).

Для системи, що складається з M елементів, запасена енергія визначається виразом

$$W_K = \sum_{i=1}^N W_i^T + \sum_{j=1}^P W_j^U, \quad (3.1)$$

де $W_i^T, i=1, (1), N \leq M$ – величина кінетичної енергії, запасеної в N елементах ЕТО;

$W_j^U, j=1, (1), P \leq M$ – величина потенціальної енергії, запасеної в P елементах ЕТО.

Якщо в системі присутні G елементів, кожний з яких розсіює енергію W_k^R , тоді розсіяна енергія визначається сумою

$$W^R = \sum_{k=1}^G W_k^R. \quad (3.2)$$

Стан ЕТО визначається набором змінних, що характеризують процеси в підсистемах. Кожен з наборів є окремою підсистемою певної фізичної природи. Для представлення процесів перетворення енергії ЕТО в цілому, і для взаємозв'язку різнорідних підсистем між собою, згідно з формалізмом Лагранжа, вводяться:

– узагальнені координати $qI, I=1..S$ – усі S фізичних змінних, незалежно від їх фізичної природи, які однозначно визначають стан ЕТО в цілому (кут повороту, заряд, температура тощо);

– узагальнені сили F_i , – сили, що діють у напрямі узагальнених координат qI , і

які визначаються
$$f_i = \frac{dW_i}{dq_i};$$

– співвідношення, що називаються силовими функціями: $L = W^T - W^U$;
 $H = W^T + W^U$.

Силові функції відомі як функція Лагранжа L і функція Гамільтона H . Вони містять досить глибоку інформацію про процеси в підсистемах і їх взаємодію. За їх допомогою досягається універсальність моделювання різнорідних підсистем ЕТО з одночасним збігом опису електричних кіл рівняннями Кірхгофа, які моделюють в ЕТО основний елемент – електротехнологічний перетворювач (ЕТП).

Рівняння Лагранжа другого роду для систем, у яких присутні елементи, що розсіюють енергію, і на які діють зовнішні сили $F(t)$ має вигляд

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_m} \right) - \frac{\partial L}{\partial q_m} + \frac{\partial D}{\partial \dot{q}_m} = F(t), \quad m = 1 \div M, \quad (3.3)$$

де D – функція розсіювання Релея;

$\dot{q}_m = \frac{dq_m}{dt}$ – швидкість змін m -ї узагальненої координати.

Рівняння має ряд властивостей, які роблять їх вибір переважним порівняно з іншими підходами:

1) Для основної частини ЕТО (джерела електричної енергії, електричного перетворювача, електротехнологічного перетворювача) рівняння Лагранжа другого роду тотожні рівнянням, складеним за законами Кірхгофа, що дає можливість використання прийомів, методів і способів, добре розроблених у теорії електричних кіл;

2) Рівняння Лагранжа, так само як і рівняння для енергій, зберігають незмінною свою форму для різнорідних підсистем, унаслідок чого відпадає необхідність використання відповідно різнорідних систем змінних параметрів і понять;

3) Рівняння Лагранжа другого роду визначають через силову функцію Лагранжа, яка дозволяє визначити всі необхідні векторні величини (сили, моменти, швидкості тощо), будучи в той самий час скалярною функцією, чим досягається спрощення процедури моделювання.

Для ілюстрації застосування енергетичного формалізму Лагранжа розглянемо найпростіший приклад: електричне RLC-коло містить активний опір R , індуктивний L і ємнісний C елементи (рис. 3.2).

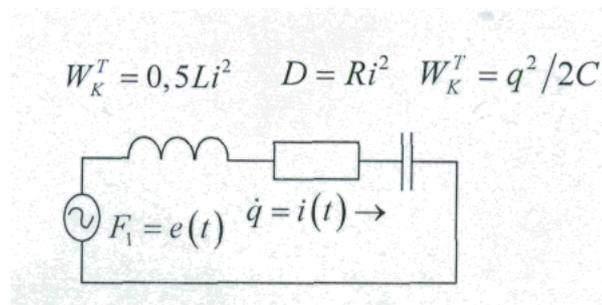


Рисунок 3.2 – Електричне коло, що містить ідеальні елементи, які запасують і розсіюють енергію

Дана система характеризується однією узагальненою координатою q_1 – електричним зарядом q . Швидкість зміни узагальненої координати q_1 є струм, що протікає у колі $\dot{q}_1 = i$. Кінетичною енергією в даному випадку є енергія

магнітного поля, що запасена індуктивним елементом $W_K^T = \frac{1}{2}Li^2$. В якості потенціальної енергії виступає енергія електричного поля, що запасється

конденсатором, $W_K^U = \frac{q^2}{2C}$. На активному опорі відбувається розсіювання частини енергії системи. За законом Джоуля–Ленца величина розсіяної енергії в одиницю часу $D = R \cdot i^2$. У системі присутня одна узагальнена сила $F_1(t)$, яка дорівнює значенню ЕРС ДЕЕ $e(t)$. Якщо значення активного опору, індуктивності і ємності, що називаються параметрами системи, вважати постійними, тоді силова функція Лагранжа має наступний вигляд

$$L_K = \frac{1}{2}Li^2 - \frac{q^2}{2C}. \tag{3.4}$$

Похідні, що входять в рівняння Лагранжа другого роду

$$\frac{\partial L_K}{\partial \dot{q}_1} = Li; \quad \frac{d}{dt} \left(\frac{\partial L_K}{\partial \dot{q}_1} \right) = L \frac{di}{dt}; \quad \frac{\partial L_K}{\partial q_1} = \frac{q}{C}; \quad \frac{\partial D}{\partial \dot{q}_1} = Ri. \quad (3.5)$$

Підставляючи отримані вирази (3.5) в рівняння Лагранжа (3.3), отримується

$$L \frac{di}{dt} + Ri + \frac{q}{C} = e. \quad (3.6)$$

Співвідношення (3.6), що складено за допомогою рівняння Лагранжа другого роду, повністю співпадає з рівнянням для даного кола, що отримується за II законом Кірхгофа. Воно може бути використане для визначення поточних значень узагальненої координати за відомими значеннями параметрів системи і закону зміни узагальненої сили (пряме завдання), або для визначення невідомих значень параметрів за відомими законами зміни узагальненої координати і узагальненої сили (завдання ідентифікації або зворотне завдання).

Однак, реальне RLC-коло (коло з реальними елементами) має більш складну модель, оскільки RLC-параметри на основі фундаментального закону загального взаємозв'язку залежать від змінних стану $I(t)$, $q(t)$ і вхідного впливу $F(t)$. Вочевидь, будь-яка математична модель лише наближено відтворює поведінку реального ЕТО. Отже постає завдання відшукати таку модель, за якої найкраще вирішується головна задача ЕТК. Така задача може бути успішно вирішена лише з використанням системного підходу.

3.1.2 Визначення електротехнічного об'єкту як системи

Важливим кроком розв'язання задачі дослідження ЕТО є її формалізація. Постановка задачі полягає у відображенні невизначеної ситуації, що пов'язана з ЕТО, у формалізовану задачу, визначену на множині кількісно порівнянних

елементів. Такими елементами є система, процеси, що в ній протікають, критерії (цілі) і стратегії їх оптимізації (досягнення цілей).

З умови фізичної реалізованості подано фундаментальні визначення стаціонарної, дискретної, безперервної, скінченновимірної, лінійної, гладкої системи та динамічної системи з точки зору їх зовнішньої поведінки, тобто, спостерігача. Для гладких систем (реальні системи майже всі гладкі) Р.Калманом доведено, що перехідна функція стану, тобто відображення

$$T \times T \times X \times \Omega \rightarrow X \quad (3.7)$$

водночас є розв'язком диференціального рівняння

$$\frac{dx}{dt} = f(t, x, u), \quad (1.8)$$

де $x \in X$, $u(t) = \omega(t) \in \Omega$; $t \in T$;

T – впорядкована множина моментів часу;

Ω , X – множина вхідних впливів і змінних стану.

Стан системи – це інформація про минуле, яка необхідна для повного опису майбутньої поведінки (тобто вихідних змінних) системи, якщо поведінка її вхідних змінних відома, починаючи з поточного часу t_0 .

Якщо множина X належить скінченновимірному простору, то модель (3.8) в загальному випадку являє собою скінченновимірну систему нелінійних нестационарних диференціальних рівнянь першого порядку.

Для гладкої системи (3.8) справедливе існування таких $X_1 \subset X$, $\omega_1 \subset \Omega$, $T_1 \subset T$, що (з точністю до наперед заданої малої похибки ε) системі (3.8) буде еквівалентна система лінійних диференціальних рівнянь із матрицями A і B постійних коефіцієнтів

$$\dot{x}_1(t) = Ax_1(t) + Bu_1(t); \quad y_1(t) = Cx_1(t), \quad (1.9)$$

де $\dot{x}_1(t)$ – вектор-функція похідних $\{\dot{x}_{1i}(t)\}$, $i = \overline{1, n}$; $x_1(t)$ – вектор-функція $\{x_{1i}(t)\}$, $i = \overline{1, n}$;

$u_1(t)$ – вектор-функція $\{u_{1j}(t)\}$, $j = \overline{1, m}$;

$y_{1i}(t)$ – вектор-функція вимірювань $\{y_i(t)\}$, $i = \overline{1, r}$;

A, B, C – матриці розміру $(n \times n), (n \times m), (r \times n)$ відповідно.

Структура і рангові властивості матриць A, B, C визначають умови керованості, спостережуваності та ідентифікаційності системи.

Критерій оптимальності або мета, поставлена системою більш високого рівня ієрархії, може мати «розмитий» характер. Однак у багатьох випадках формалізація критерію досягається шляхом завдання функціонала, який однозначно визначає ефективність поведінки системи. Під функціоналом I розуміють відображення

$$T \times T \times X \times \Omega \rightarrow R, \quad (3.10)$$

де R – множина дійсних чисел.

Мета системи – забезпечення екстремального значення I^* цього функціонала. Системам (3.8), (3.9) для фіксованих початкових $t_0 \in T$ і кінцевих $t_1 \in T$ моментів часу, станів $x_0 \in X, x_1 \in X$, поведінки системи $x(t)$, викликаній ненульовими початковими умовами та керувальним впливом $u(t)$, відповідає конкретне дійсне число I з множини R . Керувальний вплив $u^*(t)$ оптимальний,

якщо число I з урахуванням початкових умов, рівняння (3.8) чи (3.9) та інших обмежень на $x(t)$ і $u(t)$ приймає екстремальне значення. Для знаходження оптимальної стратегії управління $u^*(t)$ реальним об'єктом необхідна його модель (3.8), тобто структура і параметри відображення F . Визначення відображення F у рівнянні (3.8) за результатами вимірів $x(t)$, $u(t)$ є завданням ідентифікації. Оптимальність моделі \hat{f} відображення F оцінюється критерієм якості ідентифікації J , який також є функціоналом, що відображає множини реалізацій або конкретні реалізації $x(t)$ і $x_m(t)$ в дійсне число J , де $x_m(t)$ є розв'язком рівняння моделі для спільного з об'єктом вектору $u(t)$ вхідних впливів.

$$\frac{dx_1}{dt} = \hat{f}(t, x_1, u) \quad (3.11)$$

Оскільки в загальному випадку $\hat{f} \neq f$, то оптимальне для моделі (3.11) значення \hat{I}^* , отримане при керувальному впливі на об'єкти, буде дещо гірше істинно оптимального значення I^* .

Модель (3.11), побудовану з урахуванням не тільки функціонала J , але й I , назвемо цілеорієнтованою. Якщо на множині $\{F_I\}$ ($I=1, 2, \dots, n$) допустимих відображень відображення \hat{f} (3.11) для фіксованого управління $u(t)$ дає екстремальне значення \hat{I}^* , тобто $\hat{f} = \arg \underset{\hat{f}_i \in \{\hat{f}_i\}}{extr} I$, то таку модель назвемо цілеорієнтованою оптимальною. Близькість моделі до цілеорієнтованої оптимальної визначається близькістю вимірюваних змінних $\hat{x}(t), \hat{u}(t)$ до дійсних змінних $x(t)$ і $u(t)$ реального об'єкта. При вирішенні задачі ідентифікації в (3.11) і в $J(x, x_m)$ підставляються не $x(t)$ і $u(t)$, а їх оцінки $\hat{x}(t), \hat{u}(t)$, формування яких є задачею ідентифікації сигналів об'єкта. Оцінки $\hat{x}(t), \hat{u}(t)$ сигналів $x(t)$ і $u(t)$ в (3.8) отримуються за допомогою фільтрів

$$\frac{d\hat{z}}{dt} = f_{\hat{\delta}}(t, \hat{z}, z), \quad (3.12)$$

де $z(t)$ – вектор-функція вимірюваних первинними перетворювачами сигналів $x(t)$, $u(t)$;

$$z = \begin{bmatrix} x \\ u \end{bmatrix} + \begin{bmatrix} N_x \\ N_u \end{bmatrix},$$

N_x, N_u – похибки вимірювання x і відповідно.

Оптимальність фільтрів (3.12) оцінюється критерієм якості фільтрації J_f – функціоналом від сигналів \hat{z}, z або їх спектральних характеристик. Оскільки F_f не дає ідеального перетворення z в (x, u) , то від критерію J_f в задачі ідентифікації сигналів \hat{z}, z буде залежати значення критерію J в задачі ідентифікації відображення \hat{f} в (3.11), від якого (у свою чергу) залежить значення критерію I якості управління реальним об'єктом. Так само як і для моделі (3.11), для фільтра (3.12) введемо поняття цілеорієнтованості та оптимальності. Фільтр (3.12) цілеорієнтований, якщо при його побудові певним чином бралася до уваги задача ідентифікації \hat{f} в (3.11). Якщо на множині припустимих відображень $\{f_{\hat{\delta}^i}\}$ відображення F_f в (3.12) за фіксованого \hat{f} забезпечує екстремальне значення J^* , тобто

$$f_{\hat{\delta}} = \arg \underset{f_{\hat{\delta}^i} \in \{f_{\hat{\delta}^i}\}}{\text{extr}} J, \quad (3.13)$$

то фільтр (3.12) буде цілеорієнтованим оптимальним на задачу ідентифікації.

Оптимізація функціоналів I, J, J_f пов'язана з поняттям релаксаційного процесу. Множина $\{Q_k\}$, $k \geq 0$, кожен елемент якої належить деякій (зазвичай опуклій) області G нормованого простору, називається релаксаційним процесом (РП)

щодо функції $F(Q_k)$, якщо послідовність $F(Q_k)$ для упорядкованого за значенням k також впорядкована. Так, в задачі мінімізації послідовність $F(Q_k)$ така, що $F(Q_0) \geq F(Q_1) \geq F(Q_2) \geq F(Q_n)$. РП збігається за функціоналом,

якщо $\lim_{k \rightarrow \infty} (F(Q_k) - F(Q^*)) = 0$, і просто збігається, якщо $\lim_{k \rightarrow \infty} Q_k = Q^*$, де Q^* –

істинне значення. Надалі визначення РП поширимо на випадок, якщо елементи Q_k належать області G простору, у якому не задано поняття норми або відстані (знак \geq розуміється як символ відношення порядку). Так, в задачі ідентифікації при введенні підмножини структур Σ моделей і множин методів Opt оцінювання вектора β параметрів моделі (Σ, β) РП не є строго релаксаційним. Множину $\{Q_k\}$, $k \geq 0$, назовемо РП щодо $F(Q_k)$, якщо для будь-якого $k > 0$, існує таке значення $J > 0$, що $F_{k+J} \leq F_k$. Такий РП не є релаксаційним і збіжність його за функціоналом F або за Q_k залежить від розмірності і впорядкованості підмножини ненормованих елементів (у нашому випадку це $\{\Sigma, Opt\}$).

Системою ідентифікації в загальному вигляді будемо називати прямий добуток підмножин $\{\Sigma, \beta\}$, $\{Opt, \alpha\}$, $\{J\}$, $\{I\}$, $\{T\}$, що позначається як $\{\{\Sigma, \beta\}, \{Opt, \alpha\}, \{J\}, \{I\}, \{T\}\}$ і наділений структурою, яка дозволяє реалізувати релаксаційний процес відносно показника F , такий що $F_{k+J} \leq F_k$, $k=1, 2, \dots$; $J > 0$; $Q_k \in \{\{\Sigma, \beta\}, \{Opt, \alpha\}\}$, $F_k \in \{\{J\}, \{I\}\}$. Тут $\{\Sigma, \beta\}$ – підмножина моделей; $\{Opt, \alpha\}$ – підмножина методів оцінювання вектора β параметрів моделей $\{\Sigma, \beta\}$; α – вектор параметрів методу Opt ; $\{J\}$ – підмножина оптимізуємих за β методами $\{Opt, \alpha\}$ функціоналів від $\varepsilon(\beta, t)$; $\varepsilon(\beta, t)$ – різниця вимірюваних координат реального об'єкта Σ_∞ і моделі; $\{I\}$ – підмножина основних функціоналів, що визначають якість вирішення основної задачі для реальної системи Σ_∞ з використанням моделі $\{\Sigma, \beta\}$; $\{T\}$ – підмножина моментів t_k часу, на якому реальну систему Σ_∞ представлено набором даних $\{u(t_k), y(t_k)\}$ у

вигляді системи з точки зору її зовнішнього поводження [26]. У табл. 3.1 визначено різні за $\{Q_k\}$ і $\{F_k\}$ системи ідентифікації.

Таблиця 3.1 – Систематизація систем ідентифікації за ступенем їх досконалості

№, з/п	Тип системи	$\{Q_k\}$				$\{F_k\}$	
		β	Σ	α	Opt	J	I
1	сі	$\{\beta_k\}$	Σ	α	Opt	J	–
2	СІ	$\{\beta_k\}$	$\{\Sigma\}$	α	Opt	J	–
3	асі	$\{\beta_k\}$	Σ	$\{\alpha_k\}$	Opt	J	–
4	АСІ	$\{\beta_k\}$	$\{\Sigma\}$	$\{\alpha_k\}$	Opt	J	–
5	басі	$\{\beta_k\}$	Σ	$\{\alpha_k\}$	$\{Opt\}$	$\{J_k\}$	I
6	БАСі	$\{\beta_k\}$	$\{\Sigma\}$	$\{\alpha_k\}$	$\{Opt\}$	$\{J_k\}$	I
7	СС	$\{\beta_k\}$	$\{\Sigma\}$	$\{\alpha_k\}$	$\{Opt\}$	$\{J_k\}$	$\{I_q\}$

У практичному застосуванні система ідентифікації тим ефективніше, чим менше потрібно для її роботи апріорної інформації. Забезпечення необхідної якості ідентифікації досягається адаптацією, що полягає в цілеспрямованій зміні одного, декількох або всіх елементів підмножини $\{Q_k\}$ з метою досягнення екстремуму головного показника F. За інших рівних умов, чим краще алгоритм адаптації, тим ефективніше система. Якщо не враховувати фактор складності, то при адаптації $\{Q_k\}$ результат тим ефективніший, чим ширша підмножина $\{Q_k\}$. У разі врахування в показнику F складності системи існує оптимальна за F потужність підмножини $\{Q_k\}$. Спільним для всіх видів систем ідентифікації є наявність РП $\{\beta_k\}$ щодо показника J, а різним – склад підмножин $\{Q_k\}$ і $\{F_k\}$.

Відповідно до табл. 3.1 можуть бути наступні системи ідентифікації:

Системи ідентифікації у вузькому сенсі (сі) реалізують РП $\{\beta_k\}$ щодо J за постійних Σ , Opt, α і порожній множині I, тобто $Q_k = \{\beta_k\}$, $F=J$. Тут задача ідентифікації є еквівалентною задачі оптимізації функції $J(\beta_k)$.

Системи ідентифікації в широкому сенсі (СІ) реалізують РП $\{\Sigma_k, \beta_k\}$ щодо J за постійних Opt , α , і порожній множині I , тобто $Q_k = \{\Sigma_k, \beta_k\}$, $F=J$. Наприклад, розглянуто динамічні ортогональні, ноніусні або регресійні моделі з вектором β_k змінної розмірності, для яких оптимальна по J пара $\{\Sigma^*, \beta^*\}$

$$\{\Sigma^*, \beta^*\} = \arg \min_{\{\Sigma_k, \beta_k\}} J\{\Sigma_k, \beta_k\}.$$

визначається за умови

Адаптивні системи ідентифікації у вузькому сенсі (асі) реалізують РП $\{\beta_k, \alpha_k\}$ щодо J при постійному Opt і порожній множині I , тобто $Q_k = \{\beta_k, \alpha_k\}$, $F=J$. Так, системи [34, 35, 36], що використовують прискорений градієнтний спуск $\{\beta_k\}$ по J , як α_k мають параметр регуляризації, що оптимізується за допоміжної умови мінімуму різниці середньоквадратичних значень похибок для двох одностипних моделей, які налаштовуються однаковими алгоритмами Opt , але відрізняються параметром α : для першого $\alpha = \alpha_k$, для другого $\alpha = \alpha_k + \Delta$, де $\Delta > 0, \alpha_k > 0$.

Адаптивні системи ідентифікації в широкому сенсі (АСІ) реалізують РП $\{\Sigma_k, \beta_k, \alpha_k\}$ щодо показника J при постійному Opt і порожній множині I , тобто $Q_k = \{\Sigma_k, \beta_k, \alpha_k\}$, $F=J$. Наприклад, системи вибору «найкращої регресії», при оцінюванні вектора β , що розширюється за розмірністю, із застосуванням регуляризувального параметра α , який підбирається за алгоритмом гребеневої регресії.

Багаторазово адаптивні системи ідентифікації у вузькому сенсі (басі) реалізують РП щодо основного показника I $\{\beta_k, Opt_k, \alpha_k\}$ у вигляді композиції двох РП: а) $\{\beta_{kn}\}, n = 1, 2, \dots$, щодо J_k ;

б) $\{Opt_k, \alpha_k, J_k\}$, $k = 1, 2, \dots$, щодо I ; тобто $Q_k = \{\beta_k, Opt_k, \alpha_k\}$, $F = \{\{I_k\}, \{I\}\}$.

Поняття багаторазовості пов'язано з повторенням РП $\{\beta_{kn}\}$ для кожного елемента $\{Opt_k, \alpha_k, I_k\}$ РП більш високого рангу. Якщо кожному показнику J_k відповідає конкретний метод Opt або для евристичних алгоритмів показник J_k взагалі відсутній, то відносно I оптимізується пара «метод – його параметри» $\{Opt_k, \alpha_k\}$. Прикладом басі може слугувати система, у якій для заданої за одним із п'яти показників I (або їх зваженої суми), що включають критерії регулярності та незміщеності, точності прогнозування по моделі тощо, із семи методів $\{Opt_k\}$ і їх параметрів $\{\alpha_k\}$ – визначався оптимальний елемент $\{\beta^*, Opt^*, \alpha^*\}$ згідно з пп. 5а і 5б.

Багаторазові адаптивні системи ідентифікації в широкому сенсі (БАСІ) реалізують РП відносно головного показника I $\{\Sigma_k, \beta_k, Opt_k, \alpha_k\}$ у вигляді композиції двох РП [16]: а) $\{\beta_{kn}\}$, $n=1,2,\dots$, щодо J_k ; б) $\{\Sigma_k, Opt_k, \alpha_k, I_k\}$, $k=1,2,\dots$, щодо I , тобто $Q_k = \{\Sigma_k, \beta_k, Opt_k, \alpha_k\}$, $F = \{\{I_k\}, I\}$, де I – показник більш високого рівня. Прикладом БАСІ (для одноелементної множини $\{Opt\}$) є системи, що побудовані на основі методу групового обліку аргументів, у яких структуру Σ та параметри β підбирають з умови екстремуму I . За інших рівних умов обґрунтоване розширення множини $\{Opt_k\}$ дозволить отримати моделі, більш ефективні за критерієм I . Якісною відмінністю систем за пп. 5, 6 є їх цілеорієнтація на показник I , який регуляризує і оптимізує задачу вибору $\{\Sigma^*, \beta^*, Opt^*, \alpha^*\}$. У багаторазово адаптивних системах знімається традиційне для систем за пп. 1–4 питання: як оцінити якість моделі, отриманої в результаті ідентифікації? Якість моделі оцінюють за показником I . Це найбільш об'єктивна оцінка, оскільки найкраща та модель, застосування якої дає найкраще рішення основної задачі, якість якої визначено показником I .

Системи керування, що самоорганізуються (СС), реалізують РП щодо деякого показника $\Lambda(I)$ більш високого рангу $\{\Sigma_k, \beta_k, Opt_k, \alpha_k, J_k, I_k\}$ у вигляді композиції трьох РП:

а) $\{\beta_{qkn}\}$, $n=1,2,\dots$ (q і k постійні), щодо J_{qk} зі стаціонарною точкою $\beta_{qk} = \arg \text{extr}_{\beta_{qkn}} J_{qk}(\beta_{qkn}, \Sigma_q, Opt_{qk}, \alpha_{qk}) = \arg J_{qk}^*$;

б) $\{J_{qk}^*(\beta_{qk}, \Sigma_{qk}, Opt_{qk}, \alpha_{qk})\}$, $k=1,2,\dots$ (q постійне), щодо I_q зі стаціонарною точкою

$$J_q^* = J_q(\beta_q, \Sigma_q, Opt_q, \alpha_q) = \arg \text{extr}_{J_q} I_q(J_q^*) = \arg I_q^*; \quad \text{в) } \{J_q^*(\beta_q, \Sigma_q, Opt_q, \alpha_q)\}, q=1, 2,$$

..., щодо показника Λ більш високого рівня зі стаціонарною точкою

$$I^* = \arg \text{extr} \Lambda(I_q^*) = \arg \Lambda^*, \quad \text{тобто } Q_k = \{\Sigma_k, \beta_k, Opt_k, \alpha_k\}, F = \{J, I_k\}.$$

Наприклад, задача вдосконалення процесу автоматизованого керування кабельним виробництвом. Λ – показник ефективності підприємства $\{I_q\} = \{I_1, I_2\}$; I_1 – показник якості стабілізації технологічних змінних x агрегату в області робочих режимів x_0 , що задаються експертами (технологами); I_2 – показник якості роботи агрегату; $\{J_{qk}\}$ – множина показників якості ідентифікації локальними моделями $\{\Sigma_{q1}, \beta_{q1}\}$, які відображають перетворення керуючого впливу u у вихідну змінну x ($k=1$), і моделями якості $\{\Sigma_{q2}, \beta_{q2}\}$, що відображають перетворення I_2 ($k=2$). На першому етапі автоматизації ($I_q=I_1$) за I_{1k} визначають локальні моделі $\{\Sigma_{1k}, \beta_{1k}\}$ за допомогою найпростіших методів

ідентифікації $\{Opt_{1k}, \alpha_{1k}\}$. За моделями $\{\Sigma_{1k}, \beta_{1k}\}$ підбирають локальні регулятори, стабілізуючі x в області x_0 за критерієм мінімуму I_1 . Згідно РП пп. 7а, 7б визначають оптимальний по I_1 елемент. На другому етапі, після автоматизації процесу збирання і обробки інформації, переходять до побудови більш повної моделі $\{\Sigma_{q2}, \beta_{q2}\}$, що пов'язує I_2 зі змінними x , u . По моделям $\{\Sigma_{q2}, \beta_{q2}\}$, які містять в собі моделі $\{\Sigma_{q1}, \beta_{q1}\}$, уточнюють параметри локальних регуляторів та оптимальне значення уставок x_0^* робочих режимів x_0 . Далі оптимальний вже за I_2 елемент $\{\Sigma_2^*, \beta_2^*, Opt_2^*, \alpha_2^*\}$ визначають уже за результатом РП пп. 7а, 7б. Таким чином, перехід від Σ_1 до Σ_2 привів до заміни (згідно РП п. 7в) основного показника I_1^* на I_2^* , як аргументу показника Λ (показника якості вихідного продукту).

3.1.3 Багаторівнева декомпозиція систем з ідентифікатором електротехнічного об'єкта

Декомпозиція дозволяє представити систему великої розмірності зі складною мережею прямих і зворотних зв'язків системою більш простих підсистем, які краще піддаються формалізації. Уявімо завдання проектування системи і саму систему оптимального адаптивного управління ЕТО у вигляді трьох підзадач і відповідних їм підсистем (рис. 3.3):

- 1) Оптимального управління (рівень 3);
- 2) Ідентифікації відображення вхід-вихід ЕТО (рівень 2);
- 3) Ідентифікації сигналів (рівень 1).

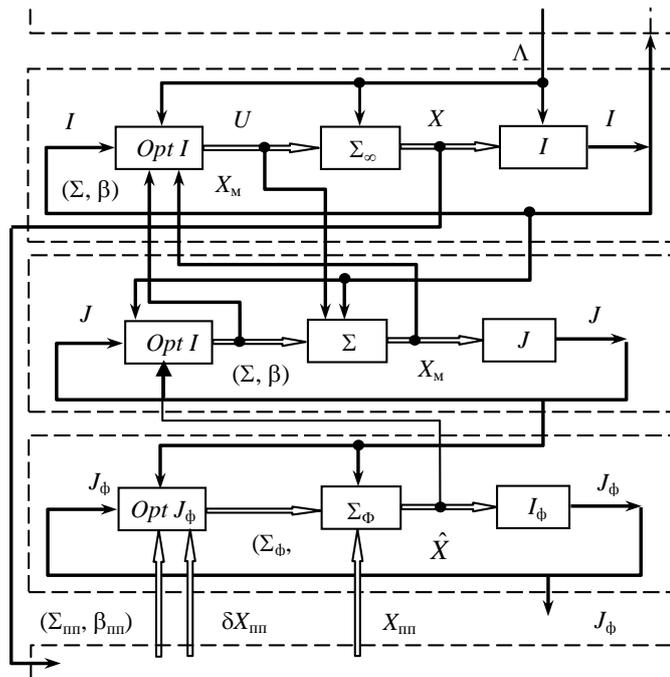


Рисунок 3.3 – Три рівні оптимізації в багаторівневій системі

Елементи кожної підсистеми утворюють замкнену за місцевим показником якості систему оптимізації, працюючу за алгоритмом систем зі зворотним зв'язком:

- 1) видача керуючого впливу на ЕТО;
- 2) вимір або розрахунок реакції ЕТО;
- 3) оцінка оптимальності підсистеми за критерієм її рівня.

Між собою підсистеми об'єднані прямими (знизу вгору) і зворотними (згори вниз) зв'язками. Крайні підсистеми об'єднані такими зв'язками з не розглянутими тут системами більш низького та високого рівнів. Наведемо позначення і фізичний зміст елементів, внутрішніх і зовнішніх прямих і зворотних зв'язків.

Підсистема ідентифікації сигналів ЕТО (рівень 1) має такі елементи та зв'язки:

J_ϕ – формувач показника якості фільтрації вектора x сигналів об'єкта;

Σ_ϕ – фільтр, що перетворює сигнал виходу первинного перетворювача $x_{\pi\pi}$ в його оцінку \hat{x} ;

$\text{Opt } J_f$ – оптимізатор по J_f структури Σ_f і вектора β_f параметрів фільтра;
 $\Sigma_{пп}$, $\beta_{пп}$ – інформація про структуру і параметри первинних перетворювачів (датчиків) фізичних змінних об'єкта;
 $\Sigma_{пп}$ – передаточна функція датчика, статична (тарувальна) характеристика тощо;
 $\delta_{хпп}$ – інформація про похибки датчиків: систематичні, випадкові, часові, частотні та ймовірнісні характеристики;
 J , J_f – інформація про критерії оптимальності систем ідентифікації об'єкта і його сигналів: J надходить по каналу зворотного зв'язку в дану підсистему, J_f – в підсистему первинних перетворювачів.

Підсистема ідентифікації ЕТО (рівень 2) містить:

J – формувач показника якості ідентифікації об'єкта;
 Σ , β – модель об'єкта зі структурою Σ і вектором β параметрів;
 $\text{Opt } J$ – оптимізатор за показником J структури Σ і вектора β параметрів моделі;
 Σ_{Φ} , β_{Φ} , \hat{X} – інформація про фільтр і оцінку \hat{X} сигналу x , отриману з сигналу $x_{пп}$ первинного перетворювача;
 Σ , β , x_m – інформацію про модель (Σ, β) об'єкта управління і оцінку x_m сигналу x , отриману за сигналом \hat{X} фільтра;
 I , J – коригувальні зворотні зв'язки з системи управління в систему ідентифікації і з системи ідентифікації об'єкта в систему ідентифікації сигналів.

