

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УКРАЇНСЬКИЙ ДЕРЖАВНИЙ ХІМІКО-ТЕХНОЛОГІЧНИЙ
УНІВЕРСИТЕТ»

МЕТОДИЧНІ ВКАЗІВКИ
ДО ЛАБОРАТОРНИХ ЗАНЯТЬ З ДИСЦИПЛІНИ
«ОСНОВИ ПРОЕКТУВАННЯ СИСТЕМ АВТОМАТИЗАЦІЇ»
ЗА ОСВІТНІМ РІВНЕМ «БАКАЛАВР» ДЛЯ СТУДЕНТІВ
СПЕЦІАЛЬНОСТІ «151 АВТОМАТИЗАЦІЯ ТА КОМП'ЮТЕРНО-
ІНТЕГРОВАНІ ТЕХНОЛОГІЇ»

Затверджено на засіданні кафедри
комп'ютерно-інтегрованих
технологій та автоматизації
Протокол № 2 від 22.11.2018

Методичні вказівки до лабораторних занять з дисципліни «Основи проектування систем автоматизації» за освітнім рівнем «Бакалавр» для студентів спеціальності «151 Автоматизація та комп'ютерно-інтегровані технології» / Укл.: О.Ю. Олійник, Ю.К. Тараненко, Є.В. Чернецький. – Дніпро:ДВНЗ УДХТУ, 2019. – 29 с.

Укладачі: О.Ю. Олійник, канд. техн. наук
Ю.К. Тараненко, д-р техн. наук
Є.В. Чернецький, канд. техн. наук

Відповідальний за випуск О.П. Мисов, канд. техн. наук

Навчальне видання

Методичні вказівки до лабораторних занять з дисципліни «Основи проектування систем автоматизації» за освітнім рівнем «Бакалавр» для студентів спеціальності «151 Автоматизація та комп'ютерно-інтегровані технології»

Укладачі: ОЛІЙНИК Ольга Юріївна
ТАРАНЕНКО Юрій Карлович
ЧЕРНЕЦЬКИЙ Євгеній Вячеславович

Технічний редактор Л.Я. Гоцуцова
Комп'ютерна верстка Л.Я. Гоцуцова

Підписано до друку 29.05.19. Формат 60×84/16. Папір ксерокс. Друк різнограф.
Умов. друк. акр. 1,32. Облік.-вид. акр. 1,37. Тираж 100 прим. Зам. № 225.
Свідоцтво ДК № 5026 від 16.12.2015.

ДВНЗ УДХТУ, 49005, м. Дніпро – 5, просп. Гагаріна, 8.

Редакційно-видавничий відділ

1 РОЗРОБКА АСК У ПРОГРАМНОМУ СЕРЕДОВИЩІ SCADA – СИСТЕМА TRACE MODE

1.1 Вибір системи керування процесом

Одним із сучасних методів впровадження систем автоматизації є застосування SCADA системи, комплексної системи автоматизації яка дозволяє не тільки керувати процесом, а й спостерігати весь процес, зберігати дані за процесом, забезпечити аварійний захист, зменшити людський фактор та полегшити роботу оператора.

Зауважимо, що в більшості випадків SCADA системи дійсно дозволяють значно прискорити процес створення ПЗ верхнього рівня АСУТП, не вимагаючи при цьому від розробника знань сучасних мов програмування.

Ринок SCADA систем на сьогодні дуже широко наданий вітчизняними та зарубіжними компаніями, порівняльна характеристика яких наведена в таблиці 1.1 [1].

Проаналізувавши таблицю 1.1, у тому числі і вартість систем та задачі нашого дослідження, можна зробити висновок, про доцільність застосування системи TRACE MODE 6. Перевагами даної SCADA є те, що в даній системі інтегровано більше 10 різних редакторів проектування автоматизованих систем керуванням виробництва та управління персоналом.

Ці компоненти систем полегшують написання програми та дозволяють об'єднувати різні рівні автоматизації виробництвом без потреби використання інших програмних засобів. Крім того, система TRACE MODE на даний момент є універсальною системою, вона вміщує більше 2500 драйверів для різноманітних пристроїв та засобів автоматизації, а для тих, яких немає, є можливість написання драйвера за допомогою вбудованого редактора, без особливих навичок програмування [2].

Присутність унікальної технології автопобудови дозволяє декількома рухами створити зв'язки між вузлами розподіленої системи управління (PCY), між джерелами даних SCADA і каналами, створити джерела даних за відомою конфігурацією контролера і т. п. Також інтерфейс програми є інтуїтивно зрозумілий та простий, що забезпечує програмісту скоротити час на написання програми, а вбудовані редактори забезпечують постійну перевірку програми, забезпечуючи безпомилковість написання програми.

1.2 Програмне середовище SCADA-система TRACE MODE

1.2.1 Архітектура середовища TRACE MODE

Створення сучасних систем управління базується на розробці та застосуванні адаптивних інтелектуальних систем, функціонування яких неможливе без використання розвинутої обчислювальної мережі, що включає персональні комп'ютери (ПК), мікроконтролери та широкий набір модулів вводу/виводу. Ускладнення технологічних процесів і виробництв ставить

завдання створення розподілених ієрархічних систем (АСУТП) та їх наскрізного програмування, що пояснює появу нових комп'ютерних технологій для інтегрованих систем, які об'єднують всі рівні виробництва.

Таблиця 1.1 – Порівняльний аналіз можливостей SCADA систем

Назва	Склад	Сигнали	Механізм обміну	Мнемосхеми	Симуляція	Драйвери
1	2	3	4	5	6	7
Advantec hStudio	Середовище розробки і середовище виконання	+	OPC	Велика бібліотека компонентів	+	ADAM, Omron, Siemens, Yokogawa
Citect SCADA	Середовище розробки + ліцензовані модулі	Звуковий супровід та анімація	OPC DA, OPC HDA, OPC AE, DDE	Лінії, фігури, зображення, 3D труби, технологічне обладнання	Використання розрахунків скриптами та програмами	Більш, ніж 120 драйверів, до 300 різних пристроїв (ПЛК, УСО)
IFix	VisiconX, i Client, i WebServer	OPC. SCADA серверів і FIX та надсилає їх клієнтам	OPC, OPC A&E	Бібліотека графічних елементів та об'єктів	VBA-скрипти	Велика бібліотека драйверів вводу-виводу з підтримкою OPC
InTouch	InTouch Development, In Touch Runtime, In Touch Read-Only	Відображення розподілених сигналів тривоги та подій	OPC, DDE	Вдосконалена графічна бібліотека SymbolFactory та нова бібліотека графічних символів SmartSymbols	Використання розрахунків скриптами та програмами	Великий вибір серверів вводу-виводу для сотень найбільш популярних пристроїв
Master SCADA	Середовище розробки, середовище виконання, модулі зв'язку з БД	Звукові та графічні варіанти сигналізації	OPC, OLE, DCOM, ActiveX, OLE DB, ODBC	Бібліотека плоских та об'ємних елементів, імпорт зображень, об'ємний трубопровід	Імітаційний режим з вибором функцій імітації сигналів	Adam, MFC, TCM52Teconics, TCM410, Wincon
TraceMode	Система програмування контролерів; система розробки розподіленої АСУТП; середовище розробки ТМ6	Система керування тривогами МРВ, пріоритети тривоги	OPC	Більше 1000 графічних зображень; більше 600 анімаційних об'єктів;	На процедурній мові TechnoST	2527 пристроїв (ABB, АВ), CAN, Profibus, FieldBus, ModBus