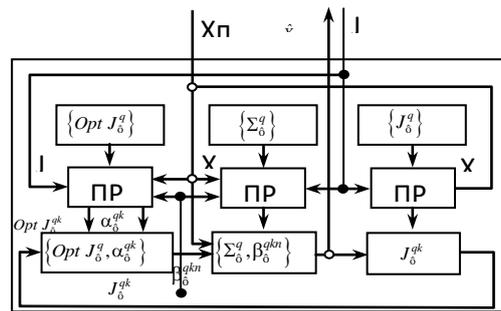
Підсистема оптимального управління ЕТО (рівень 3) містить:

I – формувач показника якості управління;
 Σ_{∞} – реальний об'єкт (його вхідні і вихідні змінні u і x);
 $\text{Opt } I$ – оптимізатор показника I по керуючому впливу u ;
 x – вихід ЕТО – вхід підсистеми первинних перетворювачів;
 u – вхід ЕТО і моделі;
 Λ , I – коригувальні зворотні зв'язки із системи більш високого рівня в систему управління і з системи управління в систему ідентифікації відповідно.

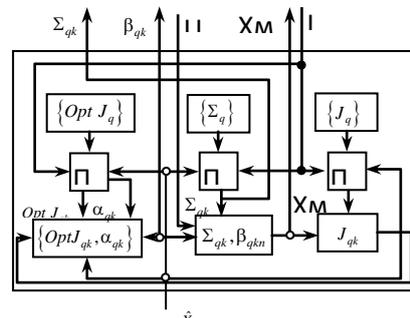
Кожен елемент (оптимізатор, модель, формувач критерію) має три складові:

- 1) безпосередньо елемент, зав'язаний на розв'язанні своєї задачі;
- 2) множину таких елементів, упорядкованих за їх властивостями;
- 3) проектор (ПР), що вибирає з множини елементів оптимальний за показником якості системи більш високого рангу.

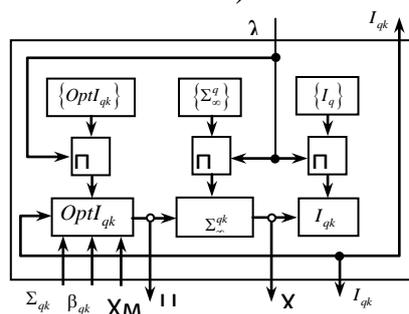
На рис. 3.4 подано структури трьох рівнів підсистем і відповідно схеми задач проектування.



1)



2)



3)

Рисунок 3.4 – Цілеорієнтована система: 1) – підсистема ідентифікації сигналів; 2) – підсистема ідентифікації об'єкта; 3) – підсистема оптимального адаптивного керування об'єктом

Індекси q , q_k , q_{kn} відповідають позначенням кроків композиційного релаксаційного процесу оптимізації системи. Наприклад, q – номер ітерації зміни виду показника I ; q_k – відповідно зміни J ; q_{kn} – номер кроку в оптимізації β в системі ідентифікації об'єкта.

Цілеорієнтована система складається з 27 елементів, серед яких елементи $\{Opt J_{\delta}^q\}$, $\{Opt J_q\}$, $\{Opt I_q\}$ – множини методів оптимізації, критеріїв $\{J_{\delta}^q\}$, $\{J_q\}$, $\{I_q\}$ і моделей $\{\Sigma_{\delta}^q\}$, $\{\Sigma_q\}$, $\{\Sigma_{\infty}^q\}$, складаються з множин наявних упорядкованих елементів.

Для конкретного критерію теоретично існує єдиний оптимальний набір цих елементів і їх параметрів. Завдання полягає у виборі «відповідного» (близького за I_q до оптимального) набору за обмежених витрат на його пошук в системі. Для повної апріорної невизначеності про всі елементи системи та їх властивості проблема знаходження «підходящого» набору може здійснюватися методом перебирання варіантів. Повне перебирання гарантує відшукання оптимального набору, якщо в критерій оптимальності не входять витрати на пошук оптимуму. Якщо ці витрати істотні, то вони впливають на оптимальність рішення. Тому, щоб дана декомпозиція не призвела до ускладнень, необхідний ретельний аналіз підсистем та їх елементів, який дозволяє з допомогою проекторів Pr систем кожного рівня (рис. 3.4) істотно звужити вихідні множини елементів до обмежених підмножин претендентів на оптимальні для конкретної ситуації.

Принцип декомпозиції ефективний для складних систем та їх елементів. Так, неможливість одним функціоналом сформулювати всі вимоги до створюваної системи призвела на практиці до розв'язання задачі декомпозиції цього неформалізованого критерію на множини критеріїв, які піддаються чіткій формалізації, і розв'язанням багатокритеріальної задачі оптимізації. Таке розв'язання не дає єдиного результату. Воно лише дозволяє виділити обмежену область (підмножина Парето) у просторі критеріїв-функціоналів від

оптимізованих змінних. Остаточний варіант розв'язання, що обирається проектувальником, належить цій підмножині.

Найпоширенішими принципами декомпозиції для методів оптимізації ЕТО є наступні:

- 1) Метод покоординатної оптимізації I (метод Гаусса–Зейделя), а також метод групової релаксації реалізують декомпозицію простору оптимізованих змінних і почергової покоординатної або групової оптимізації;
- 2) Для лінійних оптимальних за квадратичним функціоналом стохастичних систем управління ЕТО декомпозиція загальної задачі на підзадачі оптимального оцінювання станів та пошуку оптимальної стратегії управління дозволяє суттєво спростити складну задачу дуального управління;
- 3) Для задачі спільного оцінювання параметрів і станів стохастичних ЕТО (розширений фільтр Калмана, метод квазілінеаризації і інваріантного занурення), поділ на незалежні підзадачі оцінювання сигналів, їх коваріаційних матриць, а далі параметрів, дозволяє отримати рішення, близьке до оптимального, досить простими алгоритмами.

Задача декомпозиції моделі ЕТО, що описується системою нелінійних нестационарних рівнянь виду (3.1)

$$\dot{x}_g(t) = f_g(t, x_1, \dots, x_n, u_1, \dots, u_m), \quad g = \overline{1, n}, \quad (3.14)$$

полягає в отриманні множини лінійних систем виду (3.2), яка описує з необхідною точністю поведінку системи (3.14) в обмежених областях визначення змінних t , x , u .

Локальні моделі виду (3.2) дають змогу у межах лінійно-пропорційних систем легко розв'язувати задачі аналізу, синтезу і реалізації оптимальних стратегій керування та ідентифікації. За локальними моделями ЕТО можна

розв'язати задачу ідентифікації структури невідомої нелінійної залежності F моделі (3.14). Уведемо в рівняння (3.14) позначення вектора v з компонентами v_i

$$v_i = \begin{cases} x_i, & i = \overline{1, n}; \\ u_i, & i = \overline{n+1, n+m}; \\ t, & i = n+m+1 = s \end{cases} . \quad (1,15)$$

У силу безперервності і L-кратної диференційованості залежності (3.14) вона може бути подана кратним рядом Тейлора

$$y(v(t)) = y(v_0) + \sum_{i=1}^s \frac{\partial y}{\partial v_i} \Big|_{v_0} \Delta v_i(t) + \frac{1}{2} \sum_{i=1}^s \sum_{j=1}^s \frac{\partial^2 y}{\partial v_i \partial v_j} \Big|_{v_0} \Delta v_i \Delta v_j + \\ + \frac{1}{6} \sum_{i=1}^s \sum_{j=1}^s \sum_{k=1}^s \frac{\partial^3 y}{\partial v_i \partial v_j \partial v_k} \Big|_{v_0} \Delta v_i \Delta v_j \Delta v_k + \dots + R_l(v), \quad (3,16)$$

де $\Delta v_i(t) = v_i(t) - v_{i0}$; $v \in G$; $v_0 \in G$;

$\text{Sup } R_l(v) \leq \delta$; δ – допустима похибка апроксимації залежності (3.14) рядом (3.16).

З метою декомпозиції розіб'ємо область G завдання змінних v_i на підобласті G_r так, що

$$\text{а) } \bigcup_{r=0}^m G_r = G, v_{r0} \in G_r, G_r \cap G_{r'} = \emptyset, r \neq r'; \\ \text{б) } (\forall G_r) \left(\exists \beta_r = \arg \min_{\beta_r \in R} \left(\sup_{v_{r0} \in G_{r,v_{r0}}} \left| \frac{dy}{dx} - \beta_r \right| < h \right) \right), \quad (3,17)$$

де β_r – вектор параметрів лінійної моделі.

$$\Delta y_1(t) = \beta_r^T \Delta v(t), \quad (3.18)$$

де I -та компонента β_{ri} , якого є оцінка з точністю до h коефіцієнта $\left. \frac{\delta y}{\delta v} \right|_{v_0}$, розкладання (3.16) навколо центру v_{r0} .

Для області G_0 , яка містить глобальний центр v_0 розкладання (3.16) за моделлю (3.18) і даними $y_1(t)$, $v(t)$, знайдемо вектор β_r , який дорівнює

(з точністю до h) вектору $\left. \frac{\delta y}{\delta v} \right|_{v_0}$, тобто отримаємо оцінки перших похідних у

$\left. \frac{\delta y}{\delta v} \right|_{v_0}$ (3.16). Аналогічно обчислимо вектори β_r часткових моделей для решти

областей $G_r \in G$. Тепер за значеннями β_r та v_{r0} можна оцінити похідні більш високих порядків. Так, якщо об'єднати сусідні з G_0 підобласті в розширену

область $G_{01} = \bigcup_{r=1}^{m_1} G_r, m_1 \geq s$, (для простоти позначень будемо вважати, що області перенумеровані з віддаленням від «центральної» G_0), таку, що

$$\sup_{v \in G_{0j}} R_3(v) \leq \delta, \quad (3.19)$$

то розклад (3.16), з точністю до δ , буде містити члени не вище другого порядку.

Продиференціюємо (3.16) в області G_{01} по v_j , тоді з точністю до $R_3(v) / \partial v_j$,

$$\frac{\partial y(v(t))}{\partial v_j} = \frac{\partial y}{\partial y_j} \Big|_{v_0} + \sum_{i=1}^s \frac{\partial^2 y}{\partial v_i \partial v_j} \Big|_{v_0} \Delta v_i, \quad (3.20)$$

де $\frac{\delta y}{\delta v} \Big|_{v_0}$ визначено з (3.18) для підобласті G_0 .

Якщо різниця $\Delta v_i = v_{r0i} - v_{0i}$, то їй наближено відповідає похідна

$$\frac{\partial y(v(t))}{\partial v_j} \Big|_{v_{r0}} \approx \beta_{rj}. \quad (3.21)$$

Отже, вираз (3.20) можна подати у вигляді

$$\beta_{rj} = \beta_{0j} + \sum_{i=1}^s \frac{\partial^2 y}{\partial v_i \partial v_j} \Big|_{v_0} (v_{r0i} - v_{0i}). \quad (3.22)$$

Звідси для $r \geq s$ і, за умови лінійної незалежності хоча б s з r векторів

$(v_{r0} - v_0)$ однозначно визначаються s других похідних $\frac{\partial^2 y}{\partial v_i \partial v_j} \Big|_{v_0}$, $I=1, 2, \dots, s$.

Для знаходження всіх коефіцієнтів $\frac{\partial^3 y}{\partial v_i \partial v_j \partial v_k} \Big|_{v_0}$ ряду (3.16) необхідно мати не менше s розширених навколо G_{r1}^* підобластей G_{r1}^* , подібних G_{01} (число таких центральних підобластей повинно бути не менше s). Для кожної G_{r1}^* із рівняння (3.22), у якому замість β_{0j} і v_{0i} взято значення, що отримані для

центральної підобласті G_{r^*}) знаходяться другі похідні $\left. \frac{\partial^2 y}{\partial v_i \partial v_j} \right|_{v_{r^*0}}$. Після подвійного диференціювання (3.7), отримується

$$\left. \frac{\partial^2 y(t)}{\partial v_i \partial v_j} \right|_{v_{r^*0}} = \left. \frac{\partial^2 y}{\partial v_i \partial v_j} \right|_{v_0} + \sum_{k=1}^s \left. \frac{\partial^3 y}{\partial v_i \partial v_j \partial v_k} \right|_{v_0} (v_{r^*0k} - v_{0k}). \quad (3.23)$$

Звідси для $r^* \geq s$ та за умови лінійної незалежності хоча б s векторів $(v_{r^*k} - v_k)$ однозначно визначають треті похідні. Аналогічно отримуються всі похідні ряду (3.16) і тим самим визначається його структура.

За недостатньої кількості підобластей G_r можливо відновлення частини структури з тих v_j , які змінюються від області до області.

Під час проведення активного експерименту мінімізація числа областей G_r і оптимізація точності оцінок похідних в (3.16) досягається застосуванням композиційного планування експерименту, яке забезпечує ортогональність векторів $(v_{r0} - v_0), (v_{r^*0} - v_0)$.

Перевагою такого підходу є обмежена на кожному кроці розширення розмірність вектора невідомих параметрів, можливість аналізу на кожному кроці значущості знайдених коефіцієнтів ряду (3.16) та відкидання незначущих коефіцієнтів.

3.2. Класичні закони керування та їх модифікації

Всі SCADA системи мають вбудовані ПІД-регулятори, причому їх практична реалізація в кожній системі найчастіше здійснюється з певними особливостями.

Структура класичного замкненого контуру керування з ПД-регулятором показана на рис. 3.5.

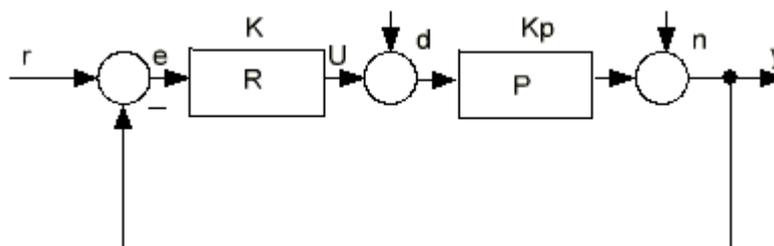


Рисунок 3.5 – Замкнений контур керування: r – уставка (завдання); d – збурення; n – шуми вимірювання; y – регульована (вихідна) змінна

Рівняння для регульованої (вихідної) змінної

$$y(p) = \frac{P(p) \cdot R(p)}{1 + P(p) \cdot R(p)} \cdot r(p) + \frac{1}{1 + P(p) \cdot R(p)} \cdot n(p) + \frac{P(p)}{1 + P(p) \cdot R(p)} \cdot d(p)$$

Передаточна функція ПД-регулятора

$$R(p) = K + \frac{1}{T_i(p)} + T_d(p) = K \left(1 + \frac{1}{KT_i(p)} + \frac{T_d}{K} p \right)$$

$$U(t) = K_e(t) + \frac{1}{T_i} \int e(t) dt + T_d \cdot \frac{de(t)}{dt}$$

П-регулятор

$$T_d = 0$$

$$T_i = \infty$$

$$y = \frac{K_p K}{1 + K_p K} \cdot r + \frac{1}{1 + K_p K} \cdot n + \frac{K_p}{1 + K_p K} \cdot d$$

де K – коефіцієнт посилення регулятор;
 K_p – коефіцієнт підсилення об'єкту.

Висновок, що вплив збурення d зменшується із збільшенням сумарного коефіцієнту передачі як і вплив шумів виміру. Похибка П-регулятора

$$\delta_y = \frac{1}{1 + K_p K} \cdot r$$

Із збільшенням коефіцієнта підсилення зменшується похибка, проте надає запас по фазі, що може призвести до періодичних коливань.

Для компенсації статичної похибки можна використати метод компенсації, яких полягає в дії додаткового сигналу на виході регулятора. Скомпенсувати похибку можна також за допомогою зміни завдання

$$r' = r \cdot \frac{1 + K_p K}{K_p K}$$

I-регулятор

$$K = 0,$$

$$T_d = 0$$

$$u = r,$$

$e = 0$ – статична похибка відсутня.

Проте зменшується запас по фазі. Тому стійкий контур з П-регулятором може втратити стійкість з об'єктом першого порядку, що має транспортне запізнення та з об'єктом другого порядку.

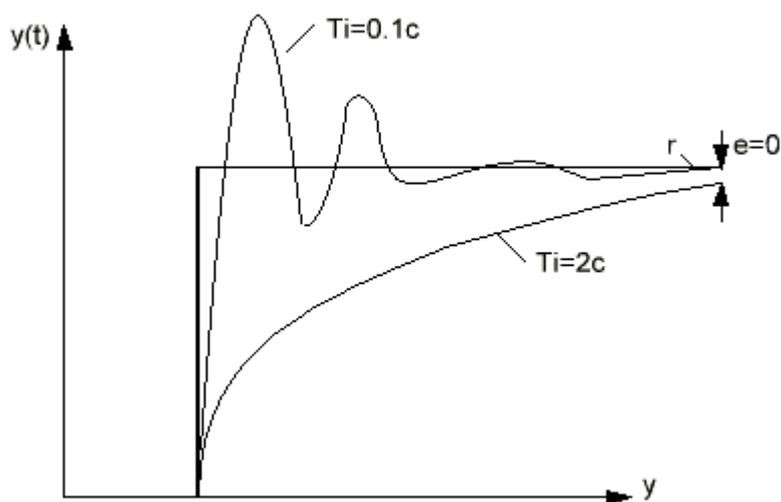


Рисунок 3.6 – Перехідний процес з I-регулятором

III-регулятор

$$T_d = 0$$

$$\delta_y(p) = \frac{T_i p}{T_i p + K T_i p \cdot P(p) + P(p)} \cdot r(p)$$

На відміну від I-регулятора підсилення на всіх частотах не може стати менше K_1 , що призводить до підвищення динамічної точності. Крім того III-регулятор в області низьких частот не вносить додаткового зсуву фаз, що збільшує запас стійкості. На відміну від II-регулятора статична похибка відсутня. I-регулятор в порівнянні з III-регулятором затягує перехідний процес.

PD-регулятор

PD-регулятор на високих частотах має великий коефіцієнт підсилення, що збільшує точність на початку перехідного процесу. В установленому режимі він вироджується в II-регулятор зі статичною похибкою, якщо її компенсувати як в II-регуляторах, то збільшується похибка на початку перехідного процесу. Тому на практиці PD-регулятори майже не використовуються. Підвищення запасу стійкості при малих значення T_D , із

збільшення TD збільшується коефіцієнт підсилення на високих частотах, що може призвести до автоколивань.

ПД – регулятор

Застосування ПД-регулятора забезпечує стійкість системи та процесу, підвищення якості керування у випадках неможливості цього зробити за допомогою ПІ- регулятора. Із збільшенням сталої часу TD зменшується амплітуда коливань та збільшується коефіцієнт затухання. Проте це призводить до росту коефіцієнту підсилення на високих частотах, що може призвести до коливального режиму, особливо в системах із запізненням.

Модифікації ПД-регуляторів

Поява модифікацій пов'язана з вимогами до зменшення часу регулювання, ослаблення дії збурень, спрощення процедури налаштування, керування об'єктами з великими транспортними затримками.

1). Регулятор з ваговими коефіцієнтами при установці (рис. 3.7).

Якість регулювання можна покращити, якщо обчислювати похибку окрема для кожної складової: П, І, Д.

$$e = r - y$$

$$e_n = br - y;$$

$$e_\partial = c \cdot r - y;$$

$$e_i = e = r - y.$$

де b,c – вагові коефіцієнти налаштування.

$$U(t) = K \cdot e_n(t) + \frac{1}{T_i} \int e_i(t) dt + T_\partial \frac{de_\partial(t)}{dt}$$

$$U(p) = (bK + \frac{1}{T_i p} + cT_\partial p) \cdot r(p) - (K + \frac{1}{T_i p} + T_\partial p) \cdot y(p)$$

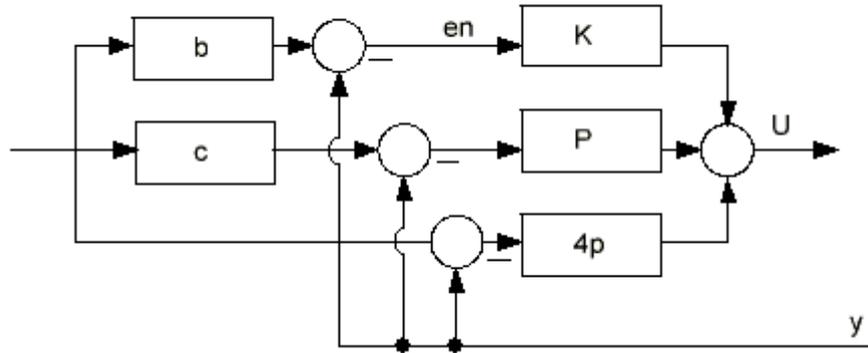


Рисунок 3.7 – Структурна схема ПІД-регулятора з ваговими коефіцієнтами при усталці

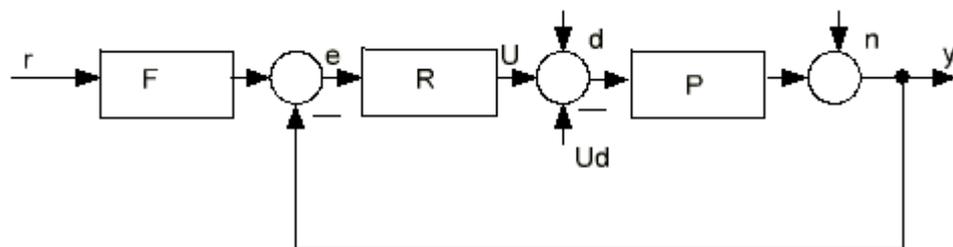


Рисунок 3,8 – Структурна схема регулятора з ваговими коефіцієнтами при установці

$$F(p) = \frac{bK + \frac{1}{T_i p} + cT_o p}{K + \frac{1}{T_i p} + T_o p}$$

Перевага: блок F не входить в контур керування, що дозволяє налаштування параметрів B та C незалежно від параметрів традиційного ПІД-регулятора в контурі.

Параметри B та C впливають на відгук системи при дії установи.

Параметри B та C не впливають на відгук системи при збурення та шумів.

Якщо $B = C = 0$ то регулятор називається І-ПІД регулятор.

Якщо $B = 1$, $C = 1$, то регулятор називається ПІ+Д регулятор.

Параметр C часто роблять близьким або рівним 0, що уникнути випадкових різких стрибків сигналу керування.

Регулятор з формуючим фільтром для сигналу установки

Подальшим удосконаленням ПІД-регулятора з ваговими коефіцієнтами є застосування фільтру для блоку F

$$F(p) = \frac{bK + \frac{1}{pT_i} + cpT_d}{K + \frac{1}{pT_i} + pT_d} \cdot F_r(p)$$

$$F_r(p) = \frac{1}{1 + pT_r}$$

Стала часу фільтру визначається за формулою

$$T_r = \frac{1}{\omega_r} \sqrt{m_r^2 - 1},$$

де m_r – показник коливальності на частоті ω_r .

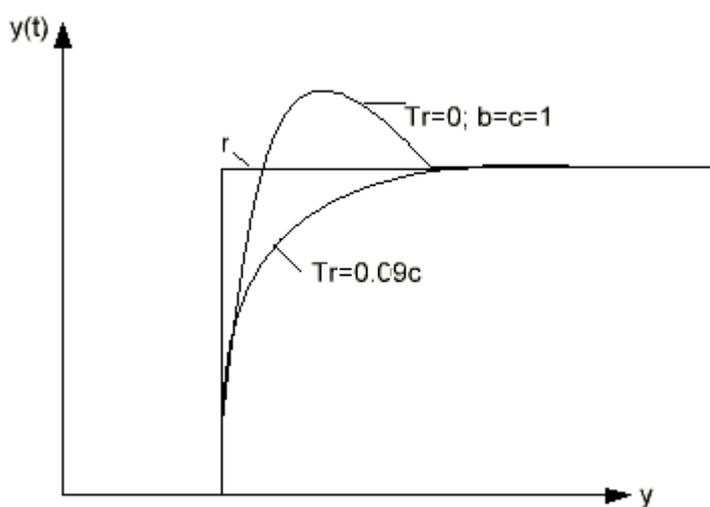


Рисунок 3.9 – Вигляд перехідних процесів

Використання формуючого фільтру зменшує перерегулювання, проте зтягує тривалість перехідного процесу.

Принцип розімкнутого керування

Якщо відомі збурення, які діють на систему та бажана реакція на зміну керуючої дії, то можливо застосувати регулювання без зворотного зв'язку. Перевагою є висока швидкодія, бо не потрібно чекати реакції системи, як при зворотному зв'язку. Недолік полягає в невисокій точності керування при похибках вимірювання збурень та зміні параметрів об'єкту. Цей принцип керування називається з прямим зв'язком.

Найкращі характеристики системи отримуються при об'єднанні прямого та зворотного керування. Такі регулятори називаються регуляторами з двома ступенями вільності, бо вирішуються дві задачі:

- 1) Забезпечення компенсації впливу збурень та робастності системи за допомогою регулятора із зворотнім зв'язком
- 2) Забезпечення заданої реакції на керуючу дію за допомогою регулятора прямого зв'язку.

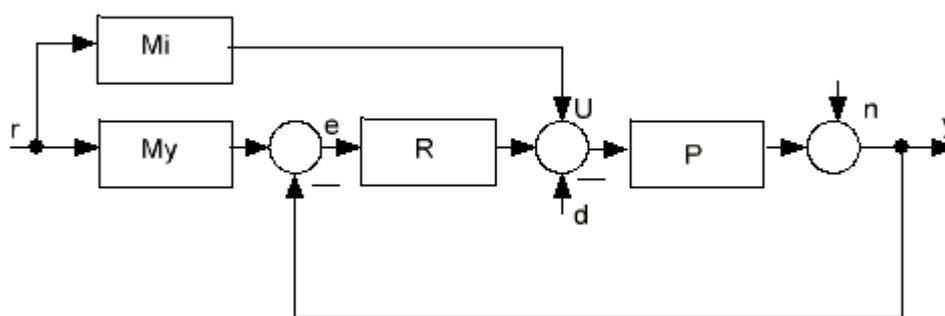


Рисунок 3.10 – Структурна схема регулятора з двома степенями вільності

Передаточна функція замкнутої системи

$$W_3(p) = M_y + \frac{pM_i - M_y}{1 + PR}$$

Для отримання в ідеалі нульової похибки ($e = 0$)

$$\frac{pM_i - M_y}{1 + PR} = 0$$

це можна зробити двома шляхами:

- 1) нескінченним коефіцієнтом передачі замкнутого контура PR;
- 2) передаточна функція вибирається за умови $M_i = p^{-1} \cdot M_y$;

Часто знаходження зворотної передаточної функції об'єкту нашоується на проблему фізичної реалізації. Наприклад, якщо об'єкт із транспортним запізненням.

Регулятори відношень

Застосовуються для підтримки не реальних значень змінних, а їх співвідношення.

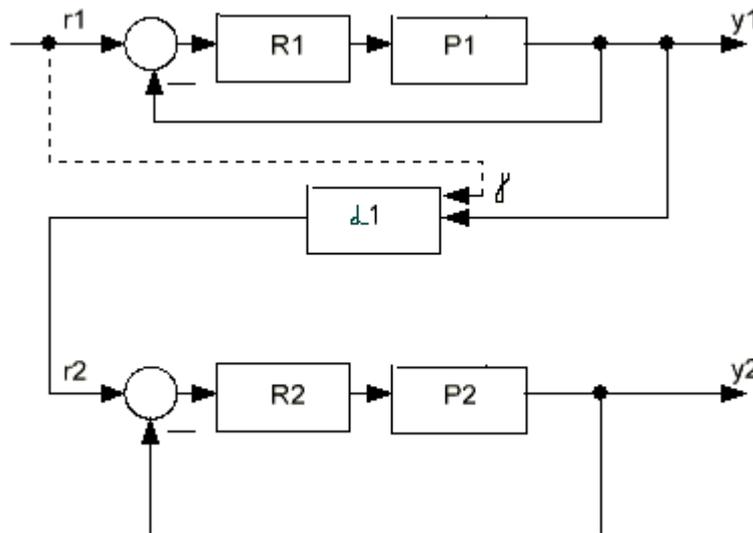


Рисунок 3.10 – Структурна схема регулятора відношень

$$r_2 = \alpha \cdot y_1(t)$$

Перевага: використання сигналу y замість r_1 підвищує точність.

Недолік: y_1 змінюється відповідно до динаміки першої системи, тобто з деякою затримкою відносно r_1 .

Тому для покращення якості r_2 змінюють за такою схемою

$$r_2 = \alpha[\gamma \cdot r_1(t) + (1 - \gamma) \cdot y_1(t)]$$

3.3. Контрольні запитання до розділу 3

1. Ідентифікація моделей динамічних систем.
2. Класичний ПД-регулятор.
3. Модифікації ПД-регуляторів.
4. Регулятори відношень.
5. ПД-регулятори для систем з транспортною затримкою.
6. Особливості реальних регуляторів.
7. Якість керування та налаштування параметрів ПД-регуляторів.

4. РОЗРОБКА ПРОЕКТІВ АВТОМАТИЗОВАНИХ СИСТЕМ УПРАВЛІННЯ ЕЛЕКТРОТЕХНОЛОГІЧНИМИ КОМПЛЕКСАМИ

4.1. Розробка функціональних схем автоматизації технологічних процесів

Задачі автоматизації вирішуються найбільш ефективно, якщо вони розробляються в процесі вивчення технологічного процесу. В цей період може виникнути необхідність змінити технологічну схему з метою адаптації до існуючих засобів автоматизації або техніко-економічних показників.

Основним документом, який визначає структуру системи автоматизації є функціональна схема. При розробці функціональної схеми автоматизації технологічного процесу вирішуються наступні задачі:

- 1) Отримання первісної інформації про стан технологічного процесу та обладнання;
- 2) Безпосередня дія на технологічний процес для керування змінними та корекції параметрів;
- 3) Контроль та реєстрація параметрів процесу та стану технологічного обладнання.

Для вирішення вказаних задач крім розробників АСУТП залучаються технологи контрольно-вимірювальних приладів і автоматики.

Попередньо створюється таблиця сигналів з їх параметрами та схемами, яка є основою не тільки для розробки функціональних схем, але і для SCADA системи.

Вимоги до оформлення функціональних схем

Функціональна схема виконується у вигляді креслення у відповідності до стандартів. На схемі схематично зображуються елементи у вигляді умовно-графічних позначень:

- 1) Технологічне обладнання;
- 2) Комунікації;

3) Органи керування та зв'язок з технологічним обладнанням.

Для великих об'єктів зі значним об'ємом автоматизації окремо виконуються схеми автоматичного контролю керування та сигналізації.

В нижній частині схеми у вигляді прямокутників зображуються пульти керування, щити із засобами автоматизації, а також мікропроцесорні засоби автоматизації, які позначаються ПЛК і розписуються сигнали або КТЗ ОП – комплекс технологічних засобів операторського приміщення.

Контури технологічного обладнання, трубопроводи, пульти-щити виконуються лінією товщиною 0,6..1,5 мм. Лінії зв'язку товщиною 0,2..0,3 мм. Всі прилади та засоби автоматизації товщиною 0,5..0,6 мм.

Технологічне обладнання на схемі повинно відповідати технологічній схемі, проте зображуватися спрощено, без другорядних конструктивних деталей. Внутрішні елементи та деталі обладнання показуються тільки тоді, якщо вони зв'язані з приладами та засобами автоматизації.

На трубопроводах показують тільки ті вентиля, засувки, клапани, які беруть участь в системі автоматизації. Не рекомендується показувати деталі допоміжного призначення.

Допускається стінки технологічних агрегатів, повітропроводів та ін., зображати двома паралельними лініями, якщо необхідно пояснити принцип з'єднання з засобами автоматизації.

На лініях трубопроводів за допомогою стрілок показується напрямок руху речовини. Трубопроводи, які йдуть від кінцевих апаратів або до них підходять і в яких немає засобів автоматизації - обриваються та закінчуються/ починаються стрілкою. Над стрілкою пишеться пояснювальний надпис.

Рідини та газу, що передаються по трубопроводам позначаються цифрами. Мінімальна відстань між цифрами одного трубопроводу 50 мм. На мнемосхемах певна рідина та газ мають свій колір:

- Вода – 1 – 1 – зелений колір
- Пара – 2 – 2 – рожевий колір
- Повітря – 3 – 3 – блакитний

Азот – 4 – 4 – темно-жовтий

Кисень – 5 - 5 – синій

Аміак – 11 – 11 – сірий

Кислота – 12 – 12 – оливковий

Луги – 13 – 13 – сіро-коричневий

Масило – 14 – 14 – коричневий

Рідке паливо – 15 – 15 – жовтий

Противопожежний трубопровід – 26 – 26 – червоний

Вакуум – 27 – 27 – світло-сірий

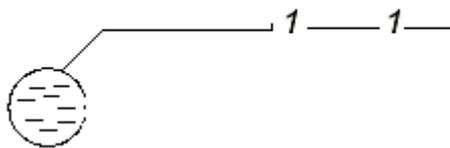


Рисунок 4.1 – Позначення трубопроводу з водою

Трубопроводи з рідиною або газом, які є основними в проекті можуть зображуватися червоним або чорним кольором. Допускається використання букв:

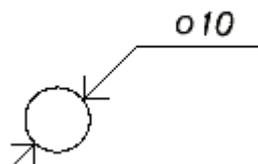
1Z – 1Z - чиста вода

2N - 2N – перегрітий пар

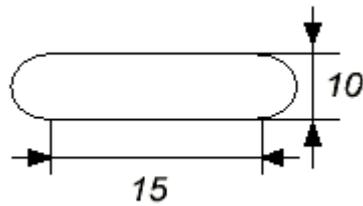
2H - 2H – насичений пар

Умовні графічні позначення (УГП) приладів та засобів автоматизації

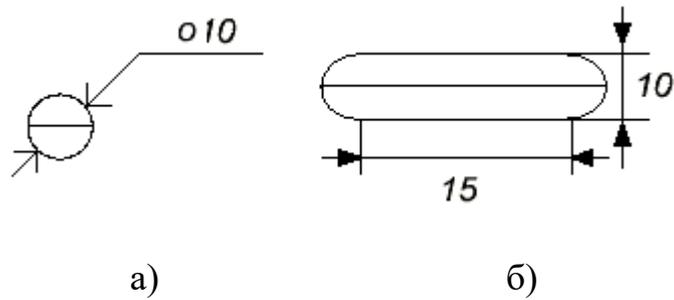
1. Первісний вимірювальний перетворювач (датчик)



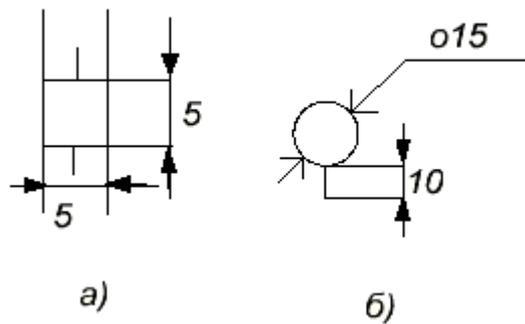
Допускається зображення в наступному вигляді



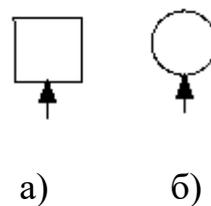
Прилад, який встановлюється на пульті чи щиті



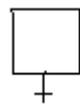
2. Виконавчий механізм а); б) електричний виконавчий механізм



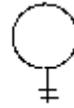
3. Виконавчий механізм, який закриває регулюючий орган при зникненні подачі енергії або керуючого сигналу



4. Виконавчий механізм, який при зникненні енергії або керуючого сигналу залишає регулюючий орган в поточному стані

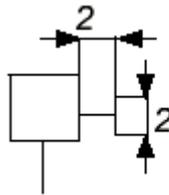


a)



б)

5. Виконавчий механізм з додатковим ручним приводом

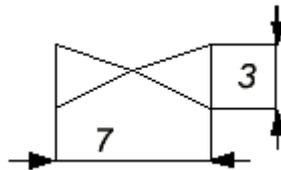


a)



б)

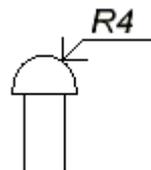
6. Регулюючий орган



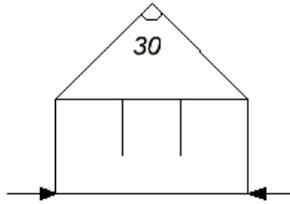
При розташуванні елементів на схемі бажано дотримуватися відстаней між центрами умовних графічних позначень кратних 12 мм.

Додатково на функціональних схемах автоматизації використовуються позначення зі схем електричних принципів:

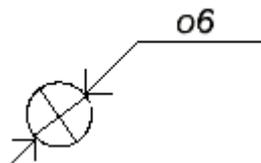
1. Дзвінок



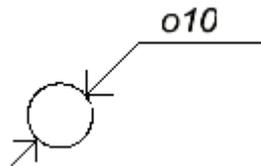
2. Сирена електрична або пневматична



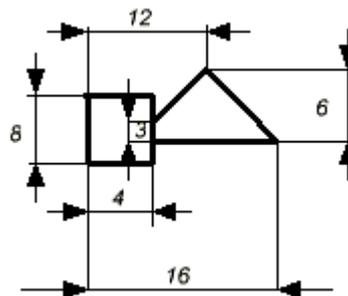
3. Лампа сигнальна або табло



4. Двигун електричний



5. Гудок електричний



Буквені позначення величин та функціональних ознак приладів

Таблиця 4.1 – Позначення величин та приладів

Позначення	Вимірювальна величина		Функції приладу		
	Основне значення (перша буква)	Додаткові позначення (друга буква)	Відображення інформації	Формування вихідного сигналу	Додаткові значення
А			Сигналізація		
В	Резервна буква				
С				Регулювання керування	
Д	Густина, щільність	Різниця, перепад			
Е	Електрична величина				
F	Витрати	Відношення, доля, дріб			
G	Розмір, положення, переміщення				
Н	Ручна дія				Верхня межа вимірювання величини
І			Показання		
J		Автоматичне переключення			
К	Час, часова				

	програма				
L	Рівень				Нижня межа вимірюва ння величини
M	Вологість				
N	Резервні букви				
O	Резервні букви				
P	Тиск, вакуум				
Q	Якість, склад	Інтегрування, сума в часі			
R	Радіо актив.		Реєстрація		
S	Швидкість, частота			Вкл, викл, перекл.	
T	Температу ра				
U	Декілька різних величин				
V	В'язкість				
W	Маса				
X	Не рекомендована резервна буква				
Y	Резервна				
Z	Резервна				

Застосовуються також додаткові позначення:

- 1) E – первісний перетворювач;
- 2) T – проміжне перетворення;
- 3) K – станція керування;
- 4) Y – перетворення, обчислення функції;
- 5) S – сигналізація;

При побудові обчислювальних пристроїв та перетворення сигналів:

- 1) Y – сума;
- 2) G - гідравлічний сигнал;

A- аналоговий сигнал;

- 3) D – дискретний сигнал;
- 4) X – перемноження;
- 5) f^n – піднесення в степінь;
- 6) $:$ – ділення сигналів;
- 7) $\sqrt[n]{}$ – добування кореня;
- 8) \max (\min) – обмеження сигналів.

Елементи на функціональних схемах позначаються в верхній та нижній частині умовно-графічного позначення.

В нижній частині за допомогою цифр або цифр та букв вказується позиційний номер елемента, починаючи від датчика і до регулюючого елемента. Перша цифра показує функціональну групу елементів.

Рекомендується наступний пріоритет позначення функціональних груп:

- 1 – температура
- 2 – тиск, розрідження вакуум
- 3 – витрати чи кількість
- 4 – рівень
- 5 – вологість
- 6 – щільність
- 7 – в'язкість
- 8 – концентрація
- 9 – запиленість, колірність, задимленість, мутність
- 10 – теплота згоряння
- 11 – кількість теплоти
- 12 – сила звуку (шумність)
- 13 – вібрація, частота механічних коливань
- 14 – лінійна швидкість

- 15 – частота обертання
- 16 – лінійне переміщення та довжина
- 17 – положення регулюючого органу
- 18 – товщина
- 19 – приріст (різниці значень)
- 20 – доза радіоактивного випромінювання

Номер елемента у функціональній групі позначається малими буквами кирилиці або цифрами через дефіс.

У верхній частині УГП вказуються букви і позначення вимірювальної величини та функціональної ознаки елемента.

Позначення здійснюється зліва на право в наступній послідовності:

- 1) Основна вимірювальна величина;
- 2) Уточнення основної величини;
- 3) Функціональні ознаки: покази, реєстрація, регулювання (рис.4,2).

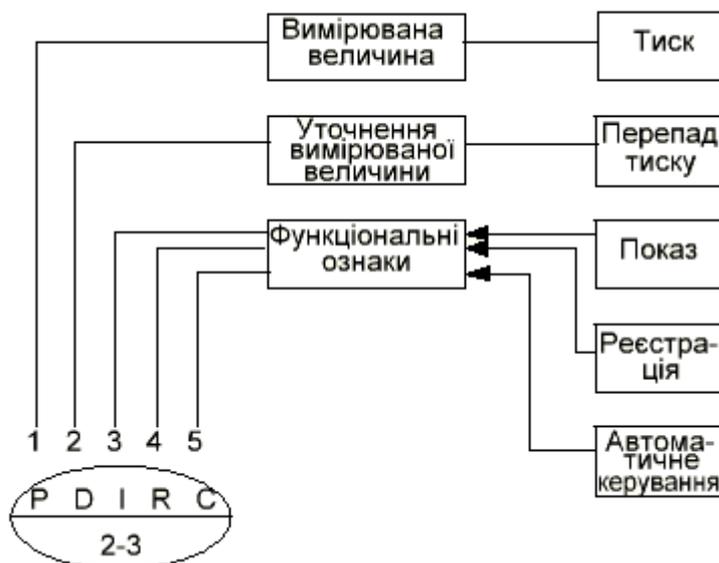


Рисунок 4.2 – Приклад умовного графічного позначення

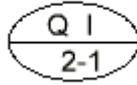


L – рівень

C – автоматичне керування

S – сигналізація

N – із блокуванням верхньої мережі



Q – вміст

I – показ

Тобто показує вміст кисню.

Приклад контуру регулювання температури води у котлі показано на рис. 4.3.

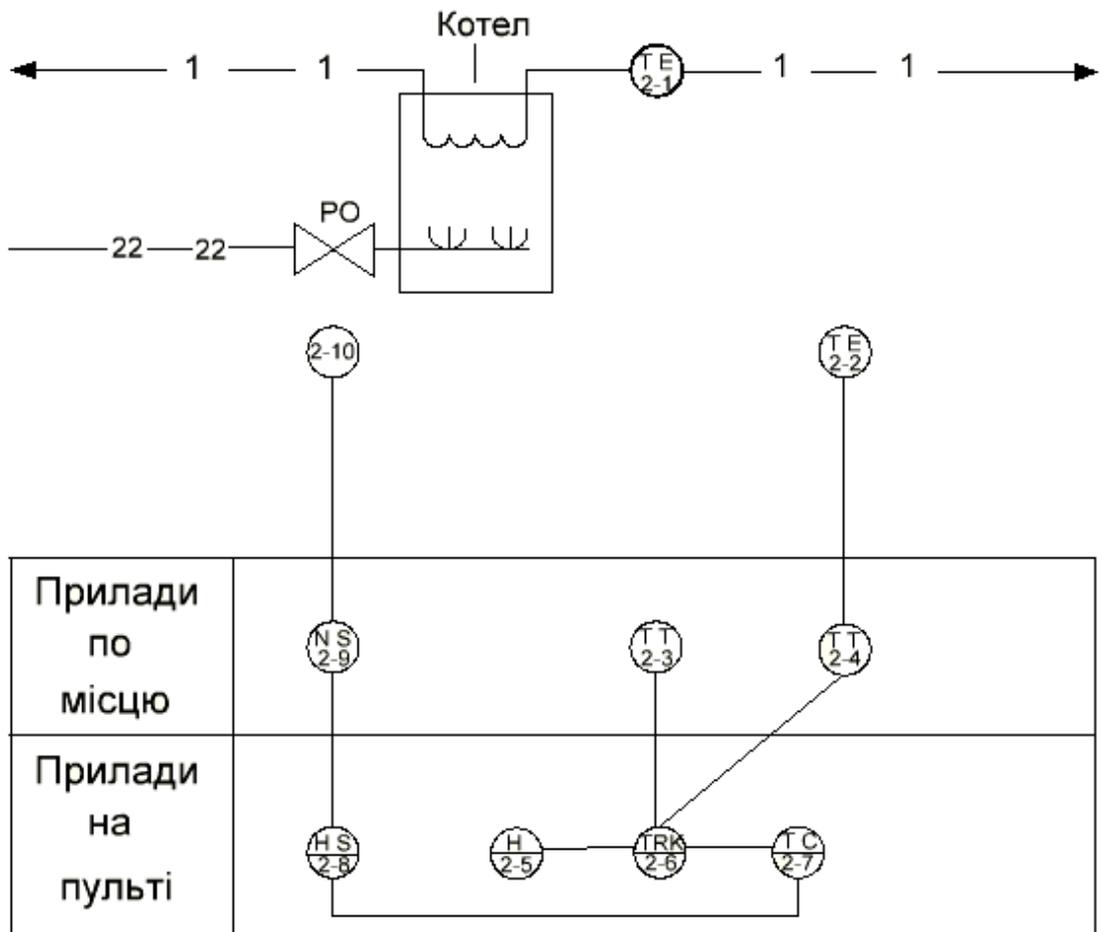


Рисунок 4.3 – Приклад контуру регулювання температури води у котлі

На рис. 4.3 позначено:

- 2-1 – терморезистор вимірює температуру гарячої води
- 2-2 – терморезистор вимірює температуру повітря
- 2-3 – перетворювачі сигналів
- 2-4 – задатчик температури
- 2-5 – задатчик температури
- 2-6 – блок регулювання
- 2-7 – блок керування
- 2-8 – ручний перемикач
- 2-9 – магнітний пускач
- 2-10 – двигун

Візуалізація технологічного процесу

Спосіб відображення інформації про стан технологічного обладнання та параметри технологічного процесу на операторських панелях або моніторах ПК. Візуалізація реалізується за допомогою декількох екранів (вікон), які можуть мати ієрархічну структуру.

Основу системи складають мнемосхеми. Інформація на мнемосхемах представляється у вигляді статичних рисунків та за допомогою анімації: зміна кольору об'єкту в залежності від його стану; мерехтіння елементів; зміна графічного образу чи розміру використання мультиплікації, переміщення об'єктів по екрану.

На мнемосхемах відображається наступні основні елементи:

- 1) Графічне зображення технологічного процесу;
- 2) Елементи керування обладнанням та системи автоматичного керування;
- 3) Технологічна сигналізація;
- 4) Аварійна сигналізація;
- 5) Тренди / графіки;
- 6) Контроль якості продукції.

Принципи проектування операторського інтерфейсу

- 1) Дружній інтерфейс;

- 2) Простота;
- 3) Мінімально-необхідна кількість інформації;
- 4) Максимум комфорту для оператора;
- 5) Конгитивна графіка – здатність визначати закономірності.

Дизайн інтерфейсу

- 1) Щільність заповнення екрану: в таблицях кожен 5-й рядок вільний і 5 пропусків між словами; 50% – займає фон.
- 2) Привертання уваги: основні елементи процесу повинні візуально відрізнятися або розташовуватися в верхньому лівому куті екрану; логічно зв'язані елементи об'єднувати кольором або рамкою; текстові повідомлення групувати праворуч, зображення ліворуч.
- 3) Обмежена кількість шрифтів.
- 4) Колір: переважно пастельні тони; обмежена кількість кольорів; об'єкти одного кольору сприймаються людиною як зв'язані; враховувати вплив кольору на людину:
 - Блакитно-зелений – заспокоює
 - Жовтий – вселяє оптимізм
 - Коричневий – пригнічує
 - Червоно-фіолетовий – тривога
 - Чорний – сприяє виникненню головних болів, проте зменшує кількість помилок.

Приклад проекту графічного екрану, створеного студентом освітньої програми представлено у додатку А.

4.2. Особливості проектування АСУ ТП в SCADA Trace Mode

SCADA-система Trace Mode версії 6. Всі програми ТМ-6 розділяються на:

- 1) Інструментальні системи розробки;
- 2) Виконавчі модулі.

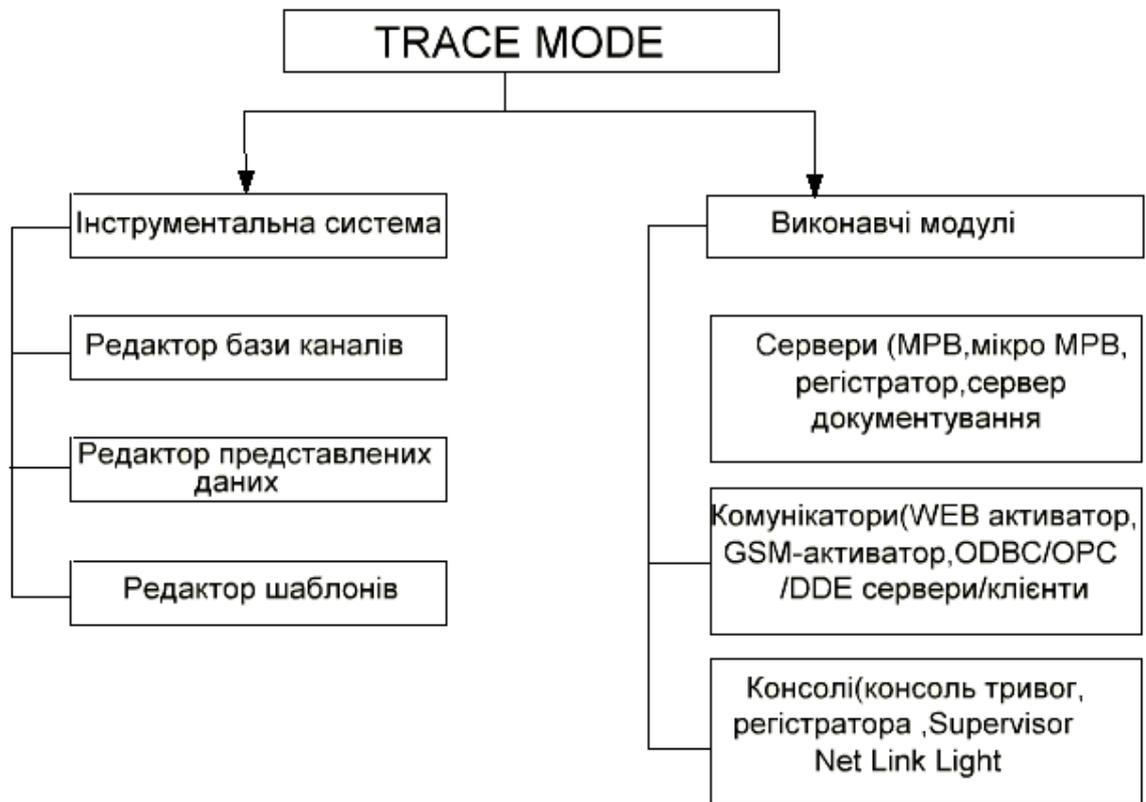


Рисунок 4.4 – Структура SCADA-система Trace Mode

Всі редактори об'єднані в одне інтегроване середовище розробки. Всі змінні проекту, до яких би елементів вони не відносились, зберігаються в єдиній базі даних, що дозволяє легко зв'язувати змінні різних контролерів та ПК.

В редакторі бази каналів створюється математична основа системи керування, а саме описується конфігурація всіх робочих ситуацій, контролерів, пристроїв зв'язку з об'єктом, а також налаштовується інформаційні потоки між ними. Тут описуються всі вхідні і вихідні сигнали, задаються періоди опитування, налаштовуються закони первинної обробки сигналів, встановлюються технологічні межі сигналів, здійснюється архівування технологічних параметрів та мережений обмін. В результаті роботи редактора створюються математичні та інформаційні структури проекту, які включають

набір баз каналів, файли конфігурування всіх контролерів та операторських станцій, файли конфігурування всіх проектів.

Редактор представлення даних призначено для створення графічної частини проекту (набори мнемосхем операторських станцій). Спочатку створюється статичний рисунок технологічного процесу. Поверх нього розміщуються динамічні форми для відображення та керування. Крім стандартних форм відображення в проекті можна встановити графічні форми користувача.

Редактор шаблонів призначено для створення шаблонів документів для вводу інформації.

Виконавча система Trace Mode складається з виконавчих модулів – це програмні модулі. В реальному часі виконуються складові проекту, що розміщені на комп'ютерах та контролерах.

Існує ряд модулів, які використовуються для оперативного та адміністративного рівнів керування. Це глобальний реєстратор, сервер документування, Web та ISM конвертори.

Глобальний реєстратор служить для надійного зберігання архівів технологічного процесу. Він архівує дані, що надходять з моніторів реального часу MPB (6400 параметрів з дискретністю 0,001 с), забезпечує автоматичне відновлення даних після збою, а також може передавати архівні дані для перегляду моніторами типу Supervisor. Глобальний реєстратор даних може виступати також як OPC та DDE сервер та підтримує обмін з базами даних через DDBC.

Сервер документування служить для документування технологічної документації як спеціальний модуль – сервер документування. Документування здійснюється за шаблонами розробленими в редакторі шаблонів. Час та умова генерування документу, ім'я файлу або пристрою друку описується в спеціальних програмах, які називаються сценаріями.

Сервер NetLink Light використовується для документування технологічної операції по команді MPB. власному сценарію або команді

оператора. Документи можуть бути роздруковані, відправленні на email або опубліковані на Web сервері.

Утиліта консоль тривоги дозволяє переглядати звіти тривоги з різних МРВ для одного проекту. Будь-яка робоча станція системи Trace Mode може виступати в якості WEB-сервера, що дозволяє керувати технологічним процесом через Internet. Для реалізації даного режиму використовується WEB-активатор, що виступає в якості WEB-шлюза локальної мережі АСУТП. Це дозволяє легко перетворити існуючий АСУТП в Інтернет/Інтранет систему без зміни баз даних реального часу (без зміни каналів). Доступ здійснюється за допомогою звичайного браузера. Зв'язок із МРВ здійснюється звичними засобами (на основі інтерфейсу RS-232, інфра-червоного порту, протоколу TCP/IP).

Для забезпечення оперативних користувачів, АСУ оперативною інформацією використовується ISM-активатор. Він дозволяє дистанційно моніторити та керувати технологічним процесом за допомогою звичайних портативних комп'ютерів. ISM-активатор може приймати в реальному часі інформацію від 64000 датчиків. ISM-активатори використовуються в нафтових, газових, електричних, теплових компаніях, комунальних та охоронних службах.

Раніше АСУТП мали централізовану структуру, основу якої складала потужна ЕОМ та до якої та від якої тягнулися тисячі ліній зв'язку від датчиків, виконавчих пристроїв. Перевагою такої структури є:

- 1) Дешевше обслуговування.
- 2) Простіше для адміністративного керування.

Недоліком є:

- 1) Велика кількість ліній зв'язку.
- 2) Можливість аварійних ситуацій при виході ЕОМ з ладу.
- 3) Необхідність резервування.

З появою персональних комп'ютерів та контролерів набула популярності розподілена структура АСУТП. Перевага – менше ліній зв'язку, більша

надійність. Недолік – більш громістке обслуговування та адміністративне керування.

Trace Mode реалізує ідеологію розподілених комплексів з трьома рівнями керування:

- 1) Нижній рівень – рівень контролерів;
- 2) Верхній рівень – рівень операторських станцій;
- 3) Адміністративний рівень.

Великі АСУТП характеризуються сотнями вузлів (ПК та контролерів).

Рівень контролерів

Для створення цього рівня ТМб існують наступні монітори:

- 1) Мікро МРВ – монітор реального часу.
- 2) Мікро МРВ-модем +.
- 3) Мікро МРВ-GSM +.

Мікро МРВ – призначено для запуску на контролері, який зв'язаний з верхнім рівнем локальною мережею або послідовним інтерфейсом.

Мікро МРВ-модем + – для зв'язку по лініях комутації.

Мікро МРВ GSM + – для зв'язку по GSM-мережі.

Операторський рівень

- 1) МРВ – монітор реального часу.
- 2) NetLink МРВ;
- 3) NetLink Light.

Ці монітори дозволяють створити робочі станції оперативного керування.

МРВ зв'язується з іншими моніторами за допомогою драйверів та вбудованих протоколів.

NetLink МРВ – мережева робоча станція здійснює обмін між іншими моніторами через мережу або послідовний інтерфейс.

NetLink Light – мережений графічний термінал для створення додаткових робочих місць операторів. Він не має засобів математичної обробки і отримує необхідні дані з моніторів вищого рівня.

Адміністративний рівень

Призначено для аналізу функціонування виробництва по архівним даним та контроль поточного стану виробничого процесу. Для вирішення цієї задачі є монітор Supervisor. Це спеціалізована графічна консоль, яка може підключатися до монітора реального часу та глобального архіву. Крім того Supervisor може переходити в режим реального часу для керування процесом. В цьому випадку він працює як консоль NetLink MPB.

Основні поняття SCADA Trace Mode

- 1) Проект – це сукупність всіх математичних та графічних елементів, що функціонують на різних операторських станціях на контролерах та об'єднані інформаційними зв'язками та єдиною системою архівування. Проект може бути масштабним (сотні вузлів) або включати лише операторську станцію.
- 2) Частина проекту, яка розміщена на окремому комп'ютері чи контролері називається вузлом проекту. В одному проекті може бути максимум 128 вузлів.
- 3) База каналів – це сукупність усіх каналів, математичних об'єктів, програм, створених для кожного конкретного вузла.
- 4) Канал – це базове поняття системи. Це набір змінних та процедур, що мають налаштування, ідентифікатори та період перерахунку змінних.
- 5) Ідентифікатор каналу має 3 поля: ім'я, коментар, кодування.

В залежності від напрямку потоку інформації канали є вхідними, що отримують інформацію від датчиків чи суміжних систем, та вихідними.

Є 7 видів каналів:

- 1) FLOAT – для дійсних чисел (4 байти);
- 2) HEX16 – цілі числа без знаку (2 байти);
- 3) HEX32 – цілі числа зі знаком (4 байти);
- 4) Double Float – дійсні числа (8 байт);
- 5) TIME – дата та час;

- 6) Событие – для моніторингу об'єктів з визначенням появи або зникненням події.
- 7) CALL – служить для виклику інших компонент.

Канал FLOAT

Вхідний канал FLOAT INPUT має наступні інструменти для обробки вхідного сигналу: масштабування, фільтрація одиничних піків, фільтрація малих змін, експоненціальне згладжування.

Вихідний канал FLOAT OUTPUT: експоненціальне згладжування, лінійне згладжування, апертура, кліпірування (обмеження min та max), масштабування.

Канали мають проміжні змінні:

- 1) Вхідне значення I_n ;
- 2) Апаратне значення A ;
- 3) Реальне значення R ;
- 4) Вихідне значення Q .

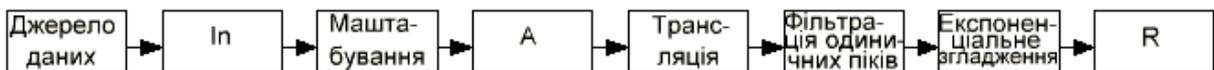


Рисунок 4.5 – Структура вхідного каналу FLOAT

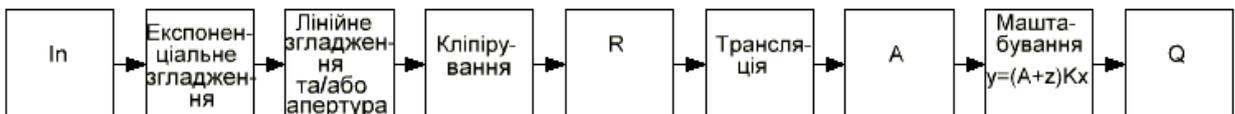


Рисунок 4.6 – Структура вихідного каналу FLOAT

Апаратне значення визначається за виразом

$$A = I_n \cdot Kx + z$$

де K_x – множник;
 z – зміщення.

Фільтрація піків полягає в тому, що значення змінної ігнорується на протязі одного такту, якщо зміна перевищила встановлене значення піку $DPic$.

Апертура – фільтрація малих змін, полягає в тому, що ігнорується зміна значення, якщо вона менше заданої величини.

Експоненціальне згладжування виконується, якщо значення $DSmoot$ знаходиться в діапазоні $(0;1]$. Згладження не виконується, якщо $DSmoot = 0$.

Крім основних значень вихідний канал має різні додаткові змінні: b меж сигналу, гістерезис (рис. 4.7).

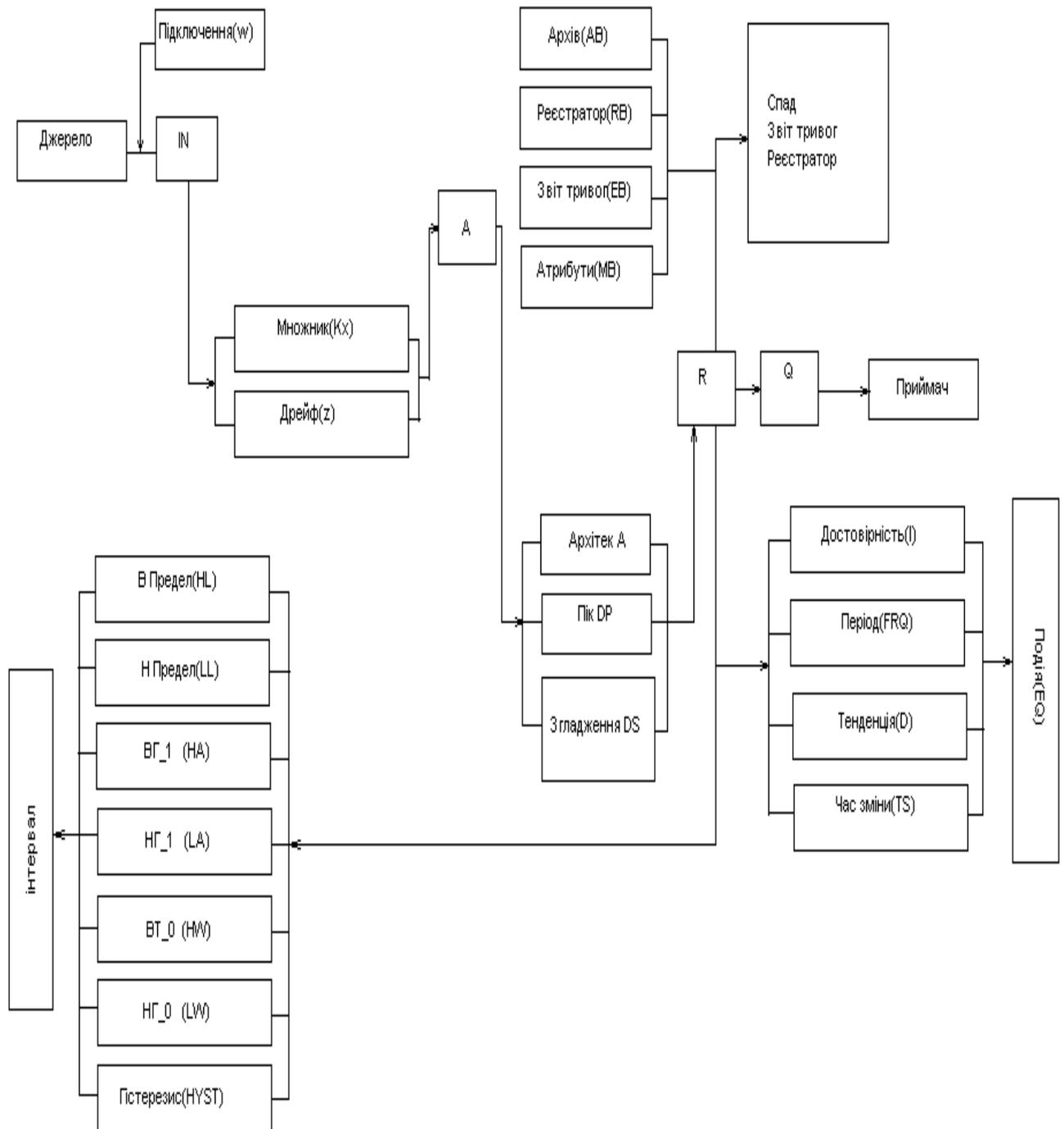


Рисунок 4.7 – Розширена структура вихідного каналу FLOAT

Крім аналогових сигналів вхідні та вихідні дані можуть бути представлені у дискретній формі. Для роботи з ними служить канал HEX 16, структури яких показано на рис. 4.8 та рис. 4.9.

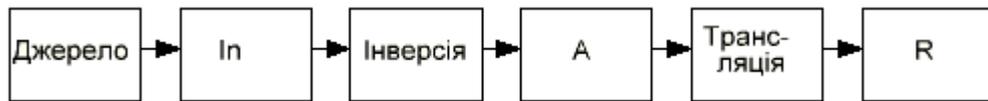


Рисунок 4.8 – Структура вхідного каналу НЕХ 16

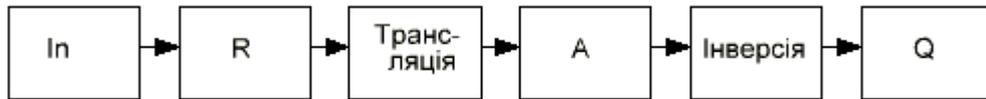


Рисунок 4.9 – Структура вихідного каналу НЕХ 16

В деяких випадках при обробці дискретних сигналів необхідно мати інверсний код. Для цього служить операція інверсія, яка активується відповідним прапорцем у вікні каналу.