Як приклад, може бути названа SCADA-система (Supervisory Control And Data Acquisition), призначена для проектування та експлуатації розподілених автоматизованих систем управління. З огляду на назву, SCADA-система призначена для диспетчерського управління та збору даних. Проте, в останніх версіях її призначення значно розширилося. Зокрема, вітчизняна фірма-виробник Ad Astra Research Group, LTD випустила 6-ту версію SCADA-системи TRACE MODE, яка має потужні засоби для створення розподілених ієрархічних АСУТП, що містять до трьох рівнів ієрархії: рівень контролерів – нижній рівень; рівень операторських станцій – верхній рівень; адміністративний рівень. На ринку програмних продуктів існує багато версій SCADA-систем в основному зарубіжних виробників, наприклад Genesis фірми Iconics, Factory Link фірми United States DATD Co. (США), WinCC фірми Siemens (Німеччина) та ін.

SCADA-система TRACE MODE містить засоби розробки операторського інтерфейсу (SCADA/HMI), програмування контролерів (Softlogic), управління основними фондами (EAM), персоналом (HRM) і виробничими процесами (MES).

Всі програми, що входять у TRACE MODE, поділяються на дві групи (рис. 1.1): інструментальну систему розробки та виконавчі модулі (runtime).

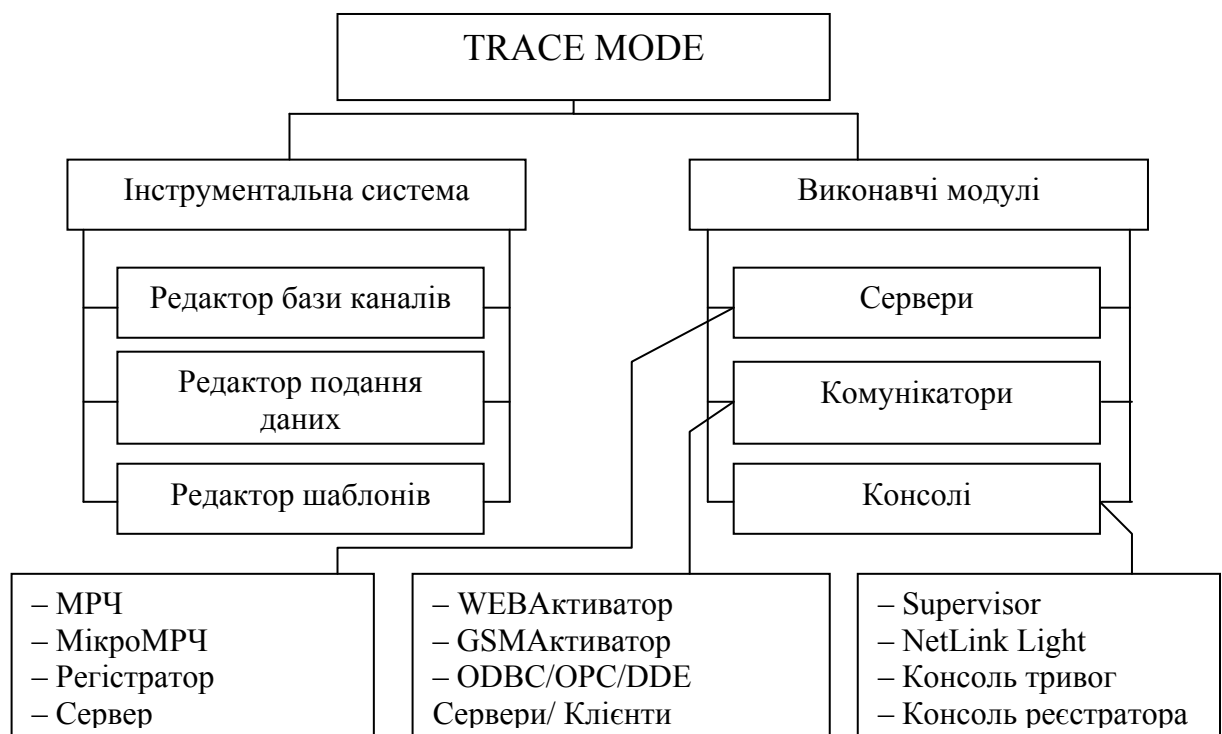


Рисунок 1.1 – Групи програм, які входять до TRACE MODE

Як видно з рисунка, інструментальна система розробки містить три редактори [2]: редактор бази каналів, редактор подання даних, редактор шаблонів. У редакторі бази каналів створюється математична основа системи управління: описуються конфігурації всіх робочих станцій, контролерів і УСО, а також налаштовуються інформаційні потоки між ними. Тут же описуються

вхідні і вихідні сигнали та їх зв'язок з пристроями збору даних і управління; задаються періоди опитування або формування сигналів, настраюються закони первинної обробки та управління, технологічні кордони, програми обробки даних і управління, здійснюється архівування технологічних параметрів, мережний обмін, а також вирішуються деякі інші завдання.

Результатами роботи в цьому редакторі є математична та інформаційна структури проекту АСУТП, які містять набір баз каналів і файлів конфігурації для всіх контролерів і операторських станцій (вузлів) проекту, а також файл конфігурації всього проекту з розширенням `cmf` (для версії 6 розширення – `prj`). Всі інші файли проекту зберігаються в робочій директорії в каталозі, ім'я якого збігається з ім'ям файлу конфігурації [3].

У редакторі подання даних розробляється графічна частина проекту системи управління. Спочатку створюється статичний малюнок технологічного об'єкта, а потім поверх нього розміщуються динамічні форми відображення та управління. Серед цих форм присутні такі, як поля виведення числових значень, графіки, гістограми, кнопки, області введення значень і переходу до інших графічних фрагментів і т. д. [6].

Крім стандартних форм відображення, TRACE MODE дозволяє вставляти в проекти графічні форми подання даних або управління, розроблені користувачами.

Всі форми відображення інформації, управління та анімаційні ефекти зв'язуються з інформаційною структурою, розробленою в редакторі бази каналів.

Для розробки шаблонів документів до складу інструментальної системи включений редактор шаблонів.

Виконавча система TRACE MODE включає в себе виконавчі модулі (монітори, МРВ) – програмні модулі різного призначення, під управлінням яких у реальному часі виконуються складові частини проекту, що розміщуються на окремих комп'ютерах, або в контролерах, призначені для роботи на всіх рівнях систем управління, про які говорилося вище.