Цифровий код може відображатися в 16-ому або 10-ому вигляді. Для цього існує прапорець DEC

ND = 1 10

ND = 0 16

До виходу каналів можна підключити процедуру, яка називається УПРАВЛЕННЯ, для реалізації алгоритмів керування. В якості аргументів програми можуть бути значення та атрибути з будь-яких каналів поточної бази.

Для реалізації алгоритму є 5 вбудованих мов:

- 1) Техно ST
- 2) Техно SFC
- 3) Техно FBD
- 4) Техно LD
- 5) Техно IL

Основною мовою є Техно ST. Алгоритми на мовах SFC, FBD, IL перед компіляцією транслуються в ST.

Проект складається з програм та каналів. через які здійснюється весь обмін даними. Зв'язок між значеннями каналів, аргументами програм та екранів

здійснюється за допомогою механізму «прив'язка». Це здійснюється за допомогою відповідної таблиці. Одному аргументу відповідає один рядок. Тип аргументу: In – для передачі в програму або на екрані; Out – для передачі з програми; Можна задати тип In/Out.

Детальний опис середовища Trace Mode наведено у додатку Б.

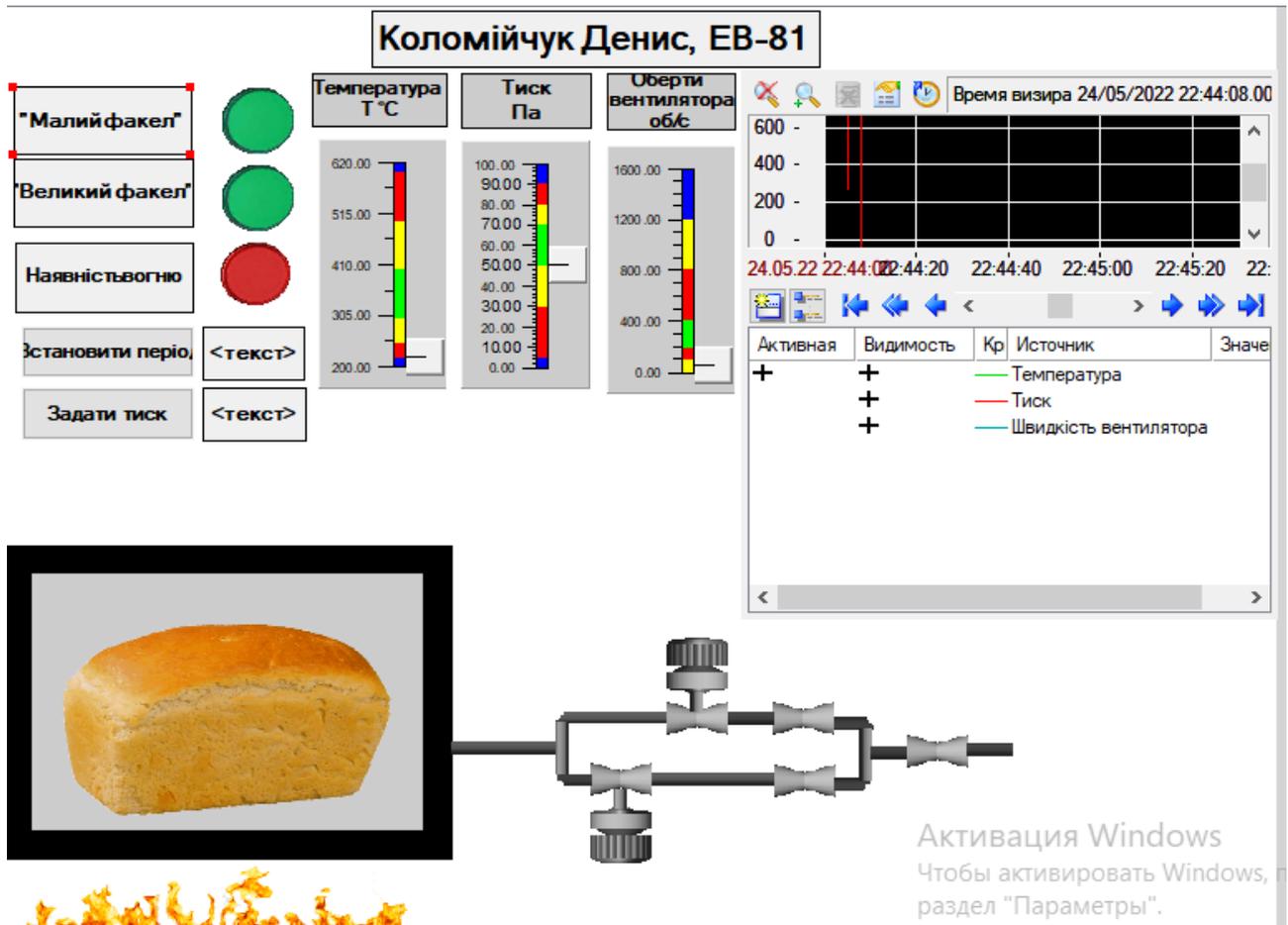
4.3. Контрольні запитання до розділу 4

1. Розробка проектів АСУТП.
2. Структура та складові SCADA Trace Mode.
3. Основні поняття SCADA Trace Mode, в т.ч. структура каналів.

СПИСОК ЛІТЕРАТУРИ

1. Автоматизація виробничих процесів: Підручник / І.В. Ельперін, О.М. Пупена, В.М. Сідлецький, С.М. Швед . – К.: Ліра-К, 2015. – 378 с.
2. Організація комп'ютерних мереж: Підручник / КПІ ім. Ігоря Сікорського ; Ю. А. Тарнавський, І. М. Кузьменко. – Київ: КПІ ім. Ігоря Сікорського, 2018. – 259 с.
3. Пупена О.М. Розроблення людино-машинних інтерфейсів та систем збирання даних з використанням програмних засобів SCADA/HMI: Посібник. – Київ, Ліра-К, 2020. – 594 с.
4. Промислові мережі та інтеграційні технології в автоматизованих системах. Навчальний. посібник / Пупена О.М., Ельперін І.В., Луцька Н.М., Ладанюк А.П. – К.: 2011. – 558 с.
5. Автоматизовані системи керування технологічними процесами: Підручник / За редакцією І.О. Фурмана. – Харків: Факт, 2006. – 317 с.
6. Головка Д.Б., Реґо К.Г., Скрипник Ю.О. Автоматика і автоматизація технологічних процесів: Підручник. – К.: Либідь, 1997. – 232 с.
7. Денисенко В.В. Комп'ютерне управління технологічним процесом, експериментом, обладнанням. – М.: Гаряча лінія-Телеком, 2009. – 608 с.
8. Бобух А.О. Автоматизовані системи керування технологічними процесами: Навчальний посібник. – Харків: ХНАМГ, 2006. – 185 с.
9. Лукінюк М.В. Автоматизація типових технологічних процесів: технологічні об'єкти керування та схеми автоматизації: Навчальний посібник. – К.: НТУУ “КПІ”, 2008. – 236 с.
10. Островерхов М.Я., Сільвестров А. М., Скриннік О.М. Системи і методи ідентифікації електротехнічних об'єктів: Монографія. – К.: НАУ, 2016. – 324 с.

А. ПРИКЛАД ПРОЕКТУ ГРАФІЧНОГО ЕКРАНУ



Б1. ОСНОВНІ ВІДОМОСТІ ПРО SCADA TRACE MODE

Trace Mode 6 (версія 6.10.1) – це програмний комплекс, призначений для розробки та запуску в реальному часі розподілених автоматизованих систем управління технологічними процесами (АСУТП) і вирішення ряду задач управління підприємством (АСУП). Для вирішення завдань АСУП використовується інтегрований пакет T-FACTORY. Комплекс програм Trace Mode 6 складається з трьох основних частин:

- інтегроване середовище розробки проекту (ІС);
- виконавчі модулі (монітори, МРВ);
- драйвери обміну.

Інтегроване середовище розробки проекту (ІС) – єдина програмна оболонка, що містить всі необхідні засоби для розробки проекту. Під проектом розуміється вся сукупність даних і алгоритмів функціонування розподіленої АСУ (АСУТП й/або АСУП), заданих засобами Trace Mode.

Результатом розробки проекту в ІС є створені файли, що містять необхідну інформацію про алгоритми роботи АСУ. Ці файли потім розміщуються на апаратних засобах (комп'ютерах і контролерах) і виконуються під керуванням виконавчих модулів Trace Mode.

В ІС вбудовані більше десяти редакторів, що автоматично відкриваються при виклику того або іншого компонента проекту, зокрема:

- редактор графічних екранних форм;
- редактор програм візуальною мовою Techno FBD;
- редактор програм візуальною мовою Techno SFC;
- редактор програм візуальною мовою Techno LD;
- редактор програм процедурною мовою Techno ST;
- редактор програм процедурною мовою Techno IL;
- редактор шаблонів документів;
- редактор SQL-запитів;

- редактор паспортів устаткування (EAM).

Виконавчі модулі (монітори, МРВ) – програмні модулі різного призначення, під керуванням яких у реальному часі виконуються складові частини проекту, що розташовані на окремих комп'ютерах чи в контролерах.

Складова частина проекту, що розташована на окремому комп'ютері або в контролері й виконується під управлінням одного або декількох виконавчих модулів, називається вузлом проекту.

У загальному випадку розміщення вузла на тому ж апаратному засобі, на якому він повинен виконуватися під управлінням монітора, не є обов'язковим – монітори можуть завантажувати вузли з віддалених апаратних засобів.

Драйвери обміну – драйвери, що використовуються МРВ для взаємодії із пристроями, протоколи обміну з якими не вбудовані в монітори.

Б1.1 Принципи розроблення проекту в інтегрованому середовищі

ІС поєднує в єдиній оболонці навігатор і набір редакторів для створення всіх складових проекту, а також має багатовіконний інтерфейс, представлений на рис. Б1.1.

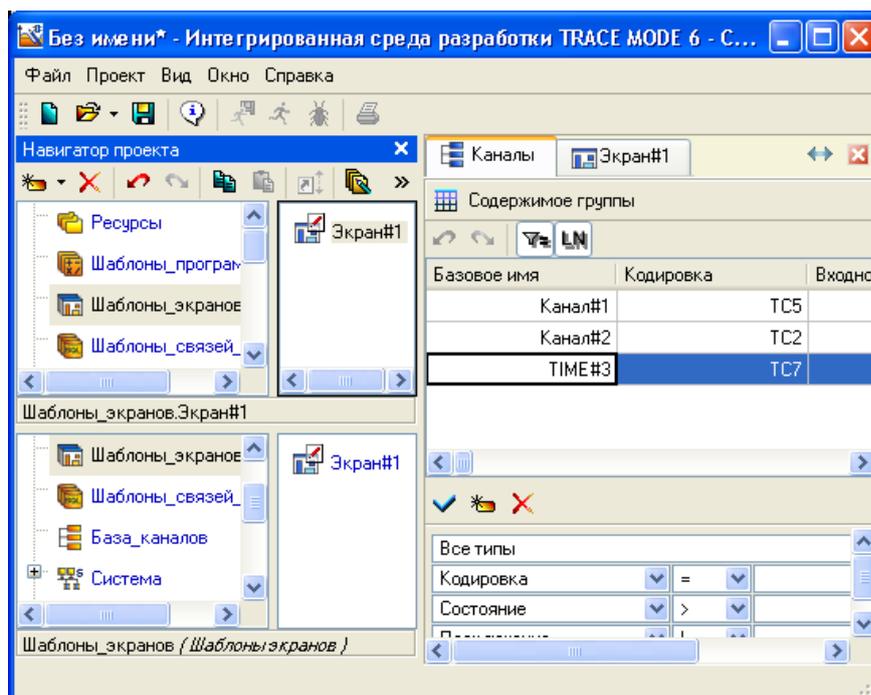


Рисунок Б1.1 – Интерфейс интегрированного среды разработки проекта

Кореневі групи дерева проекту (шари) визначені й створюються автоматично при створенні нового проекту (шари відображаються в лівому вікні навігатора). Елементарні структурні складові (листи структурного дерева) називаються компонентами проекту. Наприклад, компонентами проекту є: канал; канал, що викликає шаблон; шаблон; джерело даних і т.д.

Структура проекту редагується в навігаторі за допомогою команд меню **Проект**, контекстного меню й панелей інструментів, а також за допомогою методу drag-and-drop.

Б1.2 Технологія розробки проекту в інтегрованому середовищі

Розробка проекту в ІС включає наступні кроки:

- створення структури проекту в навігаторі;
- конфігурування або розробка структурних складових, наприклад, розробка шаблонів графічних екранів оператора, розробка шаблонів програм, опис джерел/приймачів і т.д.;
- конфігурування інформаційних потоків;
- вибір апаратних засобів АСУ (комп'ютерів, контролерів і т.п.);
- створення вузлів у шарі **Система** і їхнє конфігурування;
- розподіл каналів, створених у різних шарах структури, по вузлах і конфігурування інтерфейсів взаємодії компонентів в інформаційних потоках;
- збереження проекту в єдиний файл для наступного редагування;
- експорт вузлів у набори файлів для наступного запуску під керуванням моніторів по команді  **Сохранить для МРВ.**

Перераховані кроки, за винятком двох заключних, можуть виконуватися в довільному порядку.

Б1.3 Класифікація компонентів

За функціональним призначенням компоненти проекту відносяться до одного з наступних видів:

▶ **канали** – визначають алгоритм роботи та можуть створюватися в різних шарах, однак їхній остаточний розподіл по вузлах у шарі Система обов'язковий, інакше вони не будуть експортовані для МРВ;

▶ **шаблони** – при роботі в реальному часі можуть викликатися каналами з передачею параметрів, що настроюється при розробці проекту в ІС за допомогою прив'язки **аргументов** шаблону до каналів або **источникам/приемникам**;

▶ **источники/приемники** – описують внутрішні змінні різних пристроїв або додатків, з якими потрібно обмінюватися даними. Під пристроями розуміються контролери, а також зовнішні й внутрішні модулі/плати різного призначення, обмін з якими підтримується моніторами (у тому числі через драйвери). Системні змінні також створюються в ІС як джерела/приймачі. Джерела/приймачі є шаблонами каналів;

▶ **наборы ресурсов** – набори текстів, зображень і відеокліпів, які можуть бути використані при розробці шаблонів графічних екранів;

▶ **графические объекты** – представляють собою в загальному випадку кілька графічних елементів, згрупованих в один. Графічні об'єкти можуть бути використані при розробці шаблонів графічних екранів;

▶ **последовательные порты** – параметри СОМ-портів;

▶ **словари сообщений** – набори повідомлень, що генеруються при виникненні різних подій;

▶ **клеммы** – описують електричні контакти (наприклад, монтажних шаф), є елементами схеми електричних з'єднань АСУ.

Б1.4 Класифікація шарів

Визначені шари структури проекту мають наступне призначення:

- ▶ **Ресурси** – для створення користувацьких наборів текстів, зображень і відеокліпів, а також графічних об'єктів;

- ▶ **Шаблони програм** – для створення шаблонів програм;

- ▶ **Шаблони екранов** – для створення шаблонів графічних екранів;

- ▶ **Шаблони зв'язей с БД** – для створення шаблонів зв'язків з базами даних;

- ▶ **Шаблони документов** – для створення шаблонів документів (звітів);

- ▶ **База каналов** – є сховищем всіх каналів проекту. Виконувати операції з каналами (у тому числі створювати їх) можна в різних шарах, однак у всіх випадках ці операції реалізуються в шарі **База каналов**. У будь-якому іншому шарі, де виконується команда для здійснення операції з каналом, її результат тільки відображається – тому існують команди видалення й знищення каналів. У шарі **База каналов** можна починати розробку проекту;

- ▶ **Система** – для конфігурування вузлів та їх складових (вузол створюється як коренева група цього шару);

- ▶ **Источники/приемники** – для створення описів джерел/приймачів у різних пристроях і програмних додатках, обмін з якими підтримується моніторами, а також для конфігурування системних змінних;

- ▶ **Технология** – для розробки проекту від технології (тобто з угрупованням компонентів за ознакою їхньої приналежності до технологічного об'єкта). При налагодженні проекту шар **Технология** може відігравати роль вузла – для нього визначена команда **Сохранить для МРВ**. Крім того, для цього шару визначені команди взаємодії з технологічною базою даних;

- ▶ **Топология** – для розробки проекту від топології (тобто з угрупованням компонентів по місцю розташування);

- ▶ **КИПиА** (контрольно-измерительные приборы и автоматика) – для

опису електричних з'єднань АСУ;

► **Бібліотеки компонентів** – для створення бібліотек об'єктів – проектних рішень окремих завдань. Цей шар містить визначені групи **Системные** и **Пользовательские**. У групі **Системные** розміщені бібліотеки, підключені до ІС за згодою (автоматично).

Б1.5 Класифікація вузлів

Вузли проекту створюються як кореневі групи шару **Система**. Визначена назва вузла вказує на сімейство моніторів, для яких даний вузол призначений. Вузол може містити тільки ті компоненти, які підтримуються моніторами відповідного сімейства.

У загальному випадку, вузли можуть виконуватися під керуванням різних моніторів. Як правило, вузол виконується на окремому апаратному засобі. У випадку запуску двох і більше вузлів на одному апаратному засобі воно повинне бути обладнане відповідною кількістю мережевих карт.

Параметри вузлів задаються у відповідному редакторі.

Вузол **RTM** призначений для запуску на комп'ютері під керуванням виконавчих модулів сімейства **RTM** (MPB) – моніторів з підтримкою відображення графічних екранів оператора, підтримкою обміну по послідовному інтерфейсу й мережі з різним устаткуванням і виконуючим перерахування каналів всіх класів, крім каналів T-FACTORY.

Вузол **T-FACTORY** призначений для запуску на комп'ютері під керуванням виконавчих модулів сімейства **T-FACTORY** – моніторів для рішення завдань АСУП.

Вузол **MicroRTM** призначений для запуску на комп'ютері або в контролері під керуванням виконавчих модулів сімейства **Micro RTM**. Основна відмінність цих моніторів від MPB – відсутність підтримки відображення графічних екранів.

Вузол **Logger** призначений для запуску на комп'ютері під керуванням

виконавчого модуля **Logger** (реєстратор) – монітора, здатного вести архіви по каналах всіх вузлів проекту.

Вузол **EmbeddedRTM** призначений для запуску на комп'ютері або в контролері під керуванням виконавчих модулів сімейства **Embedded RTM** – моніторів з підтримкою мнемосхем, підтримкою обміну даними з устаткуванням по різних протоколах і виконуючому перерахунок каналів.

Вузол **NanoRTM** призначений для запуску в контролері під керуванням виконавчого модуля **Nano RTM** – монітора, аналогічного **Micro RTM**, але призначеного для роботи з малим числом каналів.

Вузол **Console** призначений для запуску на комп'ютері під керуванням виконавчих модулів, які, на відміну від МРВ, не виконують перерахунок каналів, призначених для роботи з даними. Консолі дозволяють одержувати дані від інших вузлів проекту по мережі, відображати їх на графічних екранах й керувати технологічним процесом з графіки. Консолі не можуть взаємодіяти з вузлами **T-FACTORY**.

Вузол **TFactory_Console** призначений для запуску на комп'ютері під керуванням виконавчих модулів, аналогічних консолям, але, крім того, здатних взаємодіяти з вузлами **T-FACTORY**.

Вузол **TM OPC_Server** призначений для запуску на комп'ютері під керуванням OPC-сервера Trace Mode 6.

Б2 МЕНЮ ТА ПАНЕЛЬ ІНСТРУМЕНТІВ ІНТЕГРОВАНОГО СЕРЕДОВИЩА

Оболонка ІС має головне меню, що включає пункти меню **Файл, Проект, Вид, Окно, Справка** та панель інструментів.

Редактори, вбудовані в ІС, мають свої меню та панелі інструментів, які при відкритті цих редакторів частково або повністю додаються до наявних в ІС. При відкритті редактора можливо також модифікування списку команд меню ІС. У випадку відкриття декількох редакторів, панелі інструментів і меню ІС відповідають редактору, вікно якого в даний момент є активним.

Меню й панель інструментів оболонки ІС доступні в усіх випадках.

Б2.1 Меню **Файл** та головна панель інструментів

Головна панель інструментів ІС представлена на рис. Б2.1.



Рисунок Б2.1 – Зовнішній вигляд головної панелі інструментів ІС

Панель включає наступні команди:

 **Новый (Ctrl-N)** – створити новий проект;

 – відкрити проект (файл із розширенням **prj**):

 **Открыть (Ctrl-O)** – вибрати файл у стандартному діалозі операційної системи;

 – вибрати файл зі списку останніх відкритих;

Импорт – по цій команді імпортуються проекти, що були розроблені у попередній версії Trace Mode 5.

 **Сохранить (Ctrl-S)** – зберегти проект у файл з розширенням **prj** з тим же ім'ям (користувацькі бібліотеки компонентів зберігаються по цій команді у файл **tmdevenv.tmul**);

Сохранить как... (Ctrl-Shift-S) – зберегти проект у файл з розширенням **prj** із завданням його імені (користувацькі бібліотеки компонентів зберігаються по цій команді у файл **tmdevenv.tmul**);

 **Информация о проекте** – відкрити однойменний діалог (ця команда доступна також з меню **Проект** і контекстного меню навігатора проекту). У цьому діалозі можна вказати автора проекту, організацію й коментар до проекту. У діалозі відображається час створення й час останньої зміни проекту. У нижній частині діалогу виводиться інформація про основні види компонентів проекту (максимально можливе число компонентів даного виду в проекті, використане число компонентів даного виду й число компонентів даного виду, що залишилося в розпорядженні розробника);

 **Сохранить для MRB** – експортувати вузли для наступного запуску під керуванням виконавчих модулів. Експорт одного вузла можливий з меню **Проект** і контекстного меню навігатора проекту;

 **Отладка** – завантажити виділений вузол у профайлер. Ця команда доступна після виконання команди **Сохранить для MRB** або **Сохранить узел для MRB**;

 **Шпион** – одержати в ІС реальні дані із працюючих вузлів;

 **Документировать проект (Ctrl-P)** – зберегти документацію проекту у файл;

Настройки ИС – відкрити діалог завдання загальних налаштувань ІС і редакторів шаблонів;

Последние файлы – показати список останніх файлів, відкритих в ІС. Обраний у списку файл завантажується в ІС;

Выход – вийти з інтегрованого середовища.

Б2.2 Панель переходів між вікнами інтегрованого середовища

Для переходів між відкритими вікнами редакторів використовується панель, що показана на рис. Б2.2. Жовтогаряча смужка позначає активне вікно.



Рисунок Б2.2 – Панель переходу між відкритими вікнами редакторів

Панель містить також наступні інструменти:

 – синхронізувати з деревом проекту (по цій команді в навігаторі проекту буде виділений компонент, вікно редактора якого активно, а також шар (група), що містить цей компонент);

 – закрити активне вікно.

Б2.3 Завдання загальних налаштувань інтегрованого середовища

Для завдання загальних налаштувань ІС і редакторів шаблонів призначений діалог, що відкривається по команді **Налаштування ІС** меню **Файл**. Зовнішній вигляд вікна налаштувань зображено на рис. Б2.3.

Налаштування ІС задаються на однойменній вкладці цього діалогу, набір інструментів якої змінюється залежно від обраного в лівому списку розділу, і вкладці **Базовий редактор**.

Б2.4 Вкладка **Интегрированная среда разработки**

Б2.4.1 Розділ **Общие**

При виборі цього розділу вкладка містить наступні інструменти:

► перемикачі режиму відображення редакторів компонентів – **Открывать редакторы как вкладки, MDI, SDI**.

У першому режимі кожен відкритий редактор займає всю робочу область ІС і зміна розмірів вікна редактору недоступна.

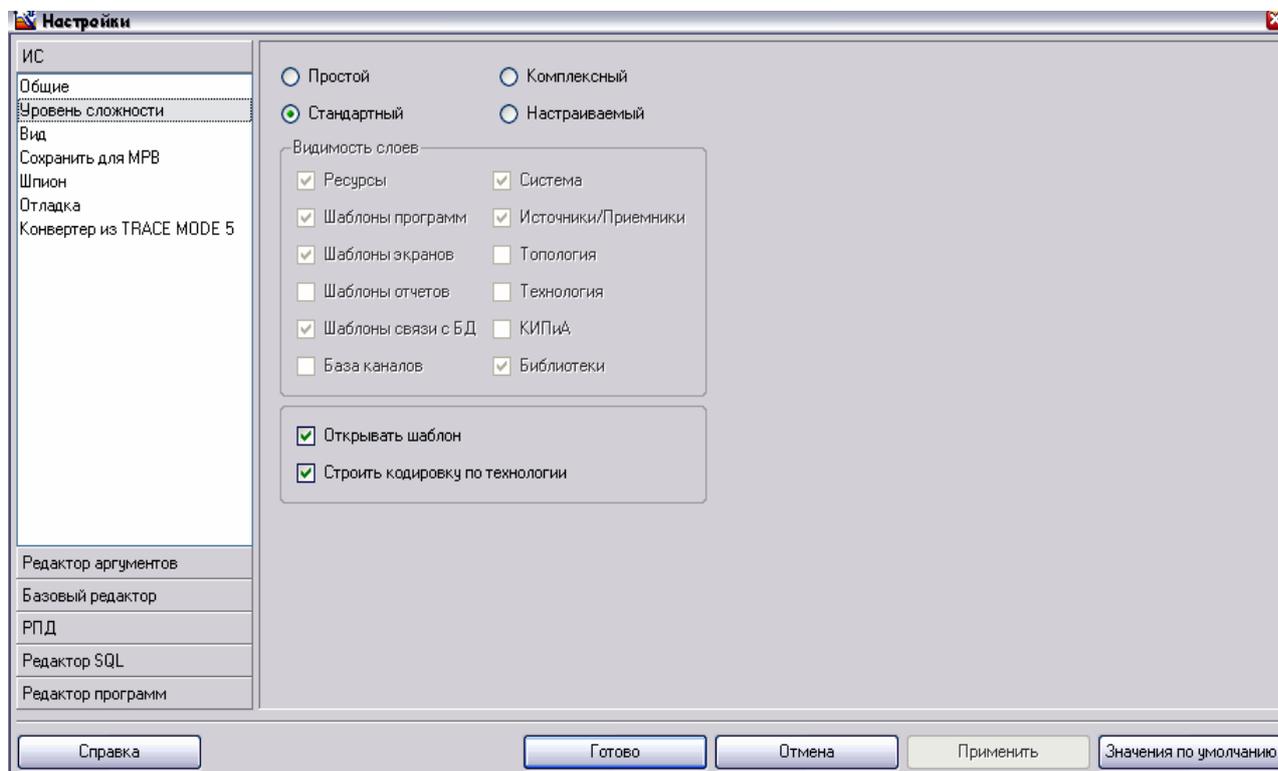


Рисунок Б2.3 – Зовнішній вигляд вікна налаштувань ІС

У режимі **MDI** розміри вікна кожного редактора можуть бути змінені, при цьому доступні стандартні інструменти вікон (☰ – відкрити меню, що містить стандартні команди для роботи з вікном, ☱ – мінімізувати вікно, ☲ – відновити вікно, ✖ – закрити вікно).

У режимі **SDI** при виділенні об'єкта структури в навігаторі відповідний редактор відкривається автоматично, займаючи всю робочу область ІС.

Б2.4.2 Розділ **Уровень сложности**

При виборі цього розділу вкладка містить перемикачі рівня складності

проекту (задають набір шарів, що будуть відображатися):

- ▶ **простой** – відображаються тільки шари **Ресурси, Система, Источники/Приемники та Библиотеки компонентов**;
- ▶ **стандартный** (встановлене за згодою) – відображаються ті ж шари, що й для простого рівня, та додатково всі шари шаблонів (екранів, програм, зв'язків із БД і документів);
- ▶ **комплексный** – відображаються всі шари, крім шару **База каналов**;
- ▶ **настраиваемый** – при виборі цього рівня в діалозі доступні перемикачі відображення всіх шарів, включаючи шар **База каналов**.

В2.4.3 Розділ **Сохранить для MPB**

Цей розділ містить наступні прапорці, що визначають ступінь деталізації інформації, що виводиться конвертером **FileCnv32.dll** у текстові файли **<ім'я файлу prj>_<ordinal>.cnv** при експорті по команді **Сохранить для MPB**:

- ▶ **Каналы по кодировке**;
- ▶ **Распределить экспорт**;
- ▶ **Файлы *.sav**;
- ▶ **Не создавать каналы CHCS**;
- ▶ **Стандартные объекты** – якщо цей прапорець встановлений, канали вузла відображаються в профайлері згрупованими в стандартні об'єкти (групи). Це угруповання має значення тільки для налагодження;
- ▶ **Подробный отчет**.

Список **Глубина отслеживания источников**, за допомогою якого задається порядковий номер джерела/приймача в ланцюжках зв'язків **канал – источник/приемник1 – источник/приемник2**, що буде використаний при експорті вузла.

В2.4.4 Розділ **Отладка**

При виборі цього розділу вкладка містить перемикачі типу профайлера, що буде запущений по команді  **Отладка** (тільки для налагодження вузлів **RTM, Console, T-Factory** й **T-Factory_Console**):

- ▶ Профайлер с поддержкой графических экранов;
- ▶ Профайлер без поддержки графических экранов.

Б2.4.5 Розділ **Конвертер из предыдущей версии**

При виборі цього розділу вкладка містить прапорець **Отключить конвертирование графической базы**. Установка цього прапорця відключає конвертування графічної бази проекту при його імпорті в Trace Mode 6.

Б2.5 Збереження проекту для редагування

По команді  **Сохранить (Ctrl-S)** або **Сохранить как (Ctrl-Shift-S)** із меню **Файл** проект зберігається в бінарний файл із розширенням **prj** (у директорії сервера проекту) для наступного редагування в ІС. В ІС передбачене резервування попередньої версії файлів **prj** й **tmul**. При повторному виконанні команди **Сохранить** розширення файлів, збережених раніше, змінюються відповідно на **~prj** й **~tmul**.

В ІС можна завантажити один проект. На одному комп'ютері можна запустити декілька ІС і завантажити в них різні проекти.

Б2.6 Збереження проекту для запуску

По команді  **Сохранить для МРВ** із меню **Файл** або панелі інструментів ІС всі вузли експортуються в набори файлів для їхнього наступного копіювання на апаратні засоби, на яких вони повинні виконуватися

під керуванням моніторів.

При виконанні команди **Сохранить для МРВ** у директорії сервера проекту створюється піддиректорія <ім'я файлу **prj** без розширення>, у якій для кожного вузла створюється папка з набором файлів. Папка вузла має ім'я <**name**>_<**ordinal+1**>, де **name** – ім'я, задане для вузла при його конфігуруванні в ІС, **ordinal** – порядковий номер вузла в шарі **Система**, починаючи з 0 (порядковий номер вузла в шарі не слід плутати з його індивідуальним номером).

Необхідною умовою експорту вузла є наявність у ньому хоча б одного каналу. По команді **Сохранить узел для МРВ** із меню **Проект** або контекстного меню навігатора виділений вузол (або шар **Технология**) експортується в довільну папку. При повторному експорті резервні копії вузла не створюються.

БЗ РЕДАГУВАННЯ СТРУКТУРИ ПРОЕКТУ

Навігатор має наступні засоби для редагування структури проекту:

- ▶ меню **Проект**;
- ▶ панелі інструментів;
- ▶ контекстне меню.

Крім того, у навігаторі підтримується метод перетягування об'єктів комп'ютерною мишкою (метод drag-and-drop), а також його модифікації (drag-and-drop з утриманням службових клавіш). Для конфігурації/розробки об'єктів структури в навігаторі передбачені команди **Свойства** й **Редактировать**, за допомогою яких для кожного об'єкта структурного дерева можуть бути відкриті відповідне вікно властивостей і редактор.

БЗ.1 Меню й головна панель інструментів навігатора проекту

Меню **Проект**, головна панель інструментів і контекстне меню навігатора проекту, що представлені на рис. БЗ.1, містять набір команд, що відповідає виділеному об'єкту структурного дерева. Для виділення об'єкта потрібно натиснути на ньому ЛК (лівою клавішею мишки), при цьому групове виділення об'єктів у навігаторі не підтримується.



Рисунок БЗ.1 – Зовнішній вигляд головної панелі інструментів навігатора проекту

Серед команд меню **Проект**, головної панелі інструментів і контекстне меню навігатора містяться як типові для створення компонентів (груп компонентів), роботи з буфером обміну й пошуку, так і специфічні:

 – такий вид здобуває типовий інструмент видалення  при виділенні каналу. При натисканні стрілки відкривається меню, що містить команди **Удалить**, що видалляє без вилучення із шару **База каналов**, і **Уничтожить**, що видалляє з вилученням із шару **База каналов**;

 – такий вид здобуває в деяких випадках типовий інструмент вставки . При натисканні стрілки відкривається додаткове меню, що містить команди  **Вставить** та  **Вставить с привязкой**;

 **Перейти по ссылке вниз/вверх** – по цій команді відкривається додаткове вікно навігатора, у якому:

- ▶ виділяється компонент, з яким зв'язаний даний компонент або який викликається даним компонентом (у випадку переходу по посиланню вниз);
- ▶ виділяється компонент, пов'язаний з даним компонентом або той, що викликає даний компонент (у випадку переходу по посиланню нагору). Якщо подібних компонентів декілька, один з них вибирається в списку, як це показано на рис. БЗ.2.

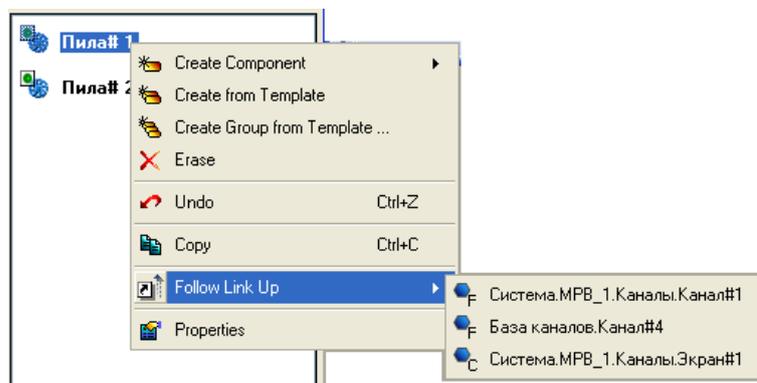


Рисунок БЗ.2 – Меню для вибору компонентів

Якщо в компоненті є зв'язки/виклики як вниз, так і нагору, даний інструмент приймає вид , і при натисканні стрілки відкривається додаткове меню, що містить команди **Перейти по ссылке вниз** та **Перейти по ссылке**

вверх;

 **Резервирование** – створити резервні вузли для виділеного вузла:

- **Нет** – не створювати (значення за згодою);
- **Дублированный** – створити один резервний вузол для виділеного вузла;
- **Троированный** – створити два резервних вузли для виділеного вузла;

 **Редактировать** – відкрити виділений об'єкт структурного дерева у відповідному редакторі;

 **Редактировать шаблон** – відкрити шаблон, викликаний даним компонентом, у відповідному редакторі;

Переименовать – перейти до редагування імені виділеного об'єкта структури. Для редагування імені можна також натиснути на ньому ЛК;

 **Свойства** – відкрити вікно властивостей об'єкта структурного дерева;

 **Сохранить узел для МРВ** – експортувати виділений вузол (шар **Технология**). По цій команді на екрані відображається діалог вибору місця розташування файлів вузла (за допомогою кнопки **Создать папку** та команд контекстного меню цього діалогу можливе керування файловою структурою);

 **Загрузить дамп узла** – відновити вміст і конфігурацію компонентів вузла з його файлу відновлення. Цей файл не слід плутати з файлом **<ім'я_вузла>.dump**, що створюється за згодою при експорті вузлів;

 **Загрузить в контроллер** – копіювання вузла в контролер;

 **Информация о проекте** – відкрити однойменний діалог. Ця команда може бути виконана також з контекстного меню навігатора проекту й за допомогою панелі інструментів ІС;

 **Импорт из БД** та  **Экспорт в БД** – ці команди, призначені для взаємодії з технологічною базою даних, доступні при виділенні шару **Технология**.

Б3.2 Створення об'єктів структури

Для створення об'єктів структури (компонентів і груп компонентів) використовуються типові команди меню **Проект**, контекстного меню й панелі інструментів навігатора. При створенні каналу класу **CALL** із установленою наперед властивістю **ВЫЗОВ** у відповідному шарі шаблонів створюється шаблон, що викликається каналом. У навігаторі можуть бути створені наступні канали класу CALL із наперед установленою властивістю **ВЫЗОВ**:

- ▶ **Экран** – канал з викликом шаблону екрана;
- ▶ **Программа** – канал з викликом шаблону програми;
- ▶ **Документ** – канал з викликом шаблону документа;
- ▶ **Связь с БД** – канал з викликом зв'язку з базою даних.

Меню **Проект**, контекстне меню й панель інструментів навігатора містять команди створення тільки тих об'єктів, які може містити виділений шар/група. При редагуванні збереженого проекту знову створені структурні об'єкти й об'єкти, що їх вміщують, виділяються в навігаторі синім кольором, як це зображено на рис. Б3.3.

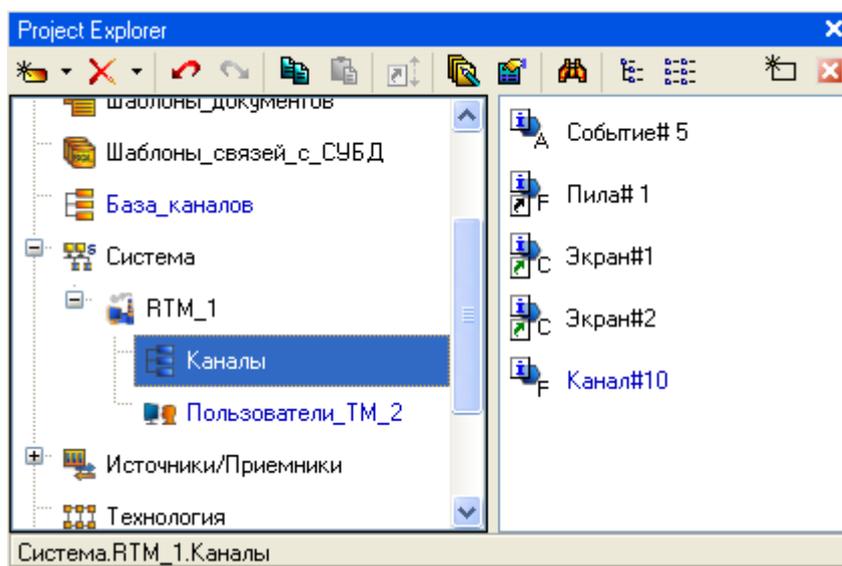


Рисунок Б3.3 – Виділення знову створених структурних об'єктів та об'єктів, що їх вміщують, при редагуванні збереженого проекту

Б3.3 Видалення об'єкта структури

Для видалення виділеного об'єкта структури (компонента або групи компонентів) використовується типова команда **Удалить**. При видаленні каналу (групи каналів) з будь-якого шару, крім шару **База каналов**, доступні дві команди – **Видалити** (видалити без вилучення із шару **База каналов**) і **Уничтожить** (видалити з вилученням із шару **База каналов**). Якщо шар **База каналов** прихований, команда видалення каналу недоступна – у цьому випадку канал можна тільки знищити.

ІС автоматично видаляє прив'язки аргументів до компонента, посилання на компонент і виклики компонента при видаленні цього компонента (у випадку каналу – при його знищенні).

Б3.4 Переміщення об'єктів структури

Операція переміщення включає дві дії – видалення об'єкта з місця його початкового розташування й вставку в зазначену групу (шар). Щоб перемістити об'єкт, потрібно перетягнути його мишею в потрібну групу (шар), утримуючи клавішу **SHIFT**. Якщо група (шар), на яку вказує курсор у процесі перетягування, може містити переміщуваний об'єкт, курсор приймає вид , у противному випадку – .

Б3.5 Копіювання й вставка об'єкта структури

Копіювання об'єкта структури в буфер обміну. Щоб помістити копію виділеного об'єкта структури в буфер обміну, потрібно виконати команду  **Копировать (Ctrl+C)**. Якщо копіюється шар, то в буфер обміну поміщується копія його вмісту (всіх дочірніх груп і компонентів).

Звичайна вставка об'єкта структури з буфера обміну. При звичайній вставці створюється новий об'єкт, що є копією об'єкта, поміщеного в буфер обміну. Операція вставки не очищає буфер обміну, тому після однократного копіювання об'єкта можлива його множинна вставка. Для звичайної вставки потрібно виділити групу (шар), у якій може бути створений об'єкт того ж виду, що й об'єкт, поміщений у буфер обміну, і виконати команду  **Вставити (Ctrl+V)**.

Копіювання й вставка об'єкта структури за допомогою мишки. Для копіювання й вставки об'єкта потрібно перетягнути його мишею в потрібну групу (шар) з утриманням клавіші **CTRL**. Якщо група (шар), на яку вказує курсор у процесі перетягування, може містити об'єкт, що копіюється, курсор приймає вид , у противному випадку – . При виконанні даної операції об'єкт не розміщується в буфері обміну.

Спеціальна вставка об'єкта структури. У навігаторі проекту підтримується перетягування об'єктів мишею з одночасним утриманням клавіш **CTRL** та **SHIFT**. Якщо об'єкт, на який вказує курсор у процесі перетягування, допускає завершення операції, курсор приймає вид , у противному випадку – . Еквівалентом цієї операції є копіювання й спеціальна вставка об'єкта по команді  **Вставити с привязкою**.

За допомогою даної операції виконуються наступні дії:

- ▶ при перетягуванні каналу (групи каналів) у шар (групу) – автоматична побудова каналу (групи каналів) з настроєною властивістю **св'язь** (кожен канал, створений таким чином, пов'язаний з відповідним вихідним каналом і має той же клас);

- ▶ при перетягуванні джерела/приймача (групи **Источники/Приемники**) у шар (групу) – автоматична побудова каналу (групи каналів) відповідного класу з настроєною властивістю **св'язь** (кожен канал, створений таким чином, пов'язаний з відповідним джерелом/приймачем);

▶ при перетаскуванні шаблона (групи шаблонів) у шар (групу) – автоматична побудова каналу (групи каналів) класу CALL (кожен канал, створений таким чином, настроєний на виклик відповідного шаблона та має таке ж ім'я, що й шаблон);

▶ при перетаскуванні джерела/приймача або шаблона на компонент – настроювання компонента на зв'язок із джерелом/приймачем або на виклик шаблона. В останньому випадку канал класу CALL приймає ім'я шаблона.

Автоматичний вибір виду операції вставки. При переміщенні об'єкта структури звичайним методом drag-and-drop (без утримання службових клавіш) вид операції вставки (звичайна або спеціальна) вибирається автоматично (ідентифікується за формою курсору).

Б3.6 Редактори каналів

Для кожного класу каналу в ІС існує вбудований редактор. На рис. Б3.4 показаний редактор каналу класу **FLOAT**.

Имя: message Кодировка: ABC Справка

Комментарий: comment

Границы

Использовать

ВП: 90

ВА: 80

ВГ: 60

НГ: 40

НА: 20

НП: 10

Гистерезис: 0

Контроль границ

Обработка

Использовать

Апертура: 0

Пик: 0

Сглаж.: 0

Множитель: 1.25

Смещение: 0

Масштабирование

Масштабирование

A	Смещение	Множитель	Q
Max 200	+ 0	× 1.25	Max 250
Min 100			Min 125

Рассчитать

Системные

Основные

Тип: Output

Размерность:

Период: 1 Единица измерения: цикл

Автопосылка

Enable

Index:

Отработать

На старте: 0

Архивация

Дополнительно

Рисунок БЗ.4 – Зовнішній вигляд редактора каналу класу FLOAT

Два розділи є загальними для всіх редакторів каналів – верхній з кнопкою виклику контекстної довідки (контекстна довідка викликається також по натисканню функціональної клавіші **F1**) і полів для завдання імені, коментаря та кодування каналу, і розділ **Системные**, що вміщує вкладки **Основные**, **Архивация** та **Дополнительно**.

Інші розділи редакторів містять інструменти завдання атрибутів, специфічних для каналів відповідних класів. Редактори каналів містять ту ж панель інструментів для роботи з буфером обміну, що й редактор вузла.

Б4 НАЛАГОДЖЕННЯ ПРОЕКТУ В ІНТЕГРОВАНОМУ СЕРЕДОВИЩІ

Для налагодження проекту ІС має наступні механізми:

- ▶ автономне налагодження шаблонів;
- ▶ запуск виділеного вузла (шару **Технологія**) під керуванням одного з моніторів (профайлерів) з інтегрованого середовища по команді  **Отладка** із протоколюванням роботи в текстовий файл;
 - ▶ використання функції **шпион** – ця функція забезпечує одержання в режимі редагування реальних даних із працюючих вузлів проекту. Для її використання потрібно виконати команду  **Шпион**.

Б4.1 Профайлери

Для налагодження вузла (шару **Технологія**) його можна запустити (у тому числі з ІС) під керуванням одного з наступних налагоджувальних моніторів:

- ▶ профайлера з підтримкою графічних екранів (**rtc.exe**);
- ▶ профайлера без підтримки графічних екранів (**rtmg32.exe**).

При конфігуруванні ІС можна вказати профайлер, що має запускатися з ІС. Для профайлера без підтримки графіки можна задати режим відображення каналів розподіленими за внутрішніми стандартними групами (об'єктами). Профайлери записують протокол своєї роботи у файл **<ім'я файлу rgj>_<порядковий номер вузла>.txt**, що зберігається в папці вузла. Ступінь деталізації налагоджувальної інформації, виведеної у файл, може бути задана.

Б4.2 Профайлер з підтримкою графічних екранів

Графічна оболонка профайлера з підтримкою графічних екранів містить меню, панель інструментів і робоче поле, як це показано на рис. Б4.1.

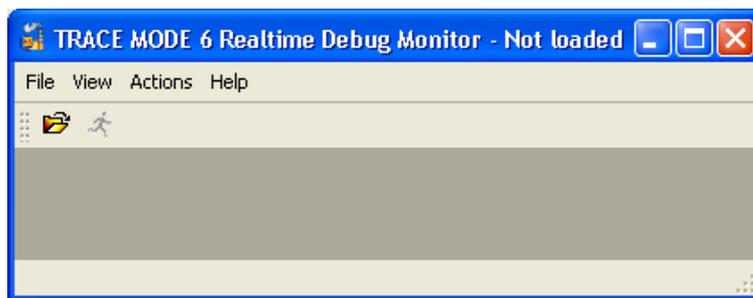


Рисунок Б4.1 – Зовнішній вигляд профайлера з підтримкою графічних екранів

Меню **File – Файл** і панель інструментів містять команди відкриття (📁, CTRL+O), перезавантаження й запуску (🏃, CTRL+R) вузла, а також команду виходу із програми.

Меню **Actions – Операції** містить команди авторизації/закінчення сеансу, а також команду посилки повідомлення у звіт тривоги.

Меню **View – Вид** містить наступні команди:

- ▶ **Полный экран (CTRL+F)** – перемикає вид відображення графічних екранів (у вікні/повноекранний);

- ▶ **Компоненты (CTRL+0)** – по цій команді на екрані з'являється діалог, у лівій частині якого відображаються канали вузла, а праворуч містяться інструменти завдання атрибутів каналу, виділеного в лівій частині, що продемонстровано на рис. Б4.2.

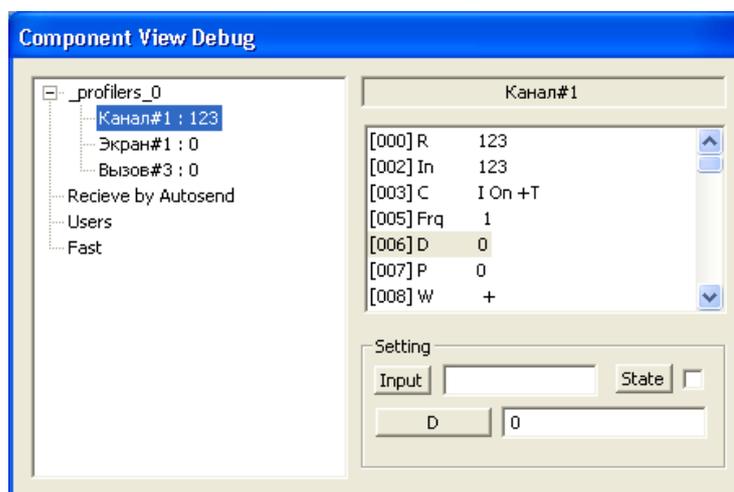


Рисунок Б4.2 – Зовнішній вигляд діалогового вікна **Компоненты**

Для завдання вхідного значення каналу (атрибута 0, **In**) потрібно ввести необхідне значення в поле праворуч від кнопки **Input** і нажати ЛК на цій кнопці. Для завдання значення довільного атрибута потрібно виділити атрибут у списку, ввести необхідне значення в поле праворуч від кнопки, на яку виводиться коротке ім'я обраного атрибута, і нажати ЛК на цій кнопці. Для вимикання каналу потрібно встановити прапорець у поле праворуч від кнопки **State** і нажати ЛК на цій кнопці.

Б4.3 Профайлер без підтримки графічних екранів

Графічна оболонка профайлера без підтримки графічних екранів містить меню, панель інструментів і робоче поле, у якому відображаються канали вузла, як це показано на рис. Б4.3.

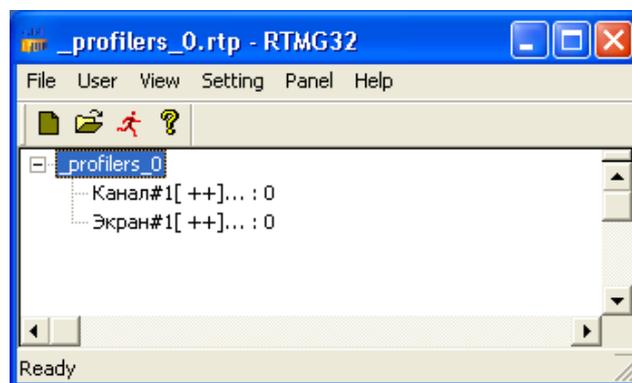


Рисунок Б4.3 – Зовнішній вигляд профайлера без підтримки графічних екранів

Меню **File** – **Файл** і панель інструментів містять команди відкриття (, CTRL+O), перезавантаження (, CTRL+R) і запуску  вузла, а також команду виходу із програми.

Меню **User** – **Пользователь** містить команди авторизації/закінчення сеансу, а також команду **Пользователи**.

Меню **View** – **Вид** містить прапорці керування видимістю рядка статусу,

панелі інструментів і поділу вікна, що продемонстровано на рис. Б4.4.

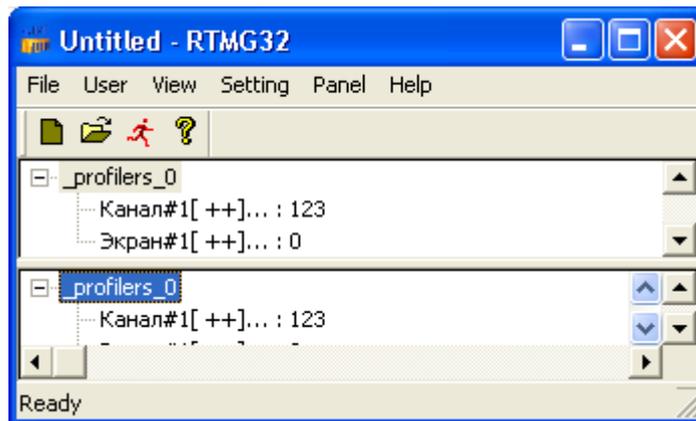


Рисунок Б4.4 – Зовнішній вигляд розділеного вікна профайлера

Б5 ПРОГРАМУВАННЯ АЛГОРИТМІВ

Для програмування алгоритмів функціонування проекту АСУ є мови **Техно ST**, **Техно SFC**, **Техно FBD**, **Техно LD** і **Техно IL**. Дані мови є модифікаціями мов **ST** (Structured Text), **SFC** (Sequential Function Chart), **FBD** (Function Block Diagram), **LD** (Ladder Diagram) і **IL** (Instruction List) стандарту IEC61131-3.

Для створення й редагування властивостей аргументів, змінних, функцій і структурних типів програми, а також для використання в програмі функцій із зовнішніх бібліотек в інтегроване середовище розробки проекту вбудовані спеціальні табличні редактори. Trace mode 6 має також засоби для налагодження програм. Показовий вигляд інтегрованого середовища при редагуванні програм показаний на рис. Б5.1

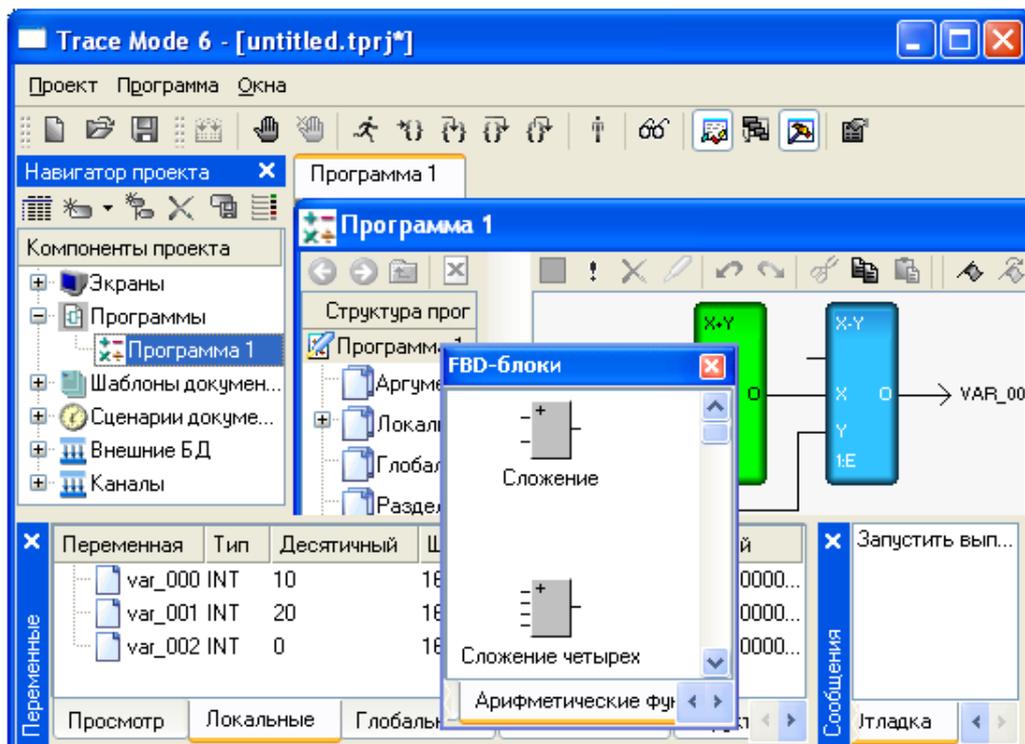


Рисунок Б5.1 – Зовнішній вигляд інтегрованого середовища при редагуванні програм

Основною мовою програмування є **Техно ST**. Програми, розроблені на мовах **Техно LD**, **Техно SFC** і **Техно FBD**, перед компіляцією транслюються в **Техно ST**. **PL**-програми перед компіляцією частково транслюються в **ST**, частково – в асемблер. Таким чином, ключові слова **Техно ST** є такими ж самими й для всіх інших мов.

Б5.1 Підключення програми до проекту

Для підключення програми до проекту її потрібно спочатку скомпілювати, а потім зберегти проект. Щоб скомпілювати програму, потрібно виконати команду **Компілювати** з меню **Програма**, або натиснути клавішу **F7** чи ЛК на іконці  панелі інструментів відладчика. Перед збереженням проекту потрібно переконатися, що компіляція пройшла успішно (у вікні повідомлень компілятора в цьому випадку виводиться відповідне повідомлення). Налаштування програми можливе тільки після її успішної компіляції.

Б5.2 Вибір мови програмування

Мова програмування може бути незалежно задана для основної програми, функції-блоку, функції й кроку SFC. Мова програмування вибирається в діалоговому вікні, представленою на рис. Б5.2.

Цей діалог автоматично з'являється на екрані при натисканні ЛК на імені знову створеної програми або її компонента (для якого мова може бути задана незалежно) у вікні структури програми. Після вибору мови програма (компонент) відкривається у відповідному редакторі.

Змінити мову можна тільки після видалення тіла програми (компонента). Для цього потрібно натиснути ЛК на іконці  панелі інструментів у вікні

структури програми, після чого автоматично з'являється діалог вибору мови.

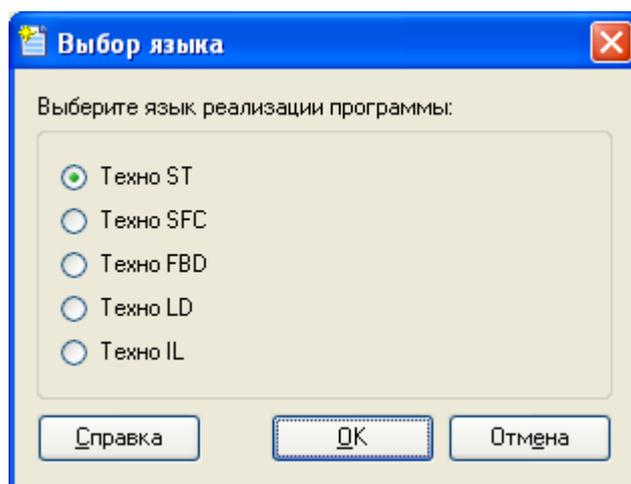


Рисунок Б5.2 – Зовнішній вигляд діалогового вікна з вибору мови програмування

Б5.3 Створення елементів програм за допомогою табличних редакторів

Табличні редактори використовуються для створення наступних компонентів й елементів програм:

- ▶ аргументи;
- ▶ локальні змінні;
- ▶ глобальні змінні;
- ▶ функції-блоки (підпрограми) і функції;
- ▶ структурні типи.

Крім того, за допомогою табличних редакторів конфігуруються звернення до функцій із зовнішніх бібліотек. Перераховані компоненти й елементи, поряд з лістингами **ST** й **IL** і діаграмами **LD**, **SFC** й **FBD**, створюють гілки дерева у вікні структури програми. Для входу у відповідний табличний редактор потрібно у вікні структури програми натиснути ЛК на кожному з перерахованих вище елементів.

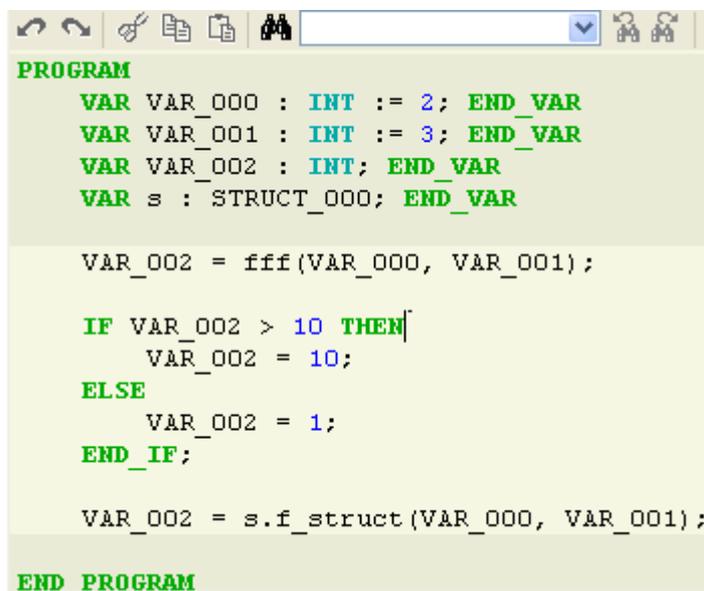
Особливості редагування. Для створення/видалення рядків і пошуку в табличних редакторах використовується типова панель інструментів. Для

переходу до редагування окремої комірки таблиці потрібно двічі нажати ЛК на ній. Редагування комірки здійснюється шляхом безпосереднього введення з клавіатури або вибору потрібного значення зі списку.

В якості роздільника цілої й дробової частини чисел використовується крапка.

Якщо в комірку стовпця [] **Массив** ввести число, рівне кількості елементів масиву, то в цій комірці відобразиться діапазон індексів елементів (починаючи з 0). Наприклад, для двовимірного масиву при уведенні 9, 8 відобразиться 0 .. 8, 0 .. 7.

Деякі елементи (наприклад, змінні), задані в табличних редакторах, автоматично додаються в лістинги текстових програм у вигляді відповідних конструкцій мови. Ці конструкції виділяються сірим кольором, як показано на рис. Б5.3. Вони недоступні для безпосереднього редагування за допомогою клавіатури.



```
PROGRAM
  VAR VAR_000 : INT := 2; END_VAR
  VAR VAR_001 : INT := 3; END_VAR
  VAR VAR_002 : INT; END_VAR
  VAR s : STRUCT_000; END_VAR

  VAR_002 = fff(VAR_000, VAR_001);

  IF VAR_002 > 10 THEN
    VAR_002 = 10;
  ELSE
    VAR_002 = 1;
  END_IF;

  VAR_002 = s.f_struct(VAR_000, VAR_001);

END PROGRAM
```

Рисунок Б5.3 – Щодо принципу виділення конструкцій мови

Доступні типи даних (стовпець **Тип данных**) для програм на всіх мовах однакові.

Початкове значення (стовпець **Начальное значение**) може бути задане в

кожній з форм, визначених для **Техно ST**.

Табличний редактор аргументів програмного компонента. Вид табличного редактора аргументів програмного компонента (функції або функції-блоку) показано на рис. Б5.4. В цьому редакторі задається ім'я аргументу, його тип (**ВХОД**, **ВЫХОД** або **ВХОД/ВЫХОД**), тип даних, початкове значення й коментар. Якщо в поле **Массив** рядка аргументу задати число, аргумент інтерпретується як масив.

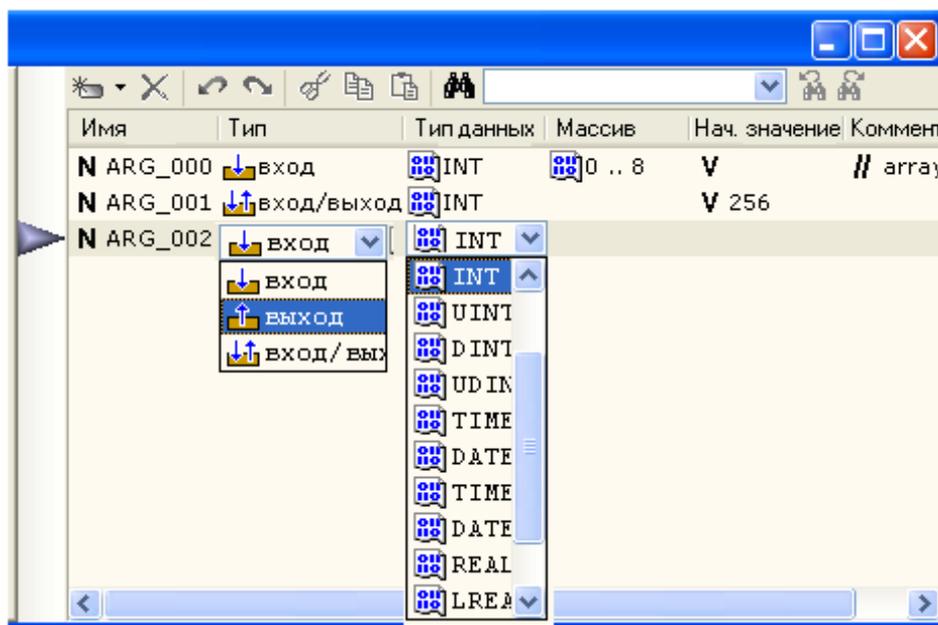


Рисунок Б5.4 – Зовнішній вигляд табличного редактора аргументів програмного компонента

Табличний редактор змінних. Вид табличного редактора змінних показано на рис. Б5.5. У цьому редакторі задається ім'я змінної, її тип даних, початкове значення й коментар. Якщо в поле **Массив** рядка змінної задати число, змінна інтерпретується як масив.

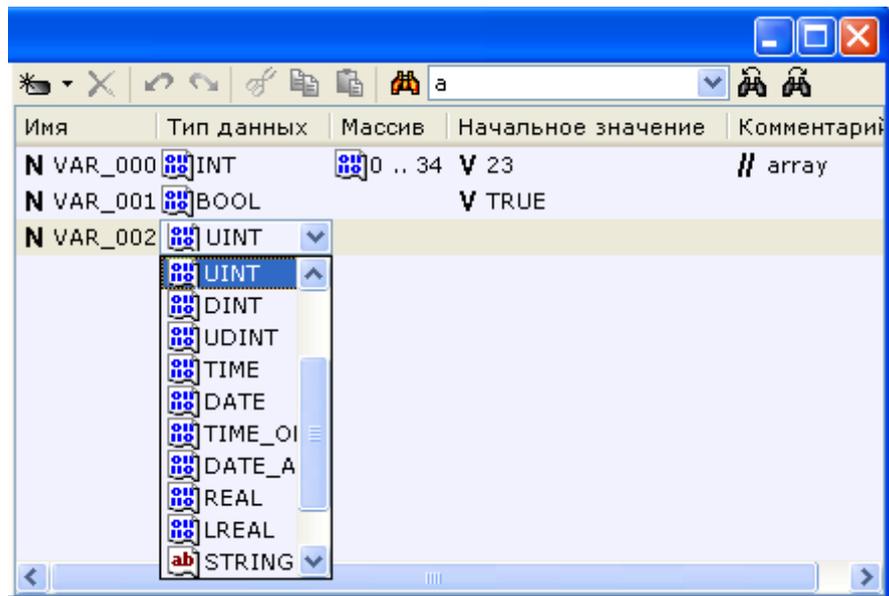


Рисунок Б5.5 – Зовнішній вигляд табличного редактора змінних

Табличний редактор функцій і функцій-блоків. Вид табличного редактора функцій і функцій-блоків показано на рис. Б5.6. У цьому редакторі задається ім'я функції (функції-блоку) і коментар. Якщо в поле **Массив** рядка функції задати число, функція повертає масив. Для функції-блоку поле **Массив** недоступно.