Існує ряд програмних модулів, призначення яких чітко не прив'язане до функцій одного з перерахованих рівнів систем управління. До таких модулів належать (рис. 5.1):

- глобальний реєстратор;
- сервер документування;
- Web-активатор;
- GSM-активатор.

Вони можуть використовуватися для створення як оперативного, так та адміністративного рівнів систем управління.

Глобальний реєстратор служить для забезпечення надійного зберігання архівів ТП. Він архівує дані, що посилаються йому по мережі моніторами реального часу (64000 параметрів з дискретністю 0,001 с), забезпечує автоматичне відновлення даних після збою, а також може передавати архівні дані для перегляду моніторам SUPERVISOR. Глобальний реєстратор може також виступати як OPC-сервер і DDE-сервер і підтримує обмін з базами даних через ODBC.

Для документування технологічної інформації в TRACE MODE передбачений спеціальний модуль – сервер документування. Документування здійснюється за шаблонами, які створюються в редакторі шаблонів. Час або умову генерування документа, ім'я файлу шаблону, а також напрямок виведення документа описуються в програмах документування – сценаріях.

Підготовка звітів (документів) найчастіше прив'язується до астрономічному часу. Наприклад, вони можуть генеруватися один раз на годину, один раз на добу, один раз на місяць, або один раз в десять хвилин. Крім того, можна встановити режим підготовки документа один раз на зміну і потім описати розбивку доби на зміни [4].

Сервер документування NetLinkLight використовується для вирішення завдання документування технологічної інформації. Він за командою MPB, власним сценарієм, або за командою оператора, інтерпретує створені заздалегідь шаблони, запрошує у MPB необхідні дані і формує за ними документи. Ці документи можуть бути роздруковані на принтері, відправлені по E-mail, або опубліковані на Web-сервері.

Утиліта консоль тривог дозволяє переглядати звіт тривог різних MPB одного проекту. Для кожного переглянутого звіту тривог створюється окреме вікно. У нього можна виводити інформацію з файлу звіту тривог, або повідомлення, що формуються MPB.

Будь-яка робоча станція системи TRACE MODE може виступати в якості Web-сервера, що дозволяє управляти технологічним процесом через Інтернет (Internet) [1]. На віддаленому комп'ютері необхідно мати тільки доступ до мережі Інтернет та Web-браузер. Для реалізації даного режиму призначений модуль Web-активатор, який використовується в якості www-шлюзу для локальних систем АСУ ТП на базі TRACE MODE, або для надання функцій Web-сервера моніторам реального часу. Використання Web-активатора дозволяє швидко перетворити існуючі АСУТП і АСУП в Internet/Intranet-системи без переробки баз даних реального часу (баз каналів).

Доступ до даних реального часу через Web-активатор здійснюється за допомогою звичайного браузера, який працює під будь-якою операційною системою, що дозволяє запуск віртуальної Java-машини. Інформація про технологічний процес представляється користувачеві у вигляді анімованих мнемосхем, трендів і таблиць [5].

Зв'язок з серверами реального часу TRACE MODE може здійснюватися практично будь-якими доступними засобами, наприклад через стільникову мережу стандарту GSM, інфрачервоний порт, мережу на основі інтерфейсу RS-232/485, або модем з використанням високонадійного протоколу TCP/IP. Можна здійснювати підключення і безпосередньо через Internet. Для цього досить увійти в Internet і набрати IP-адресу сервера TRACE MODE – підключення відбудеться автоматично.

Для доступу до даних користувачеві досить набрати Web – адреса активатора і ввести пароль, тоді весь проект завантажуватиметься у віддалений комп'ютер у вигляді Java-апрету. Використання стандартної мови Java при написанні дозволяє реалізувати на віддалених комп'ютерах не тільки Windows,

але й інші операційні системи, наприклад Unix, Linux, Mac OS і т. д., а також ОС, що використовуються в кишенькових PC. Проект TRACE MODE надходить до користувача у вигляді Java-апрету, обсяг якого не перевищує 300 Кбайт, що дає можливість використовувати Веб-активатор у мережах з низькою якістю зв'язку. Перевагою технології Java є також підвищена безпека.

При використанні Web-активатора не потрібна установка Web-серверів інших виробників (наприклад, MS IE), що вигідно відрізняє цю програму від рішень, застосованих в інших SCADA.

Для забезпечення мобільних користувачів АСУ оперативною інформацією в режимі реального часу на базі TRACE MODE розроблено програмний продукт – GSM-активатор [6]. Він призначений для дистанційного моніторингу та управління технологічними процесами, а також для отримання оперативної техніко-економічної інформації за допомогою портативних комп'ютерів handheld PC.

У реальному часі GSM-активатор може приймати інформацію від 64000 датчиків, здійснювати супервізорне управління, отримувати техніко-економічну інформацію з баз даних через сервер, який використовує стандартні інтерфейси SQL/ODBC. OPC, DDE і т. д. Вся вхідна інформація відображається графічно у вигляді анімованих мнемосхем і трендів.

GSM-активатор, який належить до нового класу систем оперативного управління, що відображають світову тенденцію до мініатюризації і автономізації комп'ютерних систем, може бути використаний в якості персональної інформаційної системи керівника.

До GSM-активатора проявляють інтерес нафтові компанії, електричні та теплові мережі РАО ЄЕС і РАО ГАЗПРОМ, комунальні та інші служби, керуючі просторово розподіленими об'єктами.

GSM-активатор придатний також до застосування в охоронних службах: отримання в реальному часі інформації про стан охоронюваного об'єкта може стати основою успіху операції групи швидкого реагування, викликаній за тривою.

Потрібно зазначити, що в останній версії TRACE MODE 6 всі редактори системи викликаються з однієї програми – інтегрованої середовища розробки (IC). IC – єдина програмна оболонка, що містить всі необхідні засоби для розробки проекту [7].

Всі змінні проекту, до чого б вони не ставилися – до контролера, до операторської станції, до управління техобслуговуванням, або виробництвом, зберігаються в єдиній базі даних проекту. Єдина база проекту усуває зайву роботу проектувальника з створення, підтримки і взаємної ув'язки багато в чому однакових баз змінних контролерів і ПК, характерну для систем попереднього покоління.

Логічна структура проекту повністю відокремлена від апаратної частини. Завдяки єдиному простору розподілених змінних, змінні з різних вузлів можуть зв'язуватися між собою також легко, як і в межах одного вузла, будь-які зміни, що вносяться в об'єкт, автоматично застосовуються скрізь, де він був задіяний.

1.2.2 Основні поняття SCADA-система TRACE MODE

Проект системи управління – це сукупність всіх математичних та графічних елементів системи, що функціонують на різних операторських станціях і контролерах однієї АСУ ТП, об'єднаних інформаційними зв'язками і єдиною системою архівування. Проект може бути масштабним (сотні вузлів), а може включати в себе тільки один контролер, або одну операторську станцію [38]. Під проектом у TRACE MODE 6 розуміється вся сукупність даних і алгоритмів функціонування розподіленої АСУ (АСУТП та/або T-FACTORY), заданих засобами TRACE MODE.