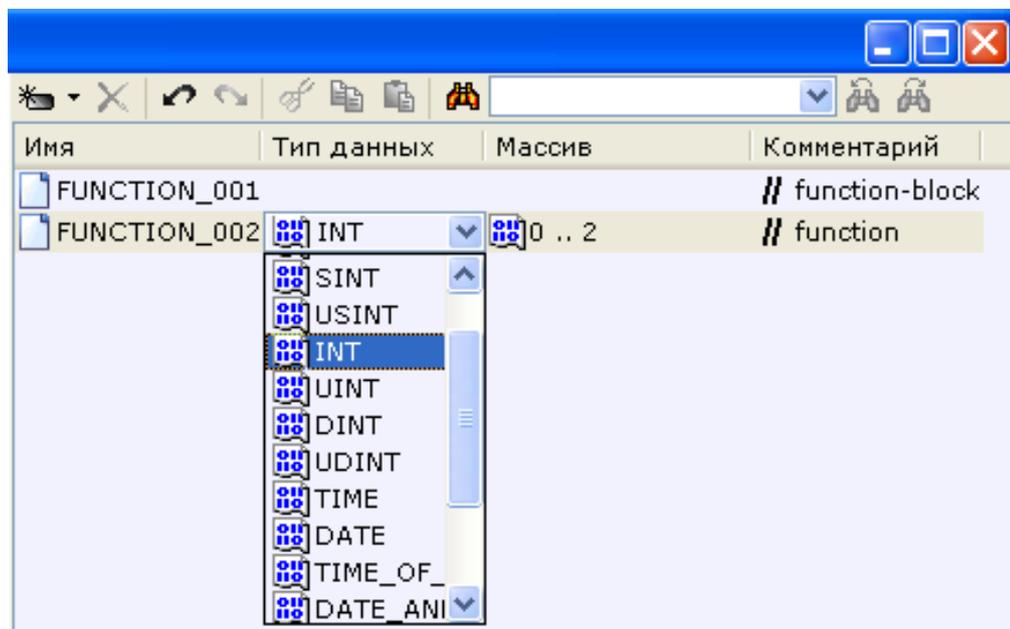


Рисунок Б5.6 – Зовнішній вигляд табличного редактора функцій і функцій-блоків

Табличний редактор структурних типів. В редакторі структурних типів, зовнішній вигляд якого показано на рис. Б5.7, задається ім'я створюваного структурного типу й коментар. Більш докладний опис алгоритмів програмування, а також опис мов програмування можна знайти в довідковій системі Trace Mode у розділі **Программирование алгоритмов.**

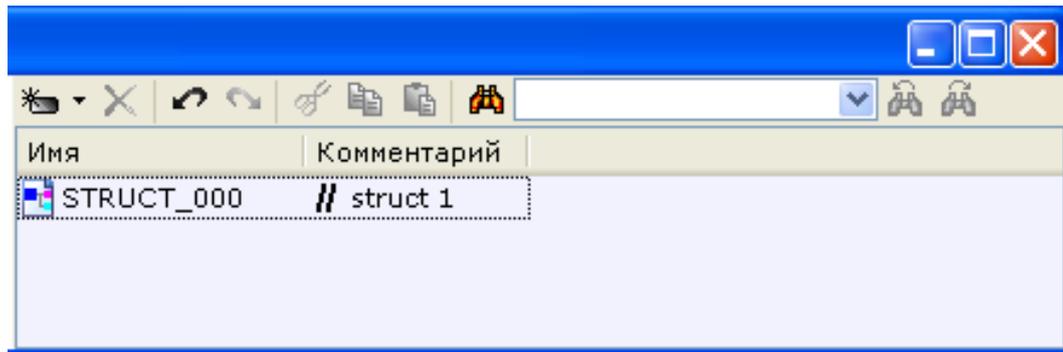


Рисунок Б5.7 – Зовнішній вигляд табличного редактора структурних типів

Б6 РОЗРОБКА ГРАФІЧНОГО ІНТЕРФЕЙСУ

Б6.1 Редактор подання даних

Графічне подання ходу виконання технологічного процесу, а також керування технологічним процесом за допомогою графічних засобів є одним з головних завдань Trace Mode. Для розробки інтерфейса оператора в інтегроване середовище вбудований **редактор представлення даних (РПД)**, інтерфейс якого зображено на рис. Б6.1.

Інтерфейс оператора розробляється у вигляді набору **графических экранов**, що є компонентами проекту. З метою взаємодії з іншими компонентами проекту для графічного екрана можуть бути задані аргументи. Сукупність графічних екранів вузла створюють його **графическую базу**. Сукупність графічних баз всіх вузлів проекту АСУТП утворюють **графическую часть** проекту.

Графічний екран може містити один або кілька графічних **слоев** (шарів), кожний з яких може містити один або кілька **подслоев**. У графічних **слоях** розміщуються **Графические элементы (ГЭ)**. Графічні елементи (ГЕ) мають набори **атрибутов, динамических свойств та функций управления**. Ці параметри визначають вид графічних елементів і виконувані ними функції відображення й керування при роботі в реальному часі. Редактор подання даних містить велику кількість вбудованих графічних елементів, що дозволяє зобразити практично будь-який технологічний процес, вивести на дисплей всю необхідну інформацію про хід його виконання, а також управляти технологічним процесом.

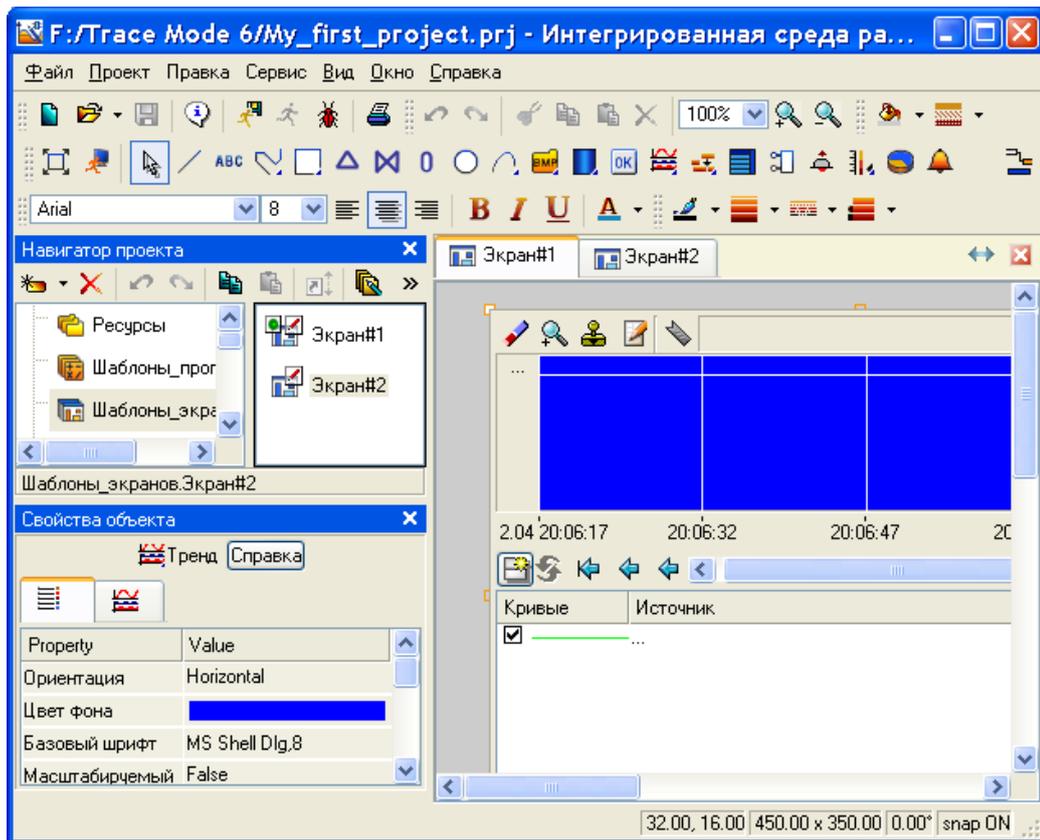


Рисунок Б6.1 – Интерфейс редактора подання даних

Б6.2 Режими роботи редактора подання даних

Редактор подання даних може перебувати в одному з наступних режимів:

- ▶ режим **размещения** призначений для заповнення графічних шарів екранів графічними елементами. Для переходу в цей режим потрібно натиснути одну із кнопок вибору ГЕ на панелі інструментів **Графические элементы**;
- ▶ режим **редактирования** призначений для внесення змін у створені раніше графічні екрани (наприклад, для видалення/додавання графічних елементів або зміни їхніх властивостей). Для переходу в цей режим треба натиснути кнопку  головної панелі інструментів;
- ▶ режим **эмуляции** використовується для перевірки роботи графічних елементів у реальному часі. Для переходу в режим емуляції треба натиснути кнопку  панелі інструментів **Графические элементы**. Для виходу з режиму

треба натиснути кнопку  повторно.

Крім того, у РПД передбачено два режими відображення графічних екранів – звичайний (у вікні) і повноекранний. Для перемикання режимів відображення використовується кнопка  панелі інструментів **Графические элементы**.

У режимах розміщення й редагування поточні координати курсору відображаються в рядку статусу (унизу праворуч). Там же відображається стан прапорця **Располагать по сетке**.

Б6.3 Головне меню й панель інструментів редактора подання даних

Панель інструментів **Графические элементы**, зовнішній вигляд якої зображено на рис. Б6.2, дозволяє вибирати графічні елементи для розміщення їх у графічних шарах екранів. При виборі ГЕ редактор переходить у режим розміщення.



Рисунок Б6.2 – Зовнішній вигляд панелі інструментів **Графические элементы**

За допомогою кнопки  даної панелі можна перейти в режим редагування, за допомогою кнопки  – у режим емуляції. Кнопка  призначена для перемикання режимів відображення графічних екранів (звичайний/повноекранний).

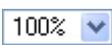
Меню й панель інструментів **Правка**, зовнішній вигляд якої зображено на рис. Б6.3, містять ряд типових інструментів для редагування графічних екранів. Дані інструменти доступні також з контекстного меню ГЕ. У списку  (Масштаб) або за допомогою кнопок  і  панелі інструментів **Правка** вибирається масштаб відображення.



Рисунок Б6.3 – Зовнішній вигляд панелі інструментів **Правка**

Меню **Сервис** і панель інструментів **Топология экрана**, зовнішній вигляд якої зображено на рис. Б6.4, містять команди для позиціювання й тиражування виділеного графічного елемента. Меню **Сервис** містить додатково команду **Параметры экрана**.



Рисунок Б6.4 – Зовнішній вигляд панелі інструментів **Топология экрана**

Панель інструментів **Параметры текста** має вигляд, зображений на рис. Б6.5. У режимі редагування за допомогою типових інструментів даної панелі задаються параметри тексту у виділеному графічному елементі (виділеній групі ГЕ). Дані команди застосовні тільки до такого тексту, що може бути введений/відредагований за допомогою клавіатури. Вид ГЕ при його розміщенні в графічному шарі залежить від параметрів, установлених за допомогою інструментів цієї панелі.



Рисунок Б6.5 – Зовнішній вигляд панелі інструментів **Параметры текста**

Панель інструментів **Параметры линии** має зовнішній вигляд, що зображено на рис. Б6.6.



Рисунок Б6.6 – Зовнішній вигляд панелі інструментів **Параметры линии**

У режимі редагування за допомогою інструментів цієї панелі задаються

параметри лінії (лінії контуру) виділеного графічного елемента (виділеної групи ГЕ):



– вибір **цвета лінії**. По цій команді на екран виводиться стандартний діалог вибору кольору;



– вибір **толщини лінії**;



– вибір **стиля лінії**. По цій команді відкривається список стилів, що містить у тому числі опцію **Без лінії**;



– вибір **края лінії** (плоский, квадратний, круглий).

Вид ГЕ при його розміщенні в графічному шарі залежить від параметрів, установлених за допомогою інструментів цієї панелі.

Панель інструментів **Параметры заливки** має зовнішній вигляд, що зображено на рис. Б6.7.



Рисунок Б6.7 – Зовнішній вигляд панелі інструментів **Параметры заливки**

У режимі редагування за допомогою інструментів цієї панелі задаються параметри заливання виділеного графічного елемента (виділеної групи ГЕ):



– вибір **цвета заливки**. По цій команді на екран виводиться стандартний діалог вибору кольору;



– вибір **стиля заливки**.

Вид ГЕ при його розміщенні в графічному шарі залежить від параметрів, установлених за допомогою інструментів цієї панелі.

Панель інструментів **Ресурсные библиотеки** має зовнішній вигляд, що зображено на рис. Б6.8. Інструменти даної панелі призначені для операцій з бібліотеками рядків, рисунків та інших ресурсів, які можуть бути використані при розробці графічних екранів.



Рисунок Б6.8 – Зовнішній вигляд панелі інструментів **Ресурсные библиотеки**

Команди меню **Вид** керують видимістю табличного редактора аргументів екрана, вікна **Слои** й таблиці графічних елементів, а також панелей інструментів **Топология экрана** та **Параметры текста**.

Б6.4 Завдання параметрів редактора подання даних

Вікно **Настройки** з меню **Файл** містить діалог завдання параметрів РПД, як це показано на рис. Б6.9. Цей діалог містить наступні інструменти:

- ▶ **Подсвечивать при наведении мыши** – якщо цей прапорець установлений, то при наведенні курсору миші на ГЕ його вершини (вузлові крапки) виділяються червоними кольорами. Не слід плутати цю функцію з функцією виділення ГЕ.

- ▶ **Открывать свойства автоматически** – від цього прапорця залежить режим РПД після розміщення графічного елемента на екрані;

- ▶ **Располагать по сетке** – якщо цей прапорець установлений, то при розміщенні, переміщенні й масштабуванні вершини прямокутника, що обмежують ГЕ, розташовуються у вузлах сітки. При розміщенні у вузлах сітки розташовуються також вузлові крапки ГЕ.

- ▶ **Показать сетку** – якщо цей прапорець установлений, сітка відображається на графічних екранах;

- ▶ **Шаг сетки** – завдання кроку сітки в пікселях (1 – 100);

- ▶ **Цвет сетки** – вибір кольору сітки;

- ▶ **Цвет контура выделения** – вибір кольору прямокутника, що обмежує ГЕ при виділенні.

- ▶ **Использовать устаревшие функции** – при установці цього прапорця будуть доступні деякі опції попередніх версій Trace Mode.

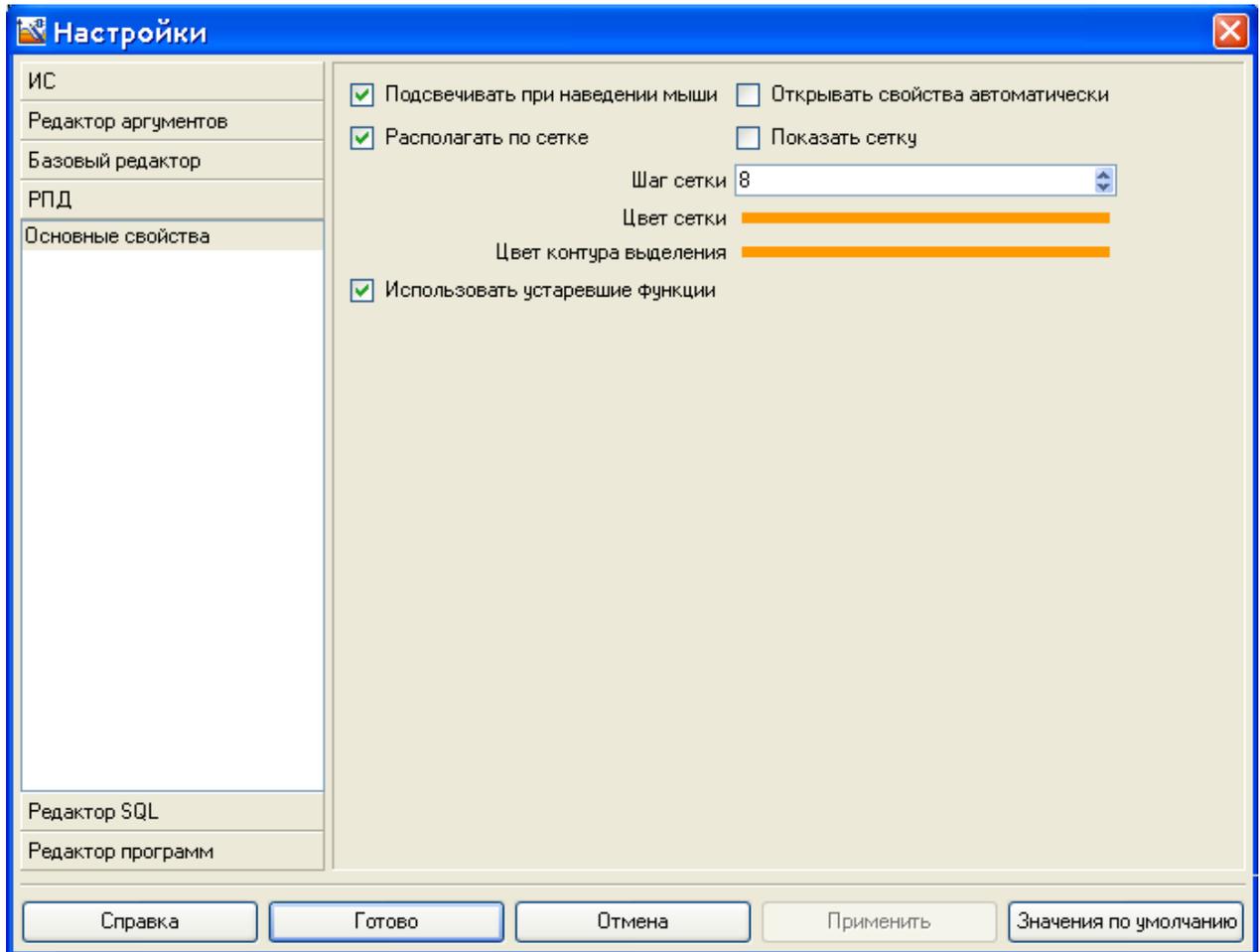


Рисунок Б6.9 – Зовнішній вигляд діалогового вікна **Настройки**

Б6.5 Завдання параметрів графічного екрана

Параметри графічного екрана, що редагується, задаються в діалозі, який відкривається при виконанні команди **Параметры экрана** меню **Сервис**, як це зображено на рис. Б6.10.

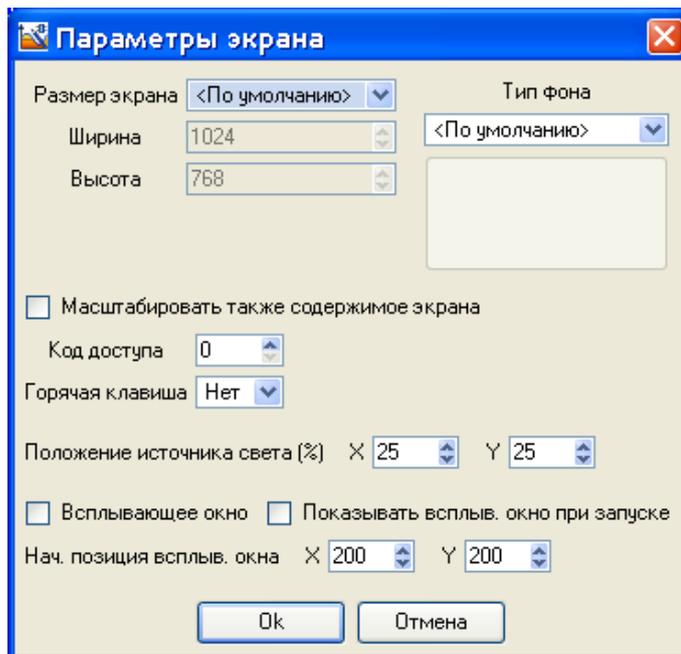


Рисунок Б6.10 – Зовнішній вигляд вікна для завдання параметрів графічного екрана

Цей діалог містить наступні інструменти:

► **Размер экрана** – завдання розміру екрана в пікселях. Розмір можна вибрати з декількох стандартних або задати свій за допомогою опції **Произвольный** і полів **Ширина** й **Высота**.

► **Тип фона** – вибір типу фона. Містить наступні варіанти:

► **По умолчанию** – фон екрана за згодою;

► **Цвет** – при виборі цієї опції для екрана можна задати кольори фона. Для цього потрібно натиснути кнопку під списком і вибрати кольори в стандартному діалозі;

► **Изображение** – при виборі цієї опції в якості фона екрана можна використати малюнок;

► **Масштабировать также содержимое экрана** – прапорець, при установці якого розміщені на екрані ГЕ масштабуються пропорційно зміні розмірів екрана;

► **Код доступа** – код доступу до екрана (0 – 255). Права на доступ до екранів задаються для користувача у вигляді маски в розділі **Доступ/Экраны**

каналу **Пользователь**. При кореляції маски з кодом доступу (результат побітового логічного множення відмінний від нуля) дозволений доступ до екрана, у протилежному випадку – заборонений;

- ▶ **Горячая клавиша** – меню вибору функціональної гарячої клавіші (**F2 – F12**). При запуску проекту в реальному часі натискання заданої клавіші буде супроводжуватися переходом на даний екран;

- ▶ **Положение источника света (%)** – положення джерела світла щодо екрана (кут з осями X й Y у відсотках). Значення (50, 50) відповідає розташуванню джерела світла на нормалі до екрана;

- ▶ **Всплывающее окно** – показувати екран у списку спливаючих екранів МРВ;

- ▶ **Показывать всплыв. окно при запуске** – показувати екран при завантаженні вузла в МРВ;

- ▶ **Нач. позиция всплыв. окна X и Y** – положення спливаючого екрана по осях X й Y при завантаженні вузла в МРВ. Дана опція не впливає на відображення звичайного екрана.

Частина параметрів для створюваних графічних екранів можна задати в редакторі групи шаблонів екранів.

Б6.6 Завдання аргументів графічного екрана

Для графічного екрана можуть бути задані аргументи з метою взаємодії з іншими компонентами проекту. Щоб відкрити табличний редактор аргументів екрана, потрібно вибрати опцію [Аргументы экрана](#) у меню **Вид**.

Б6.7 Операції із графічними елементами

Б6.7.1 Розміщення графічних елементів

Вбудовані графічні елементи розбиті на групи. Кожній групі відповідає

кнопка на панелі інструментів **Графические элементы**. На кнопку виводиться іконка одного з елементів даної групи. Щоб вибрати ГЕ для розміщення, потрібно виконати наступні дії:

- ▶ нажати ЛК на кнопці панелі інструментів **Графические элементы**. При цьому вибирається той елемент, іконка якого виведена на кнопку (елемент, заданий за згодою для відповідної групи, або елемент, обраний раніше);

- ▶ двічі нажати ЛК на кнопці й потім нажати ЛК на іконці необхідного ГЕ в меню, що з'явилося (меню не відкривається, якщо група містить тільки один графічний елемент).

Після вибору елемента його іконка переходить на кнопку групи. Наприклад, на рис. Б6.11 показано меню групи **Прямоугольники**.



Рисунок Б6.11 – Вибір групи **Прямоугольники**

При виборі графічного елемента редактор подання даних переходить у режим розміщення, при цьому курсор на графічному екрані здобуває вид **+**.

Потім у вікні **Слои** необхідно натисканням ЛК указати шар, у якому повинен бути розміщений обраний графічний елемент.

Далі продовжити процедуру розміщення ГЕ можна двома способами:

- ▶ перетягнути ГЕ з панелі інструментів на екран (метод drag-and-drop). Після розміщення ГЕ має розміри, задані за згодою, РПД переходить у режим редагування, а вікно властивостей ГЕ відкривається автоматично;

- ▶ перемістити курсор у потрібну точку екрана й натисканням ЛК установити **точку привязки** ГЕ. Далі дії по розміщенню ГЕ можуть відрізнитися, однак для більшості графічних елементів вони стандартні – переміщення миші після установки точки прив'язки виводить на екран образ ГЕ, при цьому відрізок від точки прив'язки до поточного положення курсору є діагоналлю прямокутника, що обмежує ГЕ. (Якщо при переміщенні миші

втримувати натиснутої клавішу **CTRL**, ряд ГЕ виявиться вписаним у квадрат). Повторне натискання ЛК приводить до розміщення графічного елемента в обраному графічному шарі.

Для графічних елементів груп **Ломаные** й **Кривые** кожне натискання ЛК після установки точки прив'язки задає **узловую точку** (проміжну вершину). Для установки останньої вершини й виходу з режиму розміщення цих ГЕ потрібно натиснути ПК (права кнопка мишки). Положення вузлових крапок, задане при розміщенні, надалі можна змінити.

Режим РПД після розміщення ГЕ даним способом залежить від прапора **Открывать свойства автоматически:**

- ▶ якщо прапорець установлений, то після розміщення ГЕ автоматично відкривається вікно його властивостей, а РПД переходить у режим редагування;
- ▶ якщо прапорець не встановлений, то після розміщення ГЕ РПД залишається в режимі розміщення. Цей спосіб зручний для багаторазового розміщення на екрані того самого графічного елемента.

Б6.7.2 Переміщення й масштабування графічних елементів

Для переміщення або зміни розмірів виділеного ГЕ або групи ГЕ потрібно вибрати в контекстному меню графічного елемента режим **Перемещать/масштабировать** й далі використати стандартні операції редагування.

Б6.7.3 Видалення графічних елементів

Для видалення виділеного ГЕ (групи ГЕ) потрібно натиснути клавішу **Del** на клавіатурі або виконати команду **Удалить** за допомогою меню або панелі інструментів **Правка** чи контекстного меню ГЕ. У повноекранному режимі редагування для видалення виділеного ГЕ (групи ГЕ) за допомогою клавіатури використовується комбінація клавіш **Ctrl+Del**.

Б6.7.4 Копіювання й вставка графічних елементів

Для копіювання виділеного ГЕ (групи ГЕ) у буфер обміну потрібно виконати команду **Копировать**. Для вставки вмісту буфера обміну в шар потрібно виконати команду **Вставить**.

Скопіювати в буфер обміну можна в тому числі групу графічних елементів, що лежать у різних шарах екрана, однак при вставці такої групи всі її ГЕ будуть розміщені в тому самому шарі.

Виконати команди **Копировать** й **Вставить** можна за допомогою меню або панелі інструментів **Правка** чи за допомогою контекстного меню ГЕ.

Б6.7.5 Поворот графічних елементів

Існує два способи повороту ГЕ на екрані в режимі редагування: через режим **Повернуть** або за допомогою таблиці графічних елементів.

Для переходу в режим **Повернуть** потрібно вибрати в контекстному меню виділеного ГЕ (групи ГЕ) опцію **Повернуть**. Потім встановлюється курсор в одну з вершин прямокутника, що обмежує ГЕ (групу ГЕ) (курсор при цьому приймає вид , як це показано на рис. Б6.12).

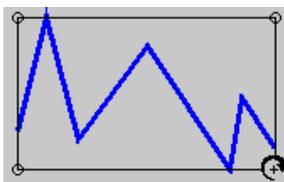


Рисунок Б6.12 – Зовнішній вигляд курсору при виборі опції **Повернуть**

Потрібно натиснути ЛК й, утримуючи кнопку натиснутою, переміщенням мишки задати потрібний кут повороту ГЕ. Для виходу з режиму треба відпустити ЛК. При використанні даного методу ГЕ (група ГЕ) обертається відносно центра

обмежуючого прямокутника (центром прямокутника є точка перетинання його діагоналей).

Кут повороту ГЕ з таблиці графічних елементів можна задати в поле **Угол (градусы)** діалогу **Геометрия**. При використанні цього методу ГЕ повертається щодо точки прив'язки (при завданні додатного значення – за годинниковою стрілкою).

Б6.7.6. Тиражування графічних елементів

Тиражування – це копіювання виділеного графічного елемента і його множинна вставка з табличним упорядкуванням. Копії вставляються вправо й нагору/униз відносно виділеного ГЕ. Тиражування виділеної групи ГЕ не підтримується.

Операція тиражування конфігурується в діалозі, що відкривається при виконанні команди **Тиражировать** з меню **Сервис** або натисканні кнопки  на панелі інструментів **Топология экрана**, як це показано на рис. Б6.13. У цьому діалозі задаються наступні параметри:

- ▶ **Число столбцов** – число елементів у рядку (включаючи виділений спочатку ГЕ);
- ▶ **Число строк**;
- ▶ **Промежуток по горизонтали** – проміжок між копіями по горизонталі в пікселях;
- ▶ **Промежуток по вертикали** – проміжок між копіями по вертикалі в пікселях;
- ▶ **Вверх** – розмноження по рядках нагору відносно виділеного ГЕ;
- ▶ **Вниз** – розмноження по рядках униз відносно виділеного ГЕ.

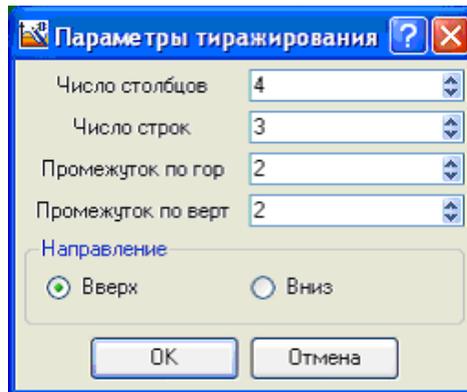


Рисунок Б6.13 – Меню для завдання параметрів тиражування

На рис. Б6.14 показаний результат тиражування ГЕ **Кнопка** (виділена на рисунку) при наступних параметрах: число стовпців – 4, число рядків – 3, проміжок по горизонталі й вертикалі – 5, напрямок тиражування – униз.



Рисунок Б6.14 – Результат тиражування ГЕ **Кнопка**

Б6.8 Завдання типових властивостей графічних елементів

Графічні елементи мають наступні властивості, що настраюються:

- ▶ **Атрибути;**
- ▶ **Динамические свойства;**
- ▶ **Функции управления.**

Ці параметри визначають вид графічних елементів і виконувані ними функції відображення й керування при роботі в реальному часі. Для завдання властивостей ГЕ (групи ГЕ) використовується вікно **Свойства объекта**, що містить різне число вкладок для різних елементів, як це показано на рис. Б6.15.

Щоб відкрити це вікно, потрібно виконати команду **Свойства** з контекстного меню виділеного ГЕ (виділеної групи ГЕ) або двічі нажати ЛК на ГЕ. При знятті виділення ГЕ вікно його властивостей автоматично

закривається. Вікно **Свойства объекта** недоступно, якщо графічний елемент розташований у шарі, редагування якого заборонене.

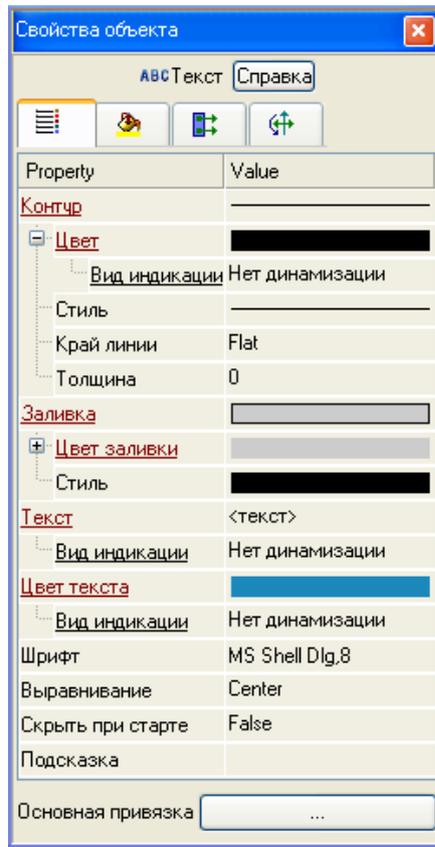


Рисунок Б6.15 – Зовнішній вигляд вікна властивостей об'єкта

Атрибути – це найпростіші властивості графічного елемента. Вони задаються на вкладці  – **Основные свойства** вікна **Свойства объекта**. У вікні властивостей атрибути можуть бути згруповані – найменування таких груп виділені підкресленням, при подвійному натисканні на них ЛК розкривається список властивостей.

Існують 2 види атрибутів ГЕ:

- ▶ **статические** – атрибути, які не змінюються при роботі в реальному часі;
- ▶ **динамизируемые** – атрибути, які можуть бути як статичними, так і динамічними (що змінюються при роботі в реальному часі залежно від значення прив'язаного аргументу). Розділи конфігурування таких атрибутів виділені червоними кольорами й містять пункт **Вид индикации**.

Б6.9 Статичні атрибути графічних елементів

Видимість при старті. При переході в режим реального часу (тобто при запуску проекту в MPV або при переході в режим емуляції) графічні елементи за згодою видимі. Щоб сховати GE при запуску режиму реального часу, потрібно встановити для нього атрибут **Сховати при старті**, як це показано на рис. Б6.16. У режимі редагування видимі всі GE. У реальному часі управляти видимістю GE можна за допомогою функції керування GE й за допомогою GE **Свободные формы**.



Рисунок Б6.16 – Меню для вибору прихованості GE

Спливаюча підказка. У розділі **Підказка** для GE задається спливаюча підказка, що відображається на екрані при наведенні курсору на GE в режимі реального часу, як це показано на рис. Б6.17.



Рисунок Б6.17 – Бокс для вводу спливаючої підказки

Колірні атрибути. Розділи конфігурування колірних атрибутів GE **Цвет**, **Цвет линии**, **Цвет заливки**, **Цвет текста** й т.п. у вікні **Свойства объекта** містять типовий інструмент завдання кольорів – кнопку кольору, що відповідає поточному кольору GE, як це показано на рис. Б6.18.

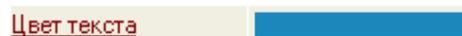


Рисунок Б6.18 – Кнопка для вибору кольору GE

Товщина лінії. Розділ конфігурування атрибута **Товщина** містить інструмент  вибору товщини лінії й вікно відображення чисельного значення товщини (у пікселях), як це показано на рис. Б6.19.

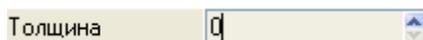


Рисунок Б6.19 – Бокс для завдання товщини лінії

Стиль лінії. Розділ конфігурування атрибута **Стиль лінії** містить кнопку , як це показано на рис. Б6.20.



Рисунок Б6.20 – Бокс для завдання стилю лінії

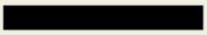
Стиль заливання. Розділ конфігурування атрибута **Стиль заливки** містить кнопку , як це показано на рис. Б6.21.



Рисунок Б6.21 – Бокс для завдання стилю заливання

Текстові атрибути. Розділ конфігурування текстових атрибутів **Текст**, **Надпись** і т.п. містить вікно звичайного текстового редактора для завдання значення атрибута (для деяких ГЕ подібному до атрибута за згодою привласнюється деяке значення), як це показано на рис. Б6.22.



Рисунок Б6.22 – Бокс для вводу тексту

Шрифт. Розділ конфігурування атрибута **Шрифт** містить кнопку, зображену на рис. Б6.23, при натисканні якої на екран виводиться меню вибору параметрів шрифту, як це показано на рис. Б6.24.

Рисунок Б6.23 – Кнопка для відкриття меню вибору параметрів шрифту

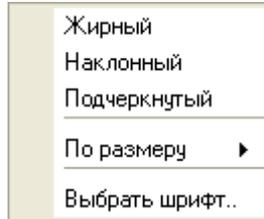


Рисунок Б6.24 – Меню вибору параметрів шрифту

Вирівнювання. Розділ конфігурування атрибута **Выравнивание** містить стандартні інструменти вирівнювання тексту: **Влево**, **По центру** й **Вправо**, як це показано на рис. Б6.25.

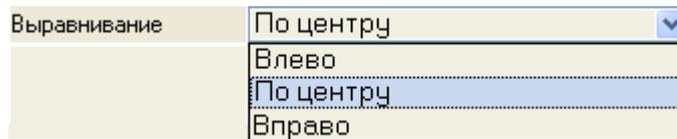


Рисунок Б6.25 – Меню вибору параметрів вирівнювання тексту

Б6.10 Динамізація атрибута графічних елементів

Динамізацією атрибута називається завдання умов його зміни залежно від значення прив'язаного аргументу. При динамізації атрибута графічний елемент стає індикатором виконання заданих умов.

При розміщенні ГЕ на екрані всі його динамізуємі атрибути за згодою статичні, і розділи їхнього конфігурування на вкладці **Основные свойства** вікна властивостей містять інструмент завдання відповідного статичного параметра. Наприклад, при розміщенні ГЕ **Текст** розділ динамізуємого атрибута **Text color** – **Цвет текста** містить інструмент вибору кольору , як це зображено на рис. Б6.26.

Text color

Рисунок Б6.26 – Кнопка для вибору кольору ГЕ

Щоб динамізувати атрибут, потрібно двічі нажати на назві ЛК, і в списку, що розкрився, настроїти динамічні властивості за допомогою розділу **Indication type – Вид индикации**, як це показано на рис. Б6.27.

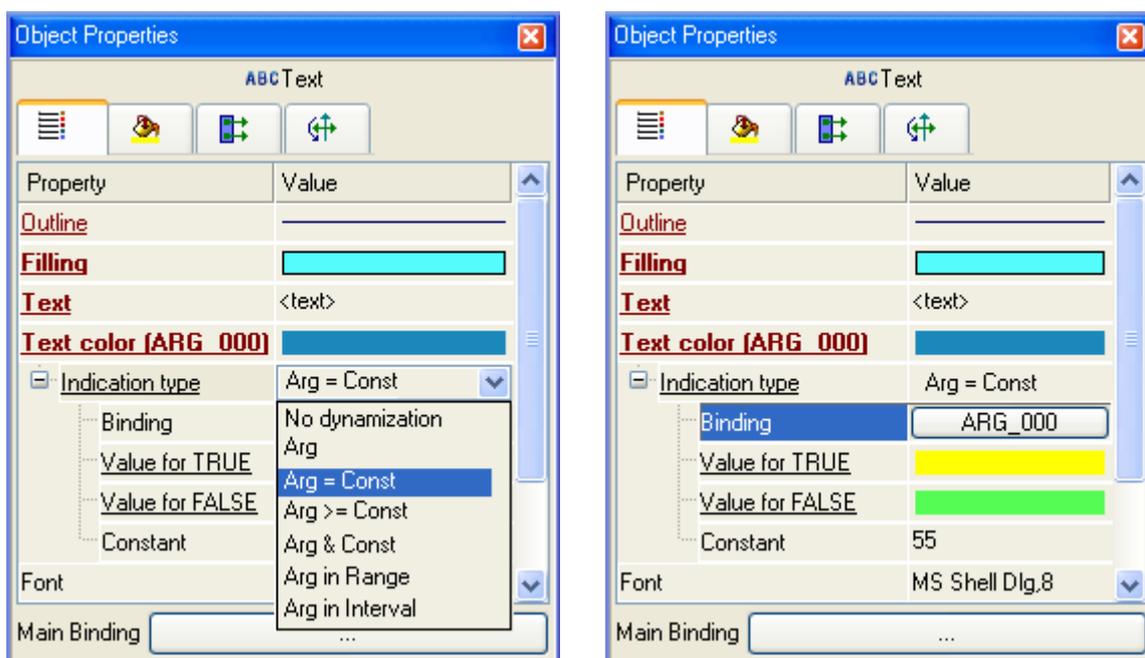


Рисунок Б6.27 – Вікно налаштування параметрів динамізації ГЕ

Вид умови (і, відповідно, вид індикатора, створюваного з ГЕ), вибирається в розділі **Indication type – Вид индикации**:

- ▶ **Value – Значение** – індикація значення аргументу;
- ▶ **Arg = Const** – індикація рівності аргументу заданій константі;
- ▶ **Arg >= Const** – індикація перевищення аргументом заданого порога;
- ▶ **Arg & Const** – індикація результату побітового множення аргументу на значення заданої константи;
- ▶ **Arg у діапазоні** – індикація знаходження аргументу в заданих діапазонах;

► **Arg в інтервалі** – індикація знаходження аргументу в інтервалах прив'язаного каналу.

Залежно від обраного виду індикації міняються інструменти його конфігурування.

Індикація значення. Вид індикації **Value – Значення** може бути заданий при динамізації атрибута **Text – Текст**. Формат виводу вибирається в списку **Format – Формат**, як це показано на рис. Б6.28.

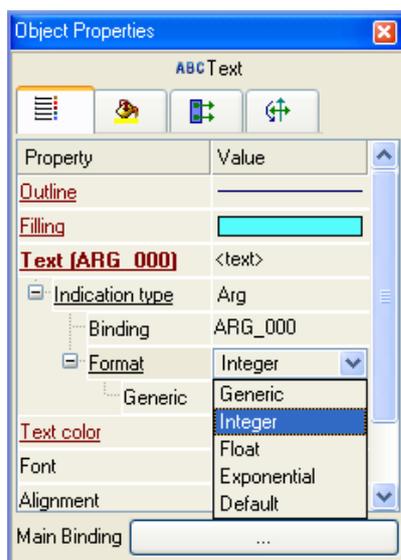


Рисунок Б6.28 – Вікно налаштування параметрів динамізації тексту

Основна прив'язка. Вікно властивостей деяких ГЕ містить розділ **Main Binding – Основная привязка**, як зображено на рис. Б6.29.



Рисунок Б6.29 – Кнопка для завдання прив'язки

При натисканні на кнопку  виводиться стандартний діалог вибору аргументу. Після установки якого-небудь аргументу як основної прив'язки він діє для всіх динамізуємих атрибутів даного ГЕ.

Б6.11 Динамічні властивості ГЕ

До динамічних властивостей графічних елементів відносяться **динамическая заливка**, три види **динамической трансформации** (**перемещение, масштабирование й вращение**) та **динамический контур**.

Динамічні властивості ГЕ, як і динамізовані атрибути, використовуються для графічного відображення значень аргументів екрана при роботі в реальному часі.

Динамічні властивості настраюються відповідно на вкладках  – **Динамическая заливка**,  – **Динамическая трансформация** і  – **Динамический контур** вікна **Свойства объекта**.

Б6.11.1 Динамічне заливання графічних елементів

При використанні даної властивості ГЕ відображає значення прив'язаного аргументу числового формату у вигляді зафарбованої області (шару). Підтримуються два види динамічного заливання – **одноуровневое** (відображає значення одного аргументу) і **многоуровневое** (відображає значення декількох аргументів). Обидва види настраюються на вкладці  – **Динамическая заливка** вікна **Свойства объекта**. Для використання динамічного заливання потрібно на цій вкладці встановити прапорець **Использовать**.

Можна задати кілька шарів заливання. Для додавання/видалення нового шару використовується контекстне меню, що визивається натисканням ПК на назвах пунктів **Слои/Слой** відповідно. Настроювання для всіх створюваних шарів мають однакове призначення.

Б6.11.2 Динамічне переміщення графічних елементів

Ця властивість настраюється в вкладці  – **Динамическая**

трансформация вікна **Свойства** **объекта** при встановленні прапорця **Перемещать**, як це зображено на рис. Б6.30.

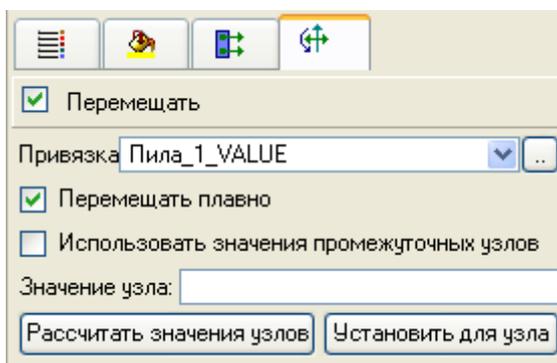


Рисунок Б6.30 – Вкладка для налаштування динамічного переміщення ГЕ

При роботі в реальному часі графічний елемент переміщається уздовж траєкторії, що задається як ламана лінія (кількість вузлів ламаної не обмежена). Поточне положення ГЕ залежить від значення прив'язаного аргументу (числовий аргумент для прив'язки вибирається в списку **Привязка**), від значень, заданих для вузлів траєкторії, і прапорця **Перемещать плавно**.

Б6.11.3 Динамічне масштабування графічних елементів

Ця властивість настраюється в вкладці  – **Динамическая трансформация** вікна **Свойства** **объекта** при встановленні прапорця **Масштабировать**, як це зображено на рис. Б6.31.

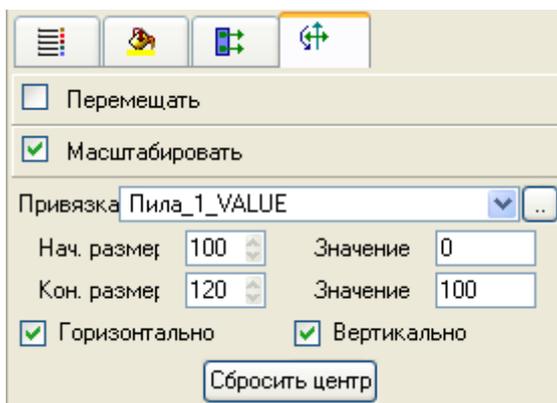


Рисунок Б6.31 – Вкладка для налаштування динамічного масштабування ГЕ

Застосування властивості **Масштабировать** змінює координати точок ГЕ відносно центра масштабування, який у режимі редагування відображається у вигляді червоної точки, як це зображено на рис. Б6.32. За згодою центр масштабування розташовується в точці прив'язки ГЕ (рис. Б6.32,а). За допомогою методу drag-and-drop центр масштабування може бути переміщений у довільну точку екрана (рис. Б6.32,б). При натисканні кнопки **Сбросить центр** центр масштабування встановлюється в точку перетинання діагоналей прямокутника, що обмежує ГЕ (рис. 6.32,в).

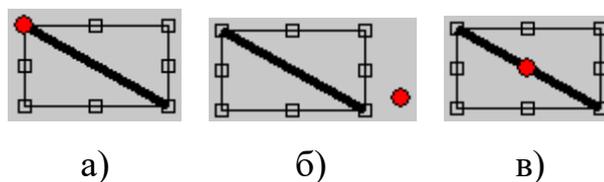


Рисунок Б6.32– Відображення центру масштабування ГЕ:

а) – за згодою; б) – при переміщенні; в) – при скиненні

Б6.11.4 Динамічне обертання графічних елементів

Ця властивість настраюється в вкладці  – **Динамическая трансформация** вікна **Свойства объекта** при встановленні прапорця **Вращать**, як це зображено на рис. Б6.33.

Відображення та зміна розташування центру обертання, відносно якого обертається ГЕ, виконується так само, як і у випадку з центром масштабування (рис. Б6.32).

Б6.11.5 Динамічний контур графічних елементів

Динамічний контур являє собою пунктир, що прокручується за годинниковою стрілкою (під прокручуванням мається на увазі дискретне

переміщення із кроком, рівним довжині штриха). Ця властивість настраюється на вкладці  – **Динамический контур** вікна **Свойства объекта**, як це зображено на рис. Б6.34.

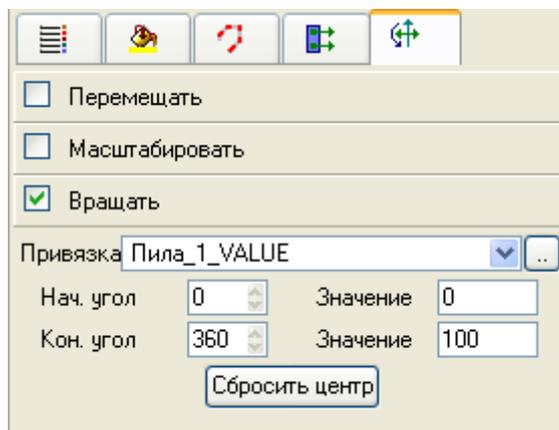


Рисунок Б6.33– Вкладка для налаштування динамічного обертання ГЕ

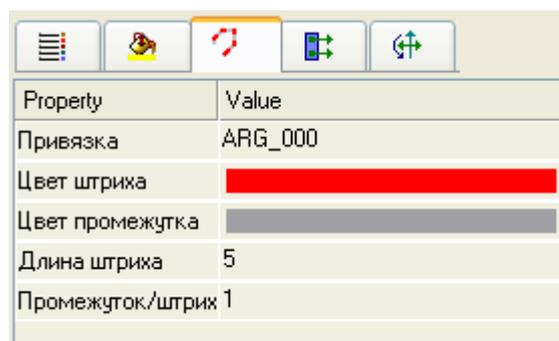


Рисунок Б6.34– Вкладка для налаштування динамічного контуру ГЕ

6.12. Функції керування графічних елементів

Функції керування ГЕ – це дії, що задані для графічних елементів на етапі редагування проекту АСУ. Виконання цих дій при роботі в реальному часі ініціалізується оператором за допомогою мишки. Завдання функцій керування для графічних елементів надає графічним екранам властивість інтерактивності й забезпечує одне з найважливіших якостей АСУ – керування технологічним процесом за допомогою графічних засобів. Функції керування задаються на вкладці  – **События** вікна **Свойства объекта**, як це зображено на рис. Б6.35.

Access Code	0
Events	
pressed	
Confirmation	False
Beep	False
released	
clicked	

Рисунок Б6.35– Вкладка для налаштування функцій керування ГЕ

Визначено наступні **события**, по яких ініціалізується виконання дій у реальному часі:

- ▶ **нажатие** ЛК на ГЕ;
- ▶ **отжатие** ЛК на ГЕ.

Для кожної з подій незалежно може бути задано кілька функцій керування, обраних з контекстного меню (меню відкривається при натисканні ПК на назві події), як показано на рис. Б6.36:

- ▶ **Send Value** – передати значення;
- ▶ **Show/Hide Elements** – показати/сховати елементи;
- ▶ **Jump to Screen** – перейти на екран;
- ▶ **Send comment** – послати коментарий.

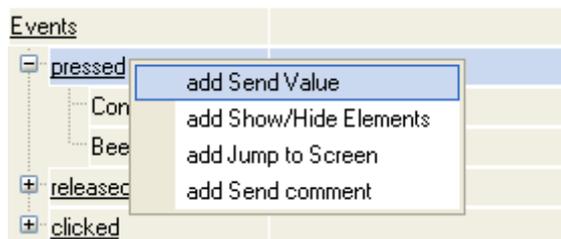


Рисунок Б6.36 – Контекстне меню для вибору функцій керування ГЕ

Функції керування відображаються у вигляді нових розділів списку властивостей об'єкта (для кожної функції створюється окремий розділ). Для видалення функції керування використовується контекстне меню, викликуване натисканням ПК на її назві. Для кожної події можна задати підтвердження й

звуковий сигнал. Для цього використовуються атрибути **Подтверждение** й **Сигнал**. При установці підтвердження в режимі реального часу при здійсненні зазначеної дії на екран виводиться діалогове вікно, що зображене на рис. Б6.37.

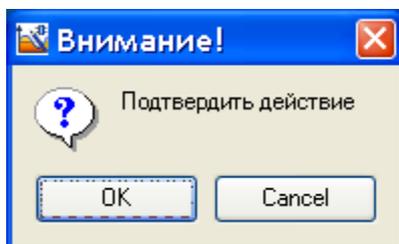


Рисунок Б6.37– Діалогове вікно, що виводитиметься щоразу при підтвердженні певної дії

Щоб підтвердити дію потрібно натиснути клавішу **ОК**, а для скасування – **Cancel**. Якщо встановлено **Сигнал**, то при здійсненні зазначеної дії система відтворює **Стандартний звук**, заданий в Windows.

Код доступа – код доступу до функцій керування (0 – 255). Права на доступ до функцій керування задаються для користувача у вигляді маски в розділі **Доступ/Формы** каналу **Пользователь**. При кореляції маски з кодом доступу (результат побітового логічного множення відмінний від нуля) доступ до функцій керування дозволено, у протилежному випадку – заборонено.

Б6.12.1 Функція передачі значення

Функція передачі значення використовується для зміни значення аргументів екрана. Для одного ГЕ можна задати кілька функцій передачі значення, стосовно до різних аргументів. При додаванні цієї функції керування в списку властивостей об'єкта з'являється розділ, зображений на рис. Б6.38.

Events	
[-] pressed	
Confirmation	False
Beep	False
[-] Send Value	
Send Type	Direct
Value	0
Destination	...
Source	...
Restore Value	False

Рисунок Б6.38 – Розділ для налагодження функції передавання значення

Поле **Send Type** – **Тип передачі** містить наступні варіанти для передачі значень:

- ▶ **Прямая** – сформоване значення задається безпосередньо в поле **Value** – **Значение**;
- ▶ **Ввести и передать** – сформоване значення задається в діалоговому вікні, що з'являється при запуску проекту в моніторі реального часу, якщо відбулася задана подія;
- ▶ **НЕ-ИЛИ** – сформоване значення є результатом логічної операції виключаючого додавання між аргументом і числом, зазначеним у полі **Value** – **Значение**;
- ▶ **ИЛИ** – сформоване значення є результатом логічної операції додавання між аргументом і числом, зазначеним у полі **Value** – **Значение**;
- ▶ **И** – сформоване значення є результатом логічної операції множення між аргументом і числом, зазначеним у полі **Value** – **Значение**;
- ▶ **Прибавить** – сформоване значення є поточним значенням аргументу, збільшеним на число, задане в полі **Value** – **Значение**;
- ▶ **Добавить процент шкалы** – сформоване значення є поточним значенням аргументу, збільшеним на відсоток від величини шкали прив'язаного до аргументу каналу (відсоток шкали задається в полі **Value** – **Значение**);
- ▶ **Умножить** – сформоване значення є добутком поточного значення

аргументу на число, задане в полі **Value – Значение**;

► **Разделить** – сформоване значення є часткою від поділу поточного значення аргументу на число, задане в полі **Value – Значение**.

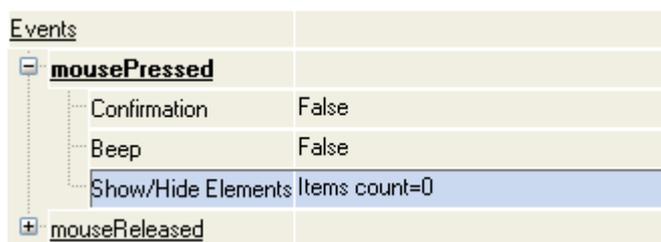
Атрибут **Source – Источник** задає вихідний аргумент, з яким проводиться обрана операція. Результат операції записується в аргумент, що задається атрибутом **Destination – Результат**. Атрибути **Source – Источник** й **Destination – Результат** можуть мати прив'язку до того ж самого аргументу.

Атрибут **Restore Value – Восстанавливать значение** використовується тільки для ГЕ, що запускають виконання дій по натисканню мишки. Якщо цей атрибут має значення **True**, то по натисканню ЛК значення аргументу буде змінено, а по відпусканню – відновлено назад.

Б6.12.2 Функція керування видимістю графічних елементів

Функція керування видимістю ГЕ служить для приховання й/або відображення (залежно від поточного стану) одного або декількох обраних ГЕ на графічному екрані.

При додаванні функції **Show/Hide Elements – показати/сховати елементи** в списку налаштувань з'являється відповідний рядок, як це показано на рис. Б6.39.



Events	
<input type="checkbox"/> mousePressed	
Confirmation	False
Beep	False
Show/Hide Elements	Items count=0
<input type="checkbox"/> mouseReleased	

Рисунок Б6.39– Рядок для налагодження видимості ГЕ

Для вибору керованих ГЕ потрібно натиснути ЛК в полі **Value – Значение** рядка **Show/Hide Elements – показати/сховати елементи**. При цьому його значення прийме вид, зображений на рис. Б6.40.

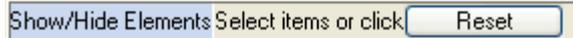


Рисунок Б6.40 – Вид рядка налаштування видимості ГЕ при натисканні ЛК

Після цього на графічному екрані потрібно ЛК вибрати ГЕ (або декілька ГЕ, натискаючи на них ЛК при натиснутій клавіші **Control**). Після цього з'явиться напис **Items count – Всего = N**, де **N** – число виділених ГЕ, як показано на рис. Б6.41.

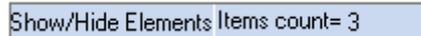


Рисунок Б6.41 – Вид рядка налаштування видимості ГЕ після вибору ГЕ

При запуску проекту в реальному часі при виникненні зазначеної події обрані ГЕ ховаються з екрана, при повторному виникненні знову відображаються і т.д.

Щоб редагувати список обраних ГЕ, потрібно натиснути на поле **Value** – **Значение** рядка **Show/Hide Elements** – **показать/скрыть элементы** і додати/зняти виділення для відповідних елементів. При цьому кнопка **Reset** – **Сбросить** очищує список обраних ГЕ.

Б6.12.3 Функція переходу на екран

Функція переходу на екран здійснює перемикавання з поточного екрана на інший при виникненні заданого **События**. При створенні функції керування в списку налаштувань з'являється відповідний рядок, показаний на рис. Б6.42.

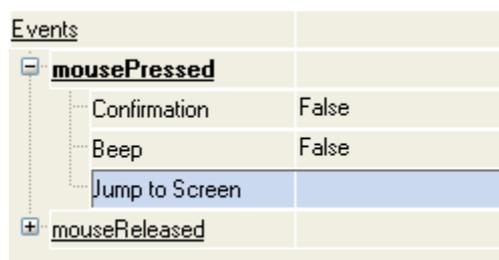


Рисунок Б6.42 – Рядок для налагодження переходу на екран

При натисканні ЛК в поле **Value – Значение** рядка **Jump to Screen – перейти на екран** з'являється список вибору екрану, на який потрібно перейти, як це показано на рис. Б6.43.



Рисунок Б6.43 – Меню вибору екрану, на який буде здійснюватись перехід

Перемиканням екранів у реальному часі можна також управляти за допомогою каналу класу **ВЫЗОВ**.

Б7 ОПИС ВБУДОВАНИХ ГРАФІЧНИХ ЕЛЕМЕНТІВ

Б7.1 Група графічних елементів **Ломаные**

У цю групу входять наступні ГЕ:

- ▶  – **Ломаная линия;**
- ▶  – **Многоугольник;**
- ▶  – **Труба.**

Графічні елементи **Ломаная линия** й **Многоугольник** розміщуються в графічному шарі стандартним способом.

ГЕ **Труба** по суті є елементом об'ємної графіки, і має з ними всі загальні специфічні властивості. Специфічні атрибути **Край 1** і **Край 2** задають різноманітну форму країв труби, показану на рис. Б7.1.

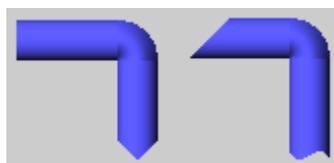


Рисунок Б7.1 – Різноманітні форми країв труби:

Край 1 – зліва; **Край 2** – праворуч

Б7.2 Група графічних елементів **Прямоугольники**

У цю групу входять наступні ГЕ:

- ▶  – **Прямоугольник;**
- ▶  – **Панель;**
- ▶  – **Рамка.**

Дані елементи розміщуються в графічному шарі стандартним способом. Поряд з типовими параметрами, ГЕ **Панель** і **Рамка** мають ряд специфічних

утопленому (праворуч) положеннях

‣ **System colors** – **Системные цвета** – при значенні атрибута **True** для ГЕ ігноруються кольори, задані настроюваннями **Border color** – **Цвет контура** й **Fill color** – **Цвет заливки**. Використаються системні кольори MS Windows.

Б7.3 Група графічних елементів **Объемная графика**

У цю групу входять наступні ГЕ:

-  – **Цилиндр**;
-  – **Сфера**;
-  – **Конус**;
-  – **Тор**;
-  – **Пирамида**;
-  – **Емкость**;
-  – **Клапан**.

Дані елементи розміщаються в графічному шарі стандартним способом. Поряд з типовими параметрами **Скрыть на старте**, **Подсказка** й **Базовый цвет**, об'ємні ГЕ мають ряд специфічних атрибутів. На рис. Б7.5 показана вкладка  – **Основные свойства** вікна властивостей ГЕ **Цилиндр**.

Property	Value
Базовый цвет	
Коэффициент яркости диффузного отражения	100
Коэффициент зеркального отражения	0
Коэффициент яркости зеркального отражения	0
Прозрачность	0
Текстура	
Масштабирование текстуры	False
Качество (%)	0
Толщина стенок	0
Край 1	Прямой
Край 2	Прямой
Ориентация	Вертикально
Скрыть при старте	False
Подсказка	

Рисунок Б7.5 – Вкладка **Основные свойства** ГЕ **Цилиндр**

Б7.3.1 Загальні специфічні атрибути об'ємних графічних елементів

Специфічні атрибути, загальні для всіх ГЕ групи **Объемная графика**, зображені на рис. Б7.6 із значеннями атрибутів за згодою.

Базовый цвет	
Коэффициент яркости диффузного отражения	100
Коэффициент зеркального отражения	0
Коэффициент яркости зеркального отражения	0
Прозрачность	0
Текстура	
Масштабирование текстуры	False
Качество (%)	0
Толщина стенок	0
Скрыть при старте	False

Рисунок Б7.6 – Загальні атрибути для всіх ГЕ групи **Объемная графика**

У РПД використовується модель висвітлення об'ємного елемента одним

джерелом білого світла (положення джерела задається в діалозі **Параметры экрана**. При цьому колірні й відбивні характеристики ГЕ задаються за допомогою наступних атрибутів:

‣ **dc** – **Коэффициент яркости диффузного отражения** – коефіцієнт (0 – 100), що визначає яскравість дифузійно-відбитого світла. У внутрішньому поданні значення коефіцієнта дифузійно-відбитого світла рівні відповідним базовим значенням RGB (задаються за допомогою атрибута **Базовый цвет**), помноженим на **0.01dc**.

‣ **k** – **Коэффициент зеркального отражения** – кількісна характеристика дзеркального відбиття, задається як число в діапазоні 0 – 128 (128 відповідає 100-процентному відбиттю).

‣ **sc** – **Коэффициент яркости зеркального отражения** – коефіцієнт (0 – 100), що визначає яскравість дзеркально відбитого світла. У внутрішньому поданні значення коефіцієнта дзеркально відбитого світла дорівнює відповідним базовим значенням RGB (задаються за допомогою атрибута **Базовый цвет**), помноженим на **0.01sc**.

Атрибут **Прозрачность** визначає ступінь прозорості ГЕ. Цей параметр задається у відсотках (0 – 100), 0 відповідає абсолютній непрозорості. На рис. Б7.7 показані абсолютно непрозорий і напівпрозорий циліндри.

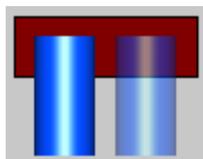


Рисунок Б7.7 – Непрозорий (зліва) і напівпрозорий (праворуч) циліндри

За допомогою атрибута **Текстура** поверхня об'ємного ГЕ може бути текстуризована. При натисканні кнопки в розділі конфігурування атрибута **Текстура** на екрані з'являється навігатор ресурсної бібліотеки растрових рисунків, зображений на рис. Б7.8.

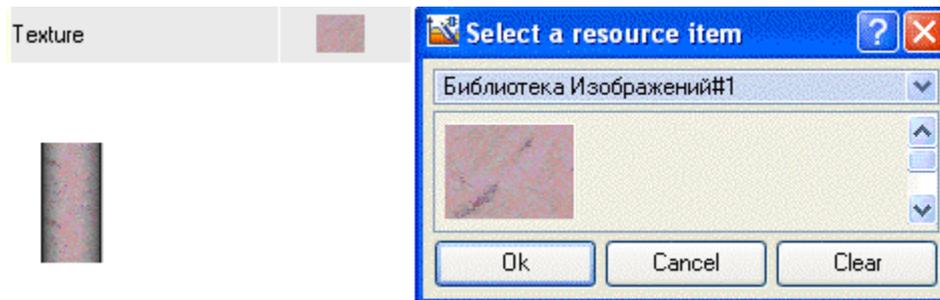


Рисунок Б7.8 – Навігатор ресурсної бібліотеки растрових малюнків

Атрибут **Масштабирование текстуры** при значенні **True** масштабує розмір зображення текстури до розмірів ГЕ. При значенні **False** текстура має оригінальний розмір.

Атрибут **Качество** визначає ступінь промальовування текстури у %. На рис. Б7.9 показана та сама текстура із установленою якістю 100, 50 й 20 %.



Рисунок Б7.9 – Промальовування текстури із установленою якістю 100, 50 й 20 % (зліва направо)

За допомогою атрибута **Толщина стенок стінки**, обмежуючі об’ємний ГЕ, можна зробити видимими.