Підсумком розробки проекту є створення файлів, що містять необхідну інформацію про алгоритми роботи АСУ.

Ці файли потім розміщуються на апаратних засобах (комп'ютерах і контролерах) і виконуються під управлінням виконавчих модулів TRACE MODE.

Складова частина проекту, що розміщується на окремому комп'ютері або в контролері і виконується під керуванням одного або декількох виконавчих модулів TRACE MODE, називається вузлом проекту.

Вузол – будь-який пристрій у рамках проекту, в якому запущено програмне забезпечення TRACE MODE, що реалізує серверні функції. Це може бути контролер, операторська станція або архівна станція. У проекті не може бути більше 128 вузлів. 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.

База каналів – сукупність всіх каналів, математичних об'єктів, FBD-програм і ІЛ-програм, створених для кожного конкретного вузла.

Об'єкт бази каналів – сукупність будь-яких каналів, яким приписаний певний набір властивостей і атрибутів. Серед останніх можна назвати ім'я, графічний ідентифікатор, прапор підпорядкування: батько, нащадок [4]. Оформлені групи каналів можуть бути підпорядковані один одному і створювати таким чином ієрархічні структури.

Драйвери обміну – драйвери, використовувані моніторами TRACE MODE для взаємодії з пристроями, протоколи обміну з якими не вбудовані в монітори.

Канал (базове поняття системи) – це структура, що складається з набору змінних і процедур, що має налаштування на зовнішні дані, ідентифікатори і період перерахунку її змінних. Ідентифікаторами каналу є: ім'я, коментар та кодування. Крім того, кожен канал має числовий ідентифікатор, використовуваний всередині системи для посилення на цей канал. Серед змінних каналу виділяються чотири основних значення: вхідна (In), апаратне (A), реальне (R) і вихідний (Q). За допомогою налаштувань вхідне значення каналу зв'язується з джерелом даних, а вихідна – з приймачем.

Залежно від напрямку руху інформації, тобто від зовнішніх джерел (дані з контролерів, ЗЗО або системні змінні) в канал або навпаки, канали підрозділяються на вхідні (тип INPUT) і вихідні (тип OUTPUT).

Дані із зовнішніх пристроїв записуються в канали, дані з каналів надсилаються на зовнішні пристрої. У канали оператор заносить керуючі сигнали. Значення з каналів записуються в архіви, операторські звіти і т. п. У каналах здійснюється перетворення даних. Змінюючи значення на системних каналах, можна управляти виведеною на екран інформацією, звуковими сигналами і т. д., тобто всією системою.

Вхідне значення каналу перетвориться в апаратне, реальне і вихідне за допомогою процедур. Процедурами каналу є [5]:

- масштабування (множення і зсув);
- фільтрація (придушення піків, апертура і згладжування);
- логічна обробка (попереднє встановлення, інверсія, контроль сполучуваності);
- трансляція (виклик зовнішньої програми);
- управління (виклик зовнішньої програми).

Порядок проходження і зміст процедур може змінюватися залежно від типу каналу (вхідний або вихідний, аналоговий або дискретний). Процедура масштабування використовується тільки в каналах, працюють з аналоговими змінними. Вона включає в себе дві операції: множення і зсув. Послідовність цих операцій змінюється в залежності від типу каналу:

- у каналів типу INPUT вхідне значення множиться на заданий множник і до отриманого результату додається величина зсуву. Результат присвоюється апаратним значенням каналу;

- у каналів типу OUTPUT до апаратного значення додається величина зсуву, потім ця сума множиться на заданий множник, а результат присвоюється значенню каналу.

Процедура трансляції визначена для всіх каналів незалежно від їх типу та виду уявлення. Біля вхідних каналів процедура трансляції перетворює апаратне значення в реальне, а для вихідних – навпаки. Для цього викликається FBD-програма.

Викликана програма вибирається при налаштуванні процедури. При налаштуванні процедури вхідні і вихідні аргументи обраної програми зв'язуються з атрибутами поточного каналу, а також будь-яких інших каналів з поточної бази. Тому процедура трансляції одного каналу може також використовуватися для формування значень інших каналів.

Набір процедур у каналі залежить від формату даних. Канали працюють з аналоговими змінними, використовують наступні процедури масштабування, трансляцію, фільтрацію та управління. У каналах, що обробляють дискретні параметри, використовуються логічна обробка, трансляція та управління.

Фільтрація – процедура, яка присутня тільки в аналогових каналах. Набір виконуваних нею операцій відрізняється для вхідних і вихідних каналів. У каналів типу INPUT фільтрація виконується після процедури трансляції до формування реального значення. Фільтрація включає в себе наступні операції: придушення випадкових сплесків у тракті вимірювання; придушення малих коливань значення каналу; експоненціальне згладжування; контроль шкали – відстеження виходу реального значення каналу за встановлені межі шкали.

У каналів типу OUTPUT дана процедура формує реальне значення за вхідним значенням [5]. При цьому виконуються наступні операції: обмеження швидкості зміни реального значення; придушення малих коливань значення каналу; експоненціальне згладжування; контроль шкали – обрізання величини керуючого впливу до кордонів шкали каналу.

Підтип каналу вказує клас джерел і приймачів даних, з якими буде зв'язуватися канал. Для каналів типу INPUT підтип характеризує одержувану ними інформацію (АНАЛОГ – значення АЦП, лічений з плати ЗЗО, СИСТЕМНИЙ – стан системи, ЗВ'ЯЗОК – дані з віддалених вузлів проекту та ін.). Канали OUTPUT мають той же набір підтипів, що і канали INPUT. Однак для них підтип визначає клас приймачів, а не джерел даних (АНАЛОГ – значення ЦАП, СИСТЕМНИЙ – стан системи, ЗВ'ЯЗОК – значення керованих каналів на віддалених вузлах проекту та ін.) [7]. Всього існує шістнадцять підтипів каналів. Всі вони можуть задаватися як для вхідних, так і для вихідних каналів. Підтип каналу задає клас джерел або приймачів даних. Крім того, підтип каналу визначає також кількість його додаткових налаштувань. Уточнення джерела або приймача, в рамках заданого підтипом класу, здійснюється за допомогою доповнення до підтипу.

1.2.3 Обмін даними в SCADA-системі TRACE MODE

Якщо мова йде про зв'язок між компонентами одного вузла, то питання про апаратно-програмний інтерфейс, який повинен бути задіяний для забезпечення зв'язку, не виникає. В цьому випадку досить виконати конфігурування властивостей зв'язок/виклик компонентів. Якщо взаємодіючі компоненти належать до різних вузлів, інтерфейс зв'язку, як правило, повинен бути вказаний і налаштований.

Монітори реального часу TRACE MODE можуть обмінюватися даними за наступними лініями: локальна мережа; послідовний інтерфейс RS-232, RS-485, RS-422; радіоканал; виділена телефонна лінія; комутовані телефонні лінії; мережі GSM. За цими носіями необхідно організувати інформаційні потоки всіх рівнів системи управління. При цьому можуть реалізуватися як вертикальні зв'язки (між рівнями), так і горизонтальні (між вузлами одного рівня). Наприклад, при завданні зв'язку двох каналів різних вузлів по RS необхідно створити у вузлах компоненти COM-порт, задати для них необхідні параметри і вказати для каналу-приймача використовуваний інтерфейс зв'язку [7].

1.2.4 Типи інтерфейсів і механізми обміну

Послідовний інтерфейс. Обмін по всіх лініях, крім локальної мережі, реалізується через послідовний порт за протоколом M-Link. Вузли у мережі M-Link нерівноправні: один має статус Master, а решта – Slave. Такі мережі слід використовувати для зв'язку між операторськими станціями і контролерами. Монітор зі статусом Master є активним. Він посилає команди управління і запиту на передачу інформації.

Монітор зі статусом Slave приймає надіслані йому команди і передає запитані дані. Команди управління містять вказівки на зміну значень атрибутів каналів віддаленого вузла [89].

Таким чином, запити, що посилаються монітором зі статусом Master, можуть бути двох типів:

1) запит даних (використовується для отримання значень каналів або іншої інформації від монітора зі статусом Slave);

2) запит на зміну (використовується для зміни значень атрибутів каналів на віддаленому моніторі). У запитах на зміну передаються нові значення коректованих атрибутів віддаленої бази.

Слід зазначити, що в одній мережі M-Link не може бути двох моніторів, для яких встановлено статус Master. Щоб один монітор виступав і як Master, і як Slave, треба створити паралельні мережі, використовуючи при цьому по два послідовних порти на кожному вузлі. Тоді два монітори зможуть працювати в режимі Master.

Обмін за протоколом M-Link. Для обміну даними між моніторами TRACE MODE по послідовному інтерфейсу використовується протокол M-Link. Він застосовується для обміну по інтерфейсах RS-232, RS-485, RS-422, радіоканалу, комутованих телефонних лініях і GSM мережі.

Використовуючи протокол M-Link, у рамках TRACE MODE можна створювати мережні комплекси на базі послідовного інтерфейсу RS-485. Такі комплекси можуть включати в себе до 128 вузлів (контролерів і операторських станцій). При цьому зв'язок може здійснюватися за декількома послідовними портами. Для зв'язку двох моніторів можна використовувати інтерфейс RS-232. Щоб зв'язатися з кількома віддаленими вузлами за цим інтерфейсом, потрібно мати відповідну кількість послідовних портів. Це дозволяє організувати зв'язок типу "Зірка". Така конфігурація може зажадати додаткових витрат на багатоканальні плати [90]. Однак вона дозволяє швидше передавати дані за рахунок розподілення обміну з різними віддаленими вузлами. TRACE MODE підтримує обмін одночасно по 32 послідовних портах.

Для зв'язку сильно рознесених у просторі моніторів можна використовувати радіоканал, виділені або комутовані телефонні лінії. У цих випадках потрібні додаткові пристрої – модеми. Вони узгодять електричні характеристики послідовних портів і використовуваного середовища передачі.

1.3 Розробка програмного забезпечення в SCADA системі Trace Mode

1. Розробка головного екрана:

Основною вимогою для розробки було зрозумілість у першу чергу для кінцевого користувача АСУТП – оператора. Для створення інтерфейсу використано стандартні бібліотеки Trace Mode.

Розроблена програма на основі SCADA системи виконує наступні функції:

- збір та реєстрація первинної інформації про хід технологічного процесу;
- подання інформації у вигляді мнемосхеми;
- ведення та перегляд історії технологічного процесу;
- формування звітної документації;
- механізм налаштування прав користувачів, рівні доступу;
- режим автоматичного та ручного керування.

Аналізуючи функції, можна зробити висновок, що SCADA система збирає інформацію про технологічний процес, забезпечує інтерфейс з оператором та здійснює автоматичне керування процесом.

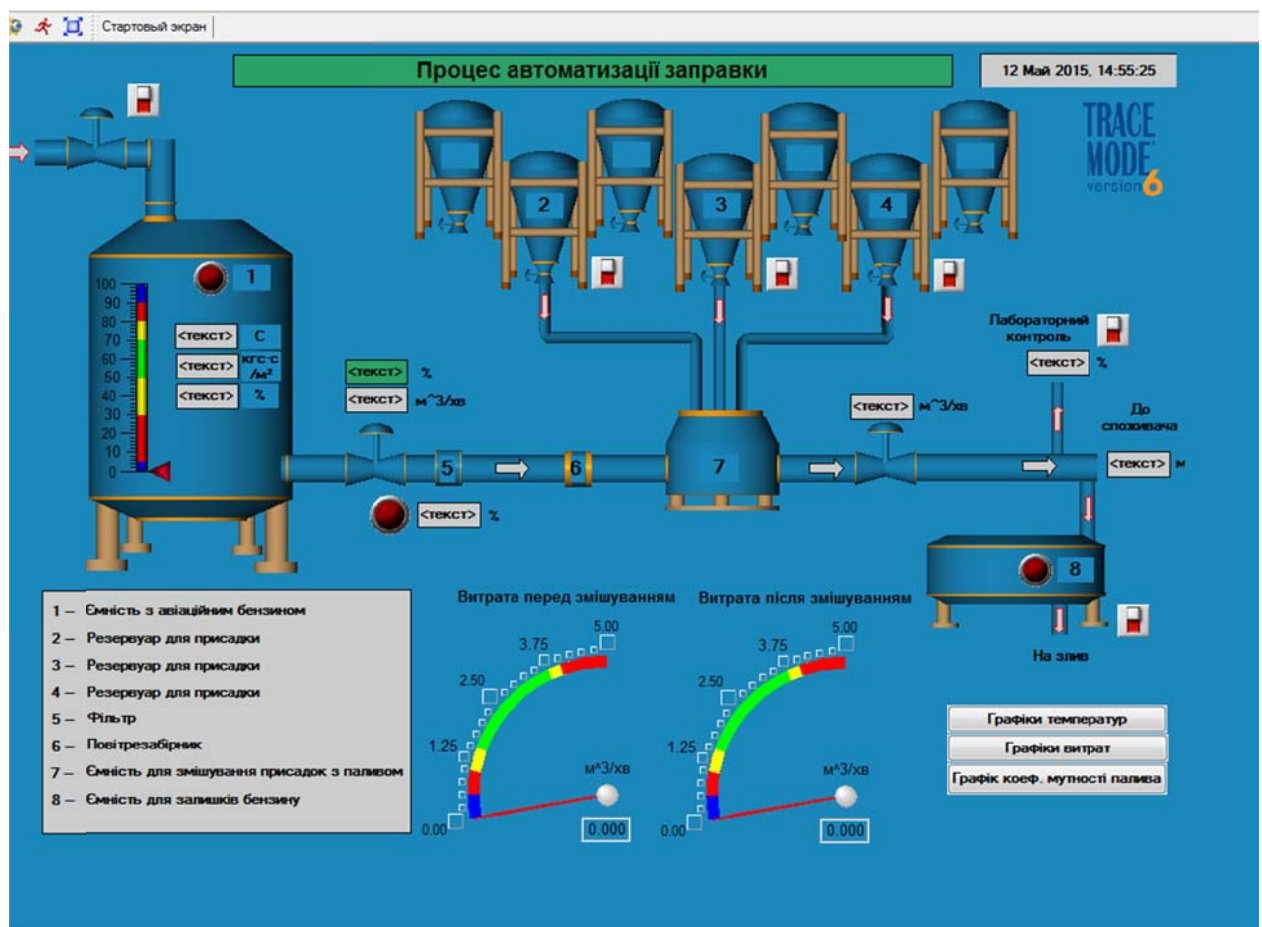


Рисунок 1.2 – Головний екран SCADA системи

Цей інтерфейс оператора реалізує наступні функції:

- моніторинг поточних параметрів технологічного процесу;
- зміна регулятора в автоматичному режимі керування;
- переключення в ручний режим керування;
- звіти про роботу технологічного процесу.

На головному екрані оператор може проконтролювати весь процес. При подачі палива оператором в ємність об'ємом 5000 тонн на ній починають працювати індикатори, які показують: рівень заповнення ємності, якість палива, температуру та швидкість подачі.

При проходженні палива по трубопроводу, воно очищується за допомогою фільтра і показники, що характеризують стан фільтра, виводяться на екран. Коли фільтр забруднюється, вмикається індикатор-лампа, яка подає сигнал червоним кольором. Також на схемі можна побачити рівень забрудненості фільтра. Потім паливо проходить через повітрязабірник та насичується присадками, якщо вони потрібні, оператор у ручному режимі вмикає ємності та додає їх до палива (присадки додаються в залежності від погодних умов, виду повітряного судна та ін.). Витрату палива до і після змішування можна відслідкувати на головному екрані. Експреслабораторний контроль показує мутність палива після змішування з присадками.

При переході на допоміжні екрани даних, оператор може переглянути всі дані про якість палива, температуру процесу, та витрату бензину. Також оператор має змогу із цього екрана створити автоматичний звіт про роботу системи та роздрукувати його.

При переході з головного екрана на вкладки: графік температур, графік витрати, графік коефіцієнта мутності палива відкриваються інші екрани, що відповідають екранам 2, 3, 4. Рисунок надані відповідно.

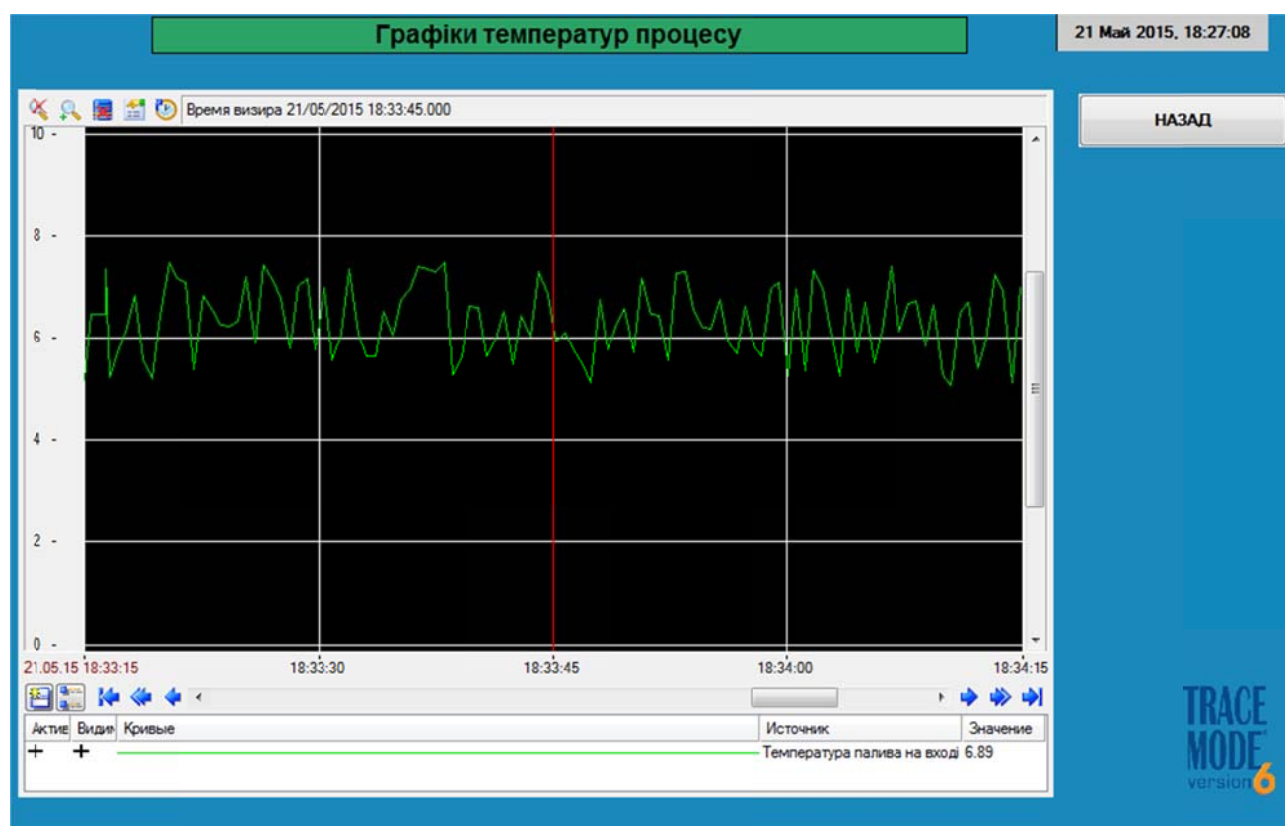


Рисунок 1.3 – Экран 2: графік температури процесу

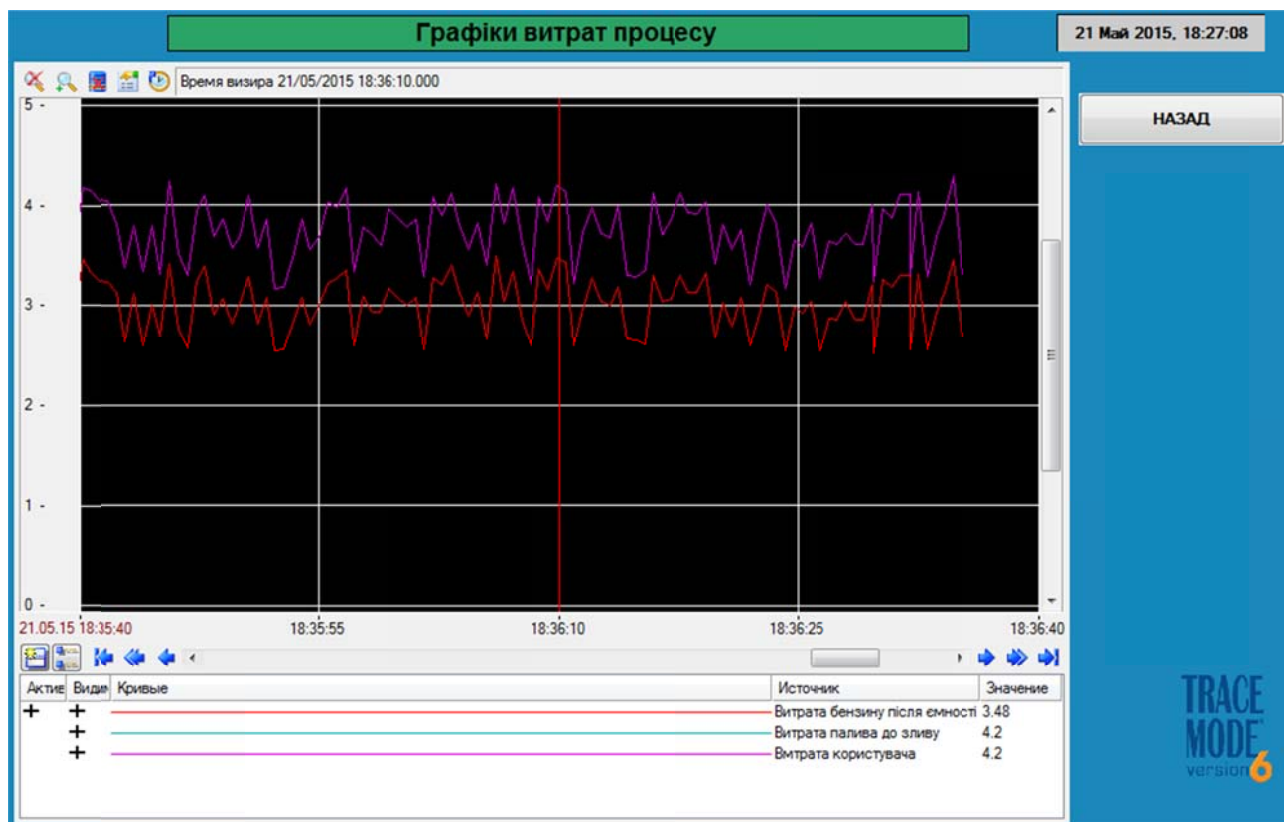


Рисунок 1.4 – Экран 3: графік витрати процесу



Рисунок 1.5 – Экран 4: графік ступеня якості палива

1.4 Розробка програми на мові FBD у SCADA системі Trace Mode

Для перевірки роботоздатності, розробленої відповідно до завдання, системи управління заправною системою виконуються моделюванням в програмному середовищі TRACE MODE.

Для роботи та тестування системи керування написані програми, які були реалізовані за допомогою FBD блоків, що дозволяють виконати емуляцію роботи. Фрагмент лістингу програми «витрата палива» емуляції наведений нижче на рисунку 1.6.

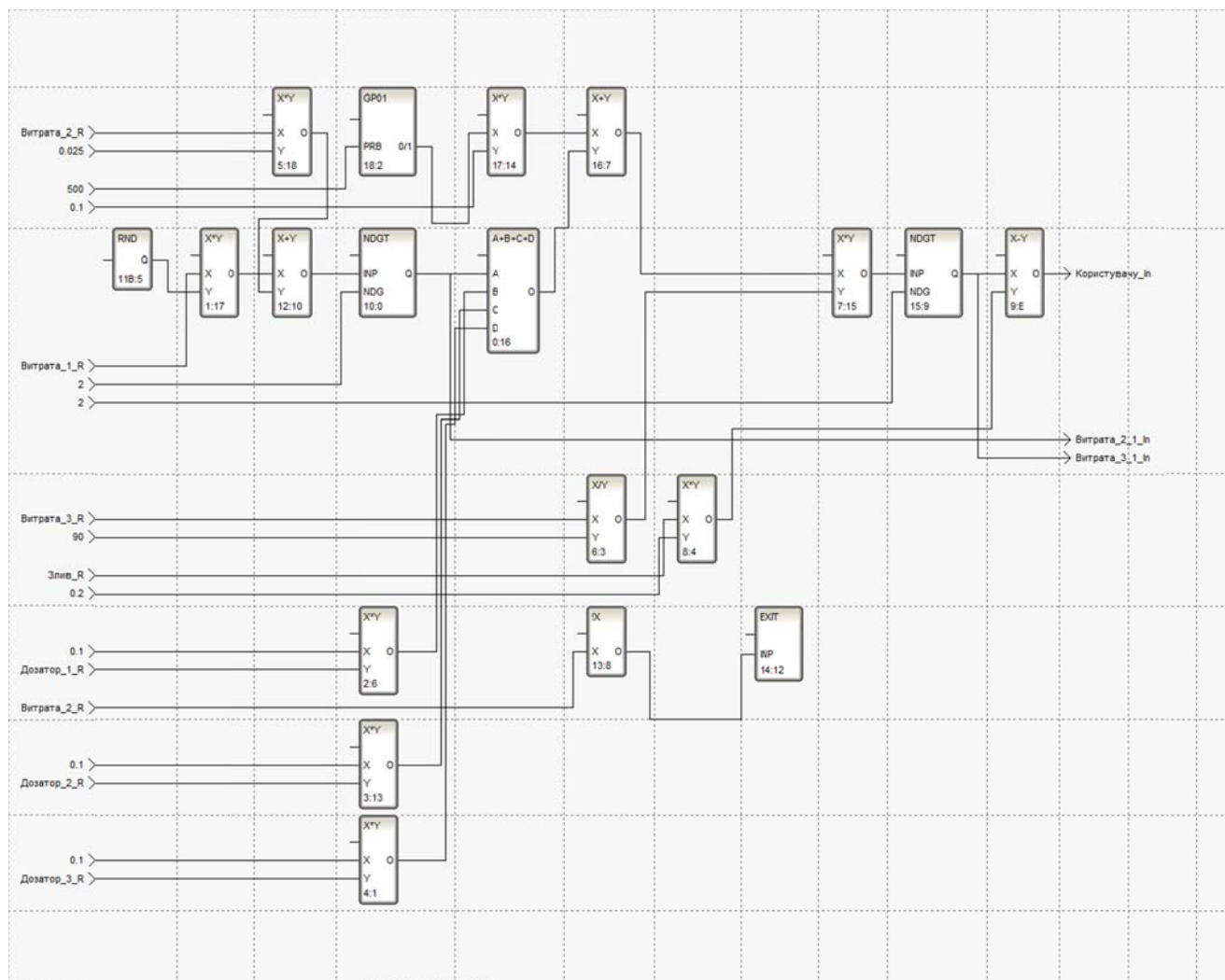


Рисунок 1.6 – Програма для реалізації процесу витрати палива

Розроблена внутрішня логіка системи управління описана з використанням вбудованої в редактор бази каналів, мови функціональних блоків (FBD). Для запуску мови функціональних блоків (FBD) необхідно в меню «навігаторі проекту» натиснути двічі на створену програму. З правого боку від навігатора з'являється нова вкладка редагування створеної програми. Щоб обрати мову програмування, в цьому вікні потрібно

натиснути в структурі програми на підрозділ «Програма» лівою кнопкою миші, відкриється вікно «Вибір мови програмування».

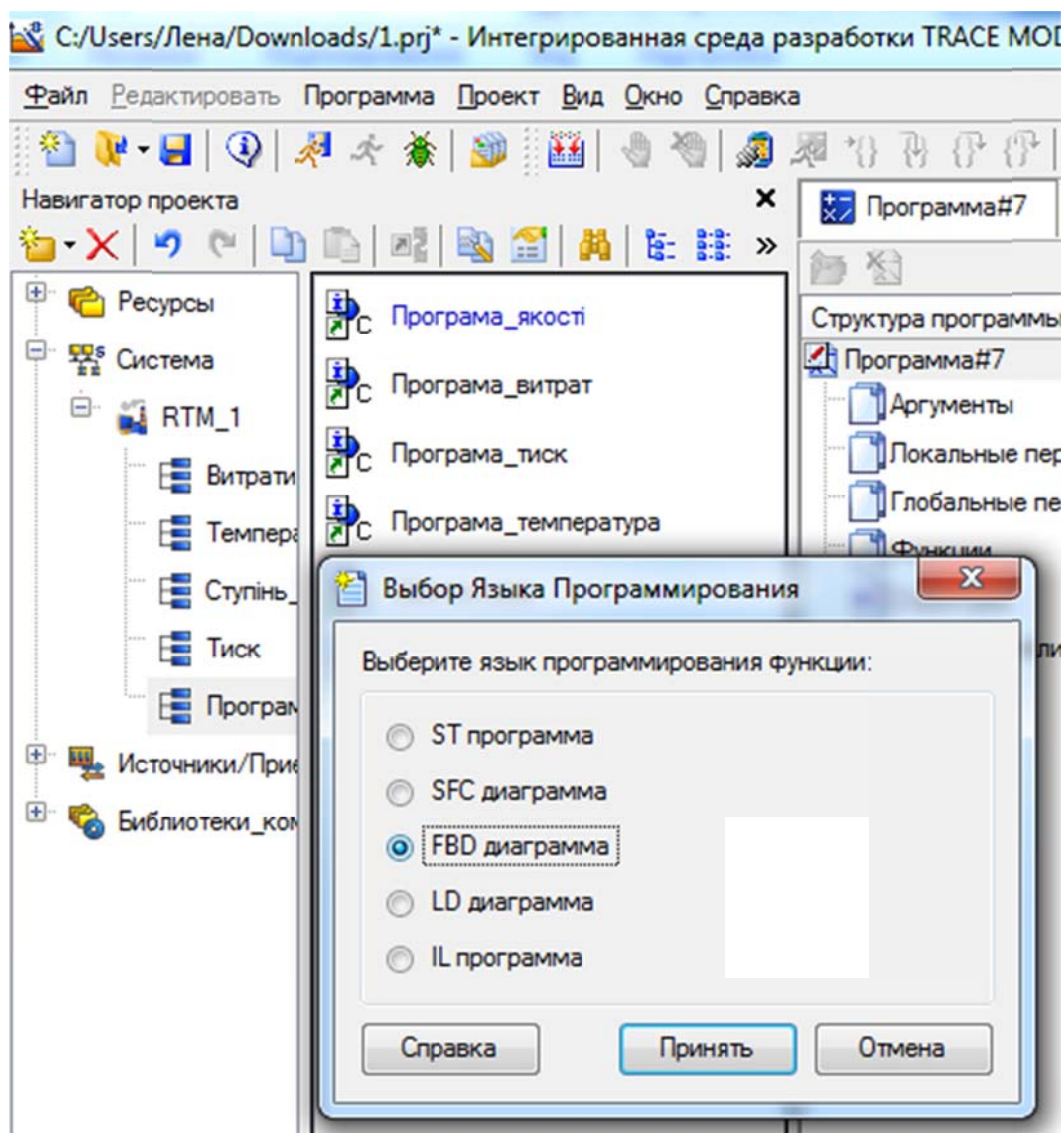


Рисунок 1.7 – Запуск мови функціональних блоків

Для установки логічних елементів (блоків) необхідно увійти в режим установки функціональних блоків, натисканням кнопки  в меню FBD редактора бази каналів. Після вибору необхідного блока, натискаємо відповідну йому кнопку в меню FBD редактора бази каналів.

При написанні даної програми були використані стандартні блоки, робота яких описана нижче в таблиці 1.2.

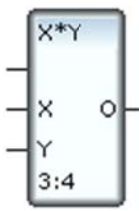
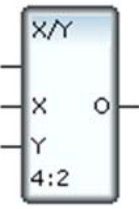
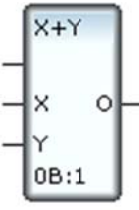
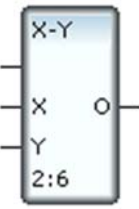
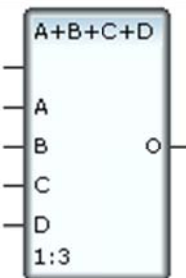
Зазначимо, що при реальному використанні програмного забезпечення дані з об'єкта будуть надходити в систему керування через драйвери технічних засобів, які можна з легкістю підключити з бібліотеки пристроїв Trace Mode.

Для роботи системи в автоматичному режимі була написана програма на мові FBD, для визначення температури процесу, частина цієї програми зображена на рисунку 1.8.

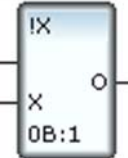
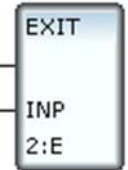
У програмі були використані такі ж блоки, які раніше були описані.

Для контролю якості палива написана програма за допомогою FBD-блоків. Дана програма зображена на рисунку 1.9.

Таблиця 1.2 – Опис FBD блоків для реалізації процесу «витрата палива»

Графічне зображення	Назва блока та скорочений опис
1	2
	Множення ($X * Y$) Блок множення величин. Так як таких блоків декілька, витрата помножується на конкретну величину, або на блок, у залежності від його розташування. Більш детальну інформацію можна отримати при запуску програми.
	Ділення (X / Y). Блок ділення величин (блоків). Протилежний блок до множення. Таких блоків у програмі використовується декілька, тому інформацію про взаємодію можна побачити на схемі програми, або при запуску програми.
	Додавання двох елементів ($X + Y$). Блок додавання величин. Так як цей блок використовується декілька разів, то більш детальну інформацію, з якими блоками він поєднується, потрібно розглядати в кожному випадку окремо.
	Віднімання двох елементів ($X - Y$). Блок віднімання величин. Блок належить до підкласу арифметичних, так як і раніше розглянуті блоки. Реалізація операції віднімання. Використовується для розрахунку розузгодження регулювання – різниці між заданим (бажаним) і реальним значенням технологічного параметра.
	Додавання чотирьох елементів ($A + B + C + D$). Блок додавання величин. Застосовується, коли потрібно