Б7.3.2 Специфічні атрибути графічного елемента **Цилиндр**

Атрибути, специфічні для ГЕ **Цилиндр**, показані на рис. Б7.10.

Край 1	Прямой
Край 2	Прямой
Ориентация	Вертикально

Рисунок Б7.10 – Атрибути ГЕ **Цилиндр**

За допомогою конфігурування атрибутів **Край 1** і **Край 2** можна задавати різноманітну форму основ циліндра, як це показано на рис. Б7.11.

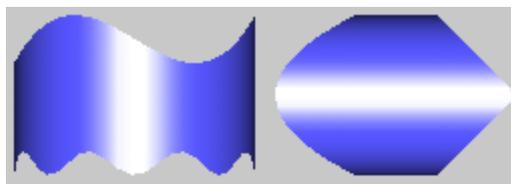


Рисунок Б7.11 – Різноманітні форми основ циліндру:

Край 1 – зліва; **Край 2** – праворуч

Атрибут **Орієнтація** задає розташування утворюючої для ГЕ **Циліндр** (не слід плутати цю функцію з поворотом ГЕ). За згодою установлене значення **Вертикально**, що відповідає вертикальному розташуванню утворюючої. На рис. Б7.12 показаний той самий циліндр, для якого атрибут **Орієнтація** встановлено значення **Вертикально** (зліва) і **Горизонтально** (праворуч).

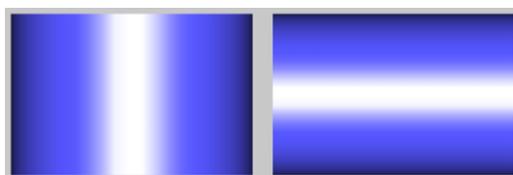


Рисунок Б7.12 – Різноманітні орієнтації циліндру:

вертикально – зліва; горизонтально – праворуч

Б7.3.3 Специфічні атрибути графічного елемента **Сфера**

Специфічним атрибутом для ГЕ **Сфера** є **Отображаемая часть**. Атрибут може приймати два значення – **Часть** й **Сектор**, залежно від яких міняються два нижніх списки, як це показано на рис. Б7.13.

Отображаемая часть	Часть	Отображаемая часть	Сектор
Параметр 1	Полностью	Начальный угол	90
Параметр 2	Часть 1	Угол разворота	360

Рисунок Б7.13 – Настройки відображуваної частини сфери для випадків:
Частина (зліва) та **Сектор** (праворуч)

У першому випадку параметр **Часть** визначає, яка саме частина повинна бути відображена (при виборі опції **Полностью** цей параметр ігнорується). На рис. Б7.14 показані варіанти відображення ГЕ **Сфера** для всіх можливих варіантів налаштування.

У другому випадку для ГЕ задається сектор відображення. Кут розвороту сектора (задається параметром **Угол разворота**) відраховується від деякого початкового напрямку; кут між початковим напрямком і горизонтальним напрямком до правої сторони екрана задається параметром **Начальный угол**. Обидва кути задаються в градусах (0 – 360) і відраховуються за годинниковою стрілкою.

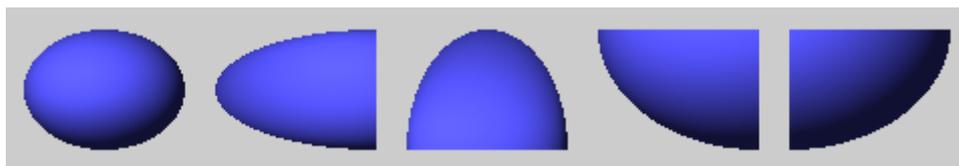


Рисунок Б7.14 – Різні варіанти відображення ГЕ **Сфера** для випадку **Отображаемая часть – Часть** при наступних параметрах (зліва направо): **Полностью; Половина – Часть 3; Половина – Часть 4; Четверть – Часть 2; Четверть – Часть 1.**

На рис. Б7.15 показані варіанти відображення ГЕ **Сфера** для випадків 90 й 180, 180 й 270.

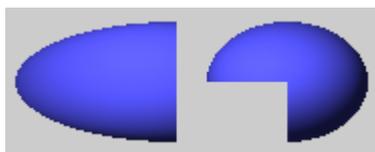


Рисунок Б7.15 – Різні варіанти відображення ГЕ **Сфера** для випадку **Отображаемая часть – Сектор** при наступних параметрах (зліва направо): 90 й 180, 180 й 270

Б7.3.4 Специфічні атрибути графічного елемента **Конус**

Специфічними атрибутами для ГЕ **Конус** є **Отображаемая часть**, **Отношение оснований** й **Ориентация**.

Інструменти конфігурування атрибута **Отображаемая часть** призначені для завдання відображуваної частини ГЕ **Конус**. Атрибут може приймати два значення – **Часть** й **Сектор**, аналогічно настройкам відображуваної частини ГЕ **Сфера**.

На рис. Б7.16 показані варіанти відображення ГЕ **Конус**, коли **Отображаемая часть** має значення **Часть** при всіх можливих варіантів налаштування.



Рисунок Б7.16 – Різні варіанти відображення ГЕ **Конус** для випадку **Отображаемая часть – Часть** при наступних параметрах (зліва направо): **Полностью; Половина – Часть 3; Половина – Часть 4; Четверть – Часть 2; Четверть – Часть 1.**

На рис. Б7.17 показані варіанти відображення ГЕ **Конус**, коли **Отображаемая часть** має значення **Сектор** для випадків 90 й 180, 180 й 270.



Рисунок Б7.17 – Різні варіанти відображення ГЕ **Конус** для випадку **Отображаемая часть – Сектор** при наступних параметрах (зліва направо): 90 й 180, 180 й 270

Значення атрибута **Отношение оснований (%)** задається у відсотках і визначає співвідношення діаметрів основи (0 – 100, значення за згодою – 50, при значенні 100 діаметр меншої основи дорівнює 1 рх, при значенні 0 діаметри основ рівні). На рис. Б7.18 зображено бокс для завдання атрибута **Отношение оснований**.



Рисунок Б7.18 – Бокс для завдання атрибута **Відношення основ**

На рис. Б7.19 показаний той самий конус при наступних значеннях атрибута **Отношение оснований**: 50, 100, 0.

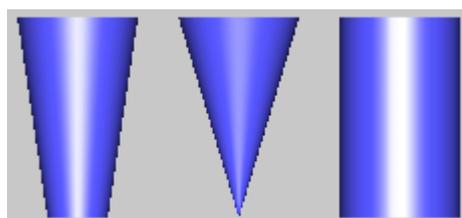


Рисунок Б7.19 – Різні варіанти відображення ГЕ **Конус** при наступних значеннях параметру **Отношение оснований** (зліва направо): 50, 100, 0

Б7.3.5 Специфічні атрибути графічного елементу **Тор**

Специфічними атрибутами для ГЕ **Тор** є **Отображаемая часть** й **Толщина**.

Інструменти конфігурування атрибута **Отображаемая часть** призначені для завдання відображуваної частини ГЕ **Тор**. Атрибут може приймати два значення – **Часть й Сектор**, аналогічно настройкам відображуваної частини сфери.

На рис. Б7.20 показані варіанти відображення ГЕ **Тор**, коли **Отображаемая часть** має значення **Часть** при всіх можливих варіантів налаштування.

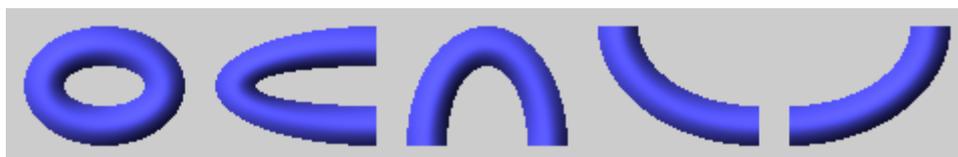


Рисунок Б7.20 – Різні варіанти відображення ГЕ **Тор** для випадку **Отображаемая часть** – **Часть** при наступних параметрах (зліва направо): **Полностью; Половина – Часть 3; Половина – Часть 4; Четверть – Часть 2; Четверть – Часть 1**

На рис. Б7.21 показані варіанти відображення ГЕ **Тор**, коли **Отображаемая часть** має значення **Сектор** для випадків 90 й 180, 180 й 270.

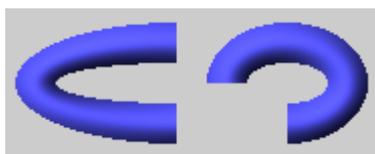


Рисунок Б7.21 – Різні варіанти відображення ГЕ **Тор** для випадку **Отображаемая часть** – **Сектор** при наступних параметрах (зліва направо): 90 й 180, 180 й 270

Атрибут **Толщина** визначає товщину тора й задається як число в діапазоні 0 – 100 (значенню за згодою – 20, при значенні 0 товщина тора дорівнює 1 px, при значенні 100 – 200 px). На рис. Б7.22 зображено бокс для завдання атрибуту **Толщина**.

Рисунок Б7.22 – Бокс для завдання атрибуту **Толщина**

На рис. Б7.23 показаний той самий тор з розмірами обмежуючого прямокутника 90x90 px при наступних значеннях атрибуту **Толщина**: 1, 10, 20.

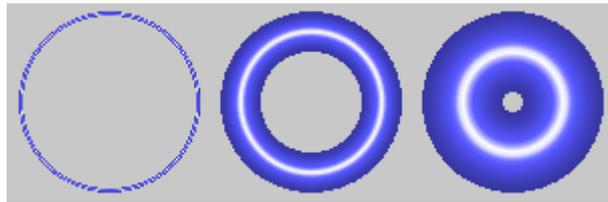


Рисунок Б7.23 – Різні варіанти відображення ГЕ **Тор** при наступних значеннях параметру **Толщина** (зліва направо): 1, 10, 20

Б7.4 Група графічного елемента **Графіки**

У цю групу входить єдиний ГЕ **Тренд** . Даний графічний елемент розміщується в графічному шарі стандартним чином. ГЕ **Тренд** призначений для відображення на графіку зміни у часі значення прив'язаних аргументів. Вигляд цього графічного елемента показаний на рис. Б7.24.

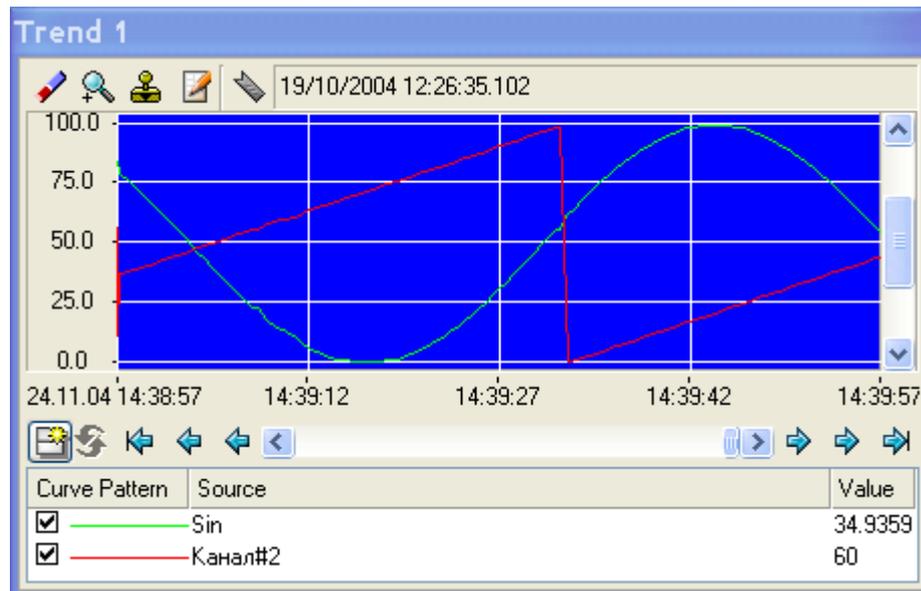


Рисунок Б7.24 – Зовнішній вигляд графічного елемента **Тренд**

Вікно властивостей ГЕ містить наступні вкладки:

 – **Основные свойства;**

 – **Кривые.**

Вкладка **Основные свойства** представлена на рис. Б7.25.

Свойство	Значение
Код доступа	0x0
Использовать архив	True
Ориентация	Горизонтально
Цвет фона	
Шрифт	MS Shell Dlg,8
Масштабируемый	False
Заголовок	
Сетка	
Легенда	
Цвет визира	
Ось времени	
Ось значений	
Буфер	500
Масштаб дискрет (%)	100
Цвета статусов	
* Видимость	True
* Подсказка	
* Слой	Слой
* Геометрия	Скрыть

Рисунок Б7.25 – Вкладка **Основные свойства** ГЕ Тренд

Атрибут **Orientation** – **Ориентация** визначає горизонтальне або вертикальне положення часової шкали на графіку.

Для можливості змінювати розміри вікна тренда в реальному часі передбачений атрибут **Resizable** – **Масштабируемый**.

Назва вікна тренда можна задати за допомогою атрибута **Caption** – **Заголовок**.

Розділ **Grid** – **Сетка** дозволяє за допомогою стандартних інструментів налаштувати кольори й стиль ліній сітки на графіку.

Розділ **Legend** – **Легенда** управляє видимістю легенди під графіком, а також дозволяє налаштовувати шрифт для тексту легенди.

Розділ **Time axe** – **Ось времени** містить наступні налаштування часової осі графіка:

- **Show** – **Показывать** – при установці значення **True** на графіку будуть

видні підписи часової осі;

► **Marks – Разбиение** – визначає, на скільки частин буде розбито видимої частину часової осі;

► **Range – Диапазон** – визначає видимий діапазон значень часової осі (0 – 100);

► **Units – Единицы** – одиниці виміру діапазону. Вибираються з меню: **секунда, минута, час, день**.

Розділ **Value axe – Ось значень** містить наступні налаштування осі значень графіка:

► **Marks – Разбиение** – визначає, на скільки частин розбивати видимої частину осі значень;

► **Show – Показывать** – задає параметри видимості значень на осі. Вибирається з меню, що зображено на рис. Б7.26.



Ось значень	
Разбиение	4
Период подписи	1
Показывать	Только активная

Рисунок Б7.26 – Меню для вибору типу видимості вісі значень

Атрибут **Buffer size – Буфер** задає кількість збережених у пам'яті значень кривих для виводу на екран у реальному часі (500 – 32000). Якщо крива відображає значення, що архівуються до СПАД каналу, то на графік виводяться всі записані значення з архіву.

Для того, щоб одержати на екрані графік, необхідно додати хоча б одну криву на вкладці **Кривые**.

Для додавання кривої (або її видалення) використовується контекстне меню, що викликається натисканням ПК на рядку **Кривые** (або, відповідно, **Кривая**).

Для редагування властивостей кривої передбачені наступні інструменти:

► **Заголовок** – назва кривої;

► **Привязка** – завдання прив'язки кривій до аргументу екрана (при

натисканні ЛК у даному полі на екрані з'являється стандартний діалог вибору аргументу);

- ▶ **Цвет** – завдання кольору кривій (при натисканні ЛК у цьому полі на екрані з'являється стандартний діалог вибору кольору);

- ▶ **Стиль линии** – завдання стилю лінії (при натисканні ЛК у цьому полі виводиться список стилів;

- ▶ **Толщина линии** – задає товщину лінії кривій у пікселях;

- ▶ **Тип меток** – визначає, чи потрібно виділяти точку на екрані на кожному такті перерахування (тип виділення вибирається з меню, представленого на рис. Б7.27).

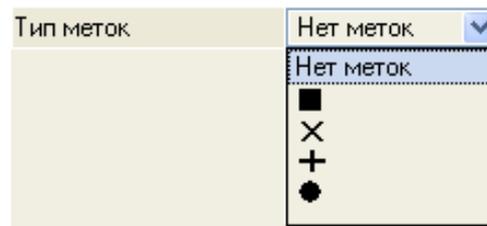


Рисунок Б7.27 – Меню для вибору типу виділення крапок кривої

- ▶ **Стиль линии вне границ** – стиль лінії при переході значень прив'язаного аргументу за межі, що задані атрибутами **Макс. значение** й **Мин. значение** (при натисканні ЛК у цьому полі виводиться стандартний список стилів);

- ▶ **Интерполирование** – при установці цьому атрибуту значення **True**, графік згладжується методом інтерполяції.

Після запуску в реальному часі за допомогою панелі інструментів можна використати наступні настроювання ГЕ **Тренд**:



– кнопка повернення до вихідного масштабу тренда;



– збільшення. Кнопка має два положення – натиснута й віджата. При натиснутій кнопці включений режим збільшення – при натиснутій ЛК потрібно виділити прямокутний діапазон графіка для його збільшеного перегляду;



– переміщати по точках;



– настроювання кривих. При натисканні цієї кнопки виводиться діалог настроювання кривих тренда, ідентичний вкладці **Кривые** вікна **Свойства объекта**. Вікно призначене для додавання/видалення кривих тренда, а також зміни їхніх настроювань, у реальному часі;



– перейти до часової мітки. При натисканні на цю кнопку з'являється діалогове вікно уведення дати й часу. Після натискання кнопки ОК на тренд виводиться інформація, починаючи із зазначеного часу;



– показати панель інструментів. Якщо кнопка не натиснута, панель інструментів не відображається;



– індикатор синхронізації відображуваних даних з архівом **СПАД**. При синхронізації іконка стає кольоровою  ;



– кнопки для переходу по тимчасовій осі графіка. Відповідно **к началу, на день назад, на час назад і на час вперед на день вперед, к концу**;

Для зсуву по осях тренда можна також використати лінійки прокручування.

Щоб у реальному часі змінити масштаб по кожній з осей треба натиснути ЛК в області тренда, а потім натисканням сполучення клавіш **CTRL+<стрілки>** установити необхідний масштаб.

Б7.5 Група графічного елемента **Приборы**

У цю групу входять наступні ГЕ:



– **Ползунок**;



– **Стрелочный прибор**.

Б7.5.1 Графічний елемент **Ползунок**

ГЕ **Slider** – **Ползунок** використовується для завдання й/або індикації значення прив'язаного аргументу, а також для індикації виконання заданої умови. Вигляд цього графічного елемента показаний на рис. Б7.28.

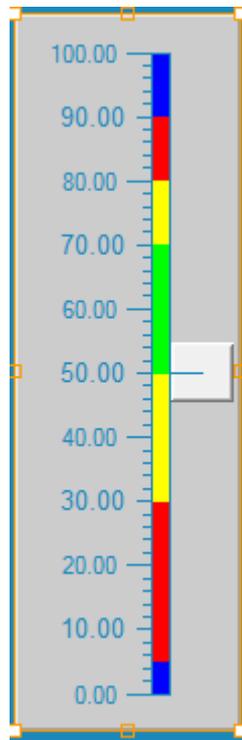


Рисунок Б7.28 – Зовнішній вид ГЕ **Ползунок**

Атрибути ГЕ **Ползунок** конфігуруються на вкладці **Основные свойства** вікна властивостей даного графічного елемента, що представлена на рис. Б7.29.

Свойство	Значение
Код доступа	0
<u>Отображаемая величина</u>	50
<u>Задаваемая величина</u>	
Положение ползунка	Справа
<u>3D-эффекты</u>	False
<u>Ползунок</u>	True
<u>Полоса</u>	True
<u>Шкала</u>	True
<u>Фон</u>	<input type="checkbox"/>
<u>Рамка</u>	<input type="checkbox"/>
* <u>Видимость</u>	True
* <u>Подсказка</u>	
* <u>Слой</u>	Слой
* <u>Геометрия</u>	Скрыть

Рисунок Б7.29 – Вкладка **Основные свойства ГЕ Ползунок**

Атрибути **Monitored Value – Отображаемая величина** й **Controlled Value – Задаваемая величина** визначають функціональне призначення ГЕ.

Щоб ГЕ відображав значення прив'язаного аргументу або виконання деякої умови, потрібно динамізувати його атрибут **Monitored Value – Отображаемая величина**.

При розкритті списку в розділі конфігурування атрибута **Controlled Value – Задаваемая величина** з'являється настроювання вибору аргументу, чиє значення буде задавати повзунок при роботі в реальному часі.

Код доступа – код доступу для можливості керування величиною, що задається (0 – 255). Права доступу задаються для користувача у вигляді маски в розділі **Доступ/Формы** каналу **Пользователь**. При кореляції маски з кодом доступу (результат побітового логічного множення відмінний від нуля) доступ до керування дозволений, у протилежному випадку – заборонений.

Атрибут **Orientation – Положение ползунка** задає положення повзунка відносно шкали й відповідне розташування штрихів і чисел на шкалі. На рис. Б7.30 показаний один і той же ГЕ з наступними значеннями атрибута **Orientation – Положение ползунка** (зліва направо): **Справа, Слева, Сверху** та

Снизу.

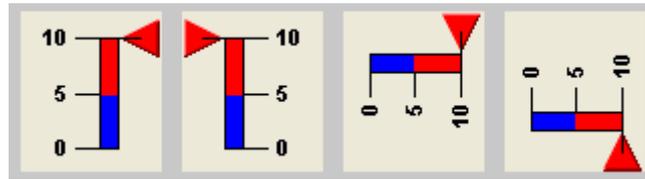


Рисунок Б7.30 – Різні варіанти відображення ГЕ **Ползунок** при наступних значеннях параметру **Положение ползунка** (зліва направо): **Справа, Слева, Сверху, Снизу**

У розділі **3D Effects – 3D-эффекты**, зображеному на рис. Б7.31, при установці атрибуту **Use – Использовать** значення **True** колірні смуга й повзунок приймають об'ємний вид, а також **Band Edges – Края полосы** можуть бути задані як **Square – Квадратные** або **Round – Круглые**, як показано на рис. Б7.32.

3D Effects	
Use	True
Band Edges	Round

Рисунок Б7.31 – Розділ **3D- эффекты** вкладки **Основные свойства ГЕ Ползунок**

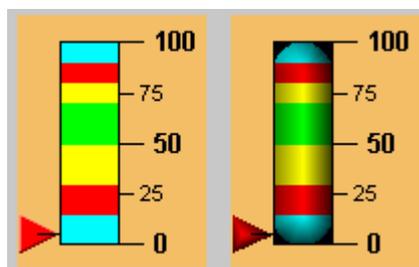


Рисунок Б7.32 – Різні варіанти відображення ГЕ **Ползунок** при наступних значеннях параметру **Края полосы** (зліва направо): **Квадратные та Круглые**

Для відображення повзунка на шкалі треба встановити атрибут **Use – Использовать** в розділі **Slider – Ползунок**, що зображений на рис. Б7.33, значення **True**. Форма повзунка (■ – Квадрат, ▲ – Треугольник, ▲ – Домик), його кольори та розмір у пікселях визначаються відповідними атрибутами. **Hairline Lenght (%)** – Длина риски на повзунку задається у відсотках відносно розміру повзунка.

Ползунок	True
Тип	Квадрат
Цвет	
Размер	31
Длина риски (%)	50

Рисунок Б7.33 – Розділ **Ползунок** вкладки **Основные свойства ГЕ Ползунок**

Розділ **Band – Полоса** містить наступні атрибути, що зображені на рис. Б7.34 та визначають параметри колірної смуги шкали.

Полоса	True
Ширина	10
+ Верхний предел шкалы	100
+ HL	90
+ HA	80
+ HW	70
+ LW	50
+ LA	30
+ LL	5
+ Нижний предел шкалы	0
Цвет (HW, LW)	
Цвет (HA, HW), (LW, LA)	
Цвет (HL, HA), (LA, LL)	
Цвет >HL, <LL	

Рисунок Б7.34 – Розділ **Полоса** вкладки **Основные свойства ГЕ Ползунок**

Для відображення колірної смуги шкали потрібно встановити атрибуту **Use – Использовать** значення **True** (за згодою). На рис. Б7.35 показана шкала з

колірною смугою й без неї.

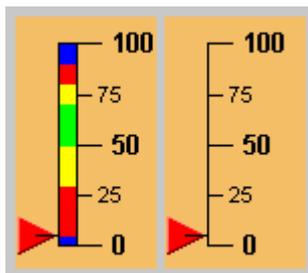


Рисунок Б7.35– Шкала з колірною смугою (ліворуч) й без неї (праворуч)

Width – Ширина колірної смуги задається в пікселях. Колірна смуга використовується для колірного кодування заданих інтервалів значень відображуваної/величини. За аналогією з каналом, визначення інтервалів відбувається за допомогою завдання шести границь. Для різних інтервалів можна задати свої кольори на смугі.

Розділ **Marks – Шкала** містить параметри розбивки й підписів шкали, як це показано на рис. Б7.36.

Шкала	
Ширина	1
Цвет текста	
[-] <u>Уровень 1</u>	
Использовать	True
Число делений	5
Длина штриха	12
Показать значения	True
Шрифт	MS Shell Dlg.8
Десятичные знаки	2
[+] <u>Уровень 2</u>	
[+] <u>Уровень 3</u>	

Рисунок Б7.36 – Розділ **Шкала** вкладки **Основные свойства** ГЕ **Ползунок**

Атрибути **Width – Ширина** (цей параметр задається в пікселях) і **Text Color – Цвет** визначають відповідно ширину й кольори штрихів і тексту підписів всіх рівнів розбивки шкали. Атрибут **Width – Ширина** визначає

одночасно товщину контуру колірної смуги шкали.

Для шкали можуть бути задані три рівні розбивки. Для кожного рівня (приналежність до рівня вказує число в найменуванні розділу) можуть бути задані наступні параметри:

- ▶ **Use – Использовать** – при установці цього атрибуту значення **True** використовуються розбивка даного рівня, а відповідні штрихи відображаються на шкалі;

- ▶ **Partition – Число делений** – число розподілів даного рівня;

- ▶ **Length – Длина штриха** – довжина штриха даного рівня (у пікселях);

- ▶ **Numbers – Показать значения** – при установці цього атрибуту значення **True** відображаються чисельні значення, що відповідають штрихам даного рівня;

- ▶ **Font – Шрифт** – шрифт, що буде використовуватись для відображення чисельних значень даного рівня;

- ▶ **Fractional – Десятичные знаки** – кількість десяткових знаків у чисельних значеннях даного рівня.

Якщо атрибуту **Use – Использовать** привласнене значення **False**, то інші атрибути цієї розбивки ігноруються.

При конфігуруванні цих атрибутів потрібно враховувати пріоритет рівнів розбивки. Атрибут **Partition – Число делений N** задає число розподілів, на яке розбивається кожен розподіл найближчого більш високого рівня (**N-1** або **N-2**) при його використанні. Якщо жоден з більш високих рівнів розбивки не використовується, то цей атрибут задає число розподілів, на яке розбивається весь діапазон шкали.

Б7.5.2 Графічний елемент **Стрелочный прибор**

ГЕ **Основные свойства** призначений для відображення поточного значення каналу. Колірна смуга цього ГЕ відображає інтервали значень каналу. На рис. Б7.37 показана вкладка **Основные свойства** й вид стрілочного

приладу.

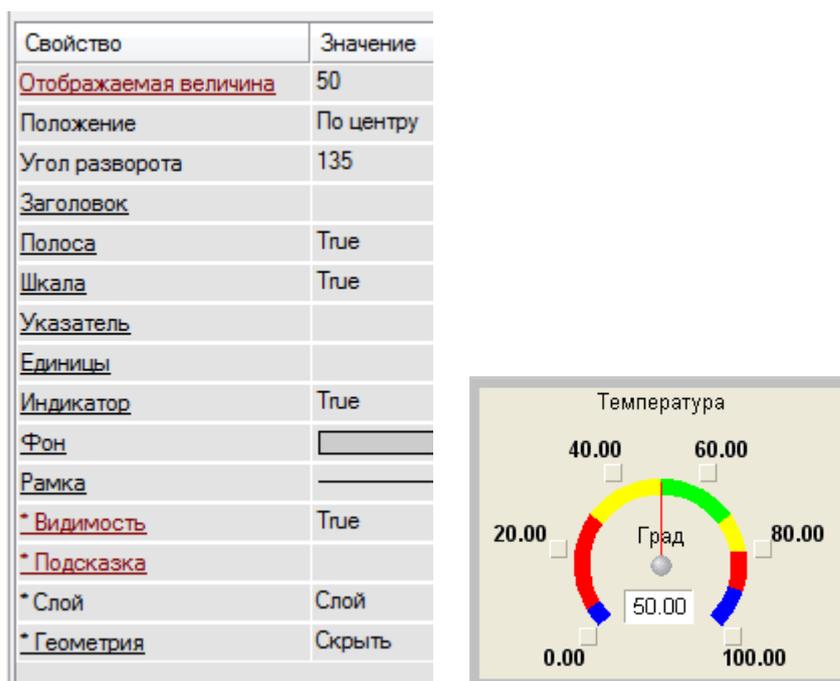


Рисунок Б7.37 – Вкладка **Основные свойства ГЕ Стрелочный прибор** (зліва) та його зовнішній вигляд

Динамізуємий атрибут **Monitored Value – Отображаемая величина** задає значення для відображення на приладі.

Атрибут **Orientation – Положение** задає орієнтацію стрілки приладу відносно шкали та відповідне розташування штрихів і чисел на шкалі. На рис. Б7.38 показаний один і той же ГЕ з різними значеннями атрибута **Положение**.

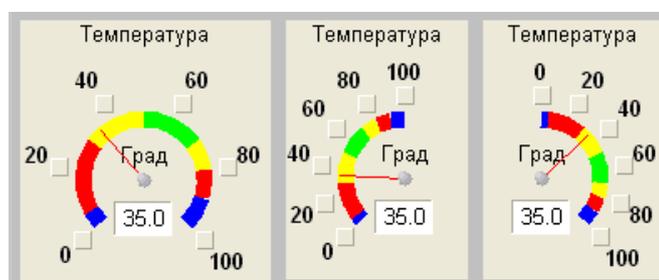


Рисунок Б7.38 – Різні варіанти відображення ГЕ **Стрелочный прибор** при наступних значеннях параметру **Положение** (зліва направо): **По центру**,

Справа, Слева

Scale Angle – Угол разворота шкалы може приймати значення від 45 до 135 градусів, як це показано на рис. Б7.39.

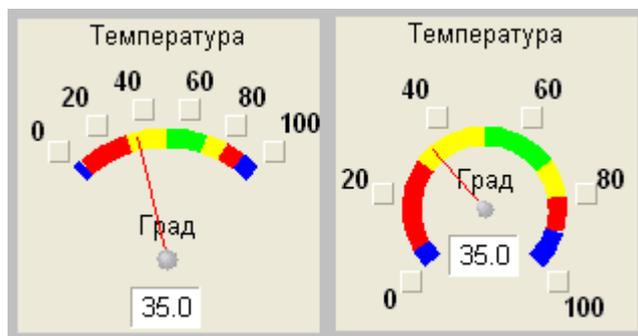


Рисунок Б7.39 – Різні варіанти відображення ГЕ **Стрелочный прибор** при наступних значеннях параметру **Угол разворота** (зліва направо): 45 та 135

Розділ **Caption** – **Заголовок**, зображений на рис. Б7.40, містить атрибути для стандартного настроювання тексту заголовка, його кольору та шрифту.

Заголовок	
Текст	TM Arrow gage
Цвет	
Шрифт	Arial, 10

Рисунок Б7.40 – Розділ **Заголовок** вкладки **Основные свойства ГЕ Стрелочный прибор**

Розділи **Band** – **Полоса** й **Marks** – **Шкала** мають атрибути, аналогічні атрибутам ГЕ **Ползунок**.

У розділі **Arrow** – **Указатель** можна встановити параметри стрілки, такі як **Width** – **Ширина** (в пікселях), **Color** – **Цвет** та **Root Color** – **Цвет основания**, як це показано на рис. Б7.41.

Указатель	
Ширина	1
Цвет	
Цвет основания	

Рисунок Б7.41 – Розділ **Указатель** вкладки **Основные свойства ГЕ**
Стрелочный прибор

Настроювання розділу **Unit – Единицы**, зображеного на рис. Б7.42, визначають параметри виведеної одиниці виміру.

Единицы	
Текст	
Цвет	
Шрифт	Arial, 10

Рисунок Б7.42 – Розділ **Единицы** вкладки **Основные свойства ГЕ**
Стрелочный прибор

За допомогою розділу **Current Value – Индикатор** можна розташувати на ГЕ індикатор поточного значення. Для нього можна встановити властивості **Color – Цвет**, **Font – Шрифт**, **Fractional – Десятичные знаки**, як це показано на рис. Б7.43.

Индикатор	
True	
Цвет	
Шрифт	Arial, 10
Десятичные знаки	3

Рисунок Б7.43 – Розділ **Индикатор** вкладки **Основные свойства ГЕ**
Стрелочный прибор

Розділи **Background – Фон** й **Frame – Рамка** за допомогою стандартних інструментів дозволяють редагувати зовнішній вигляд ГЕ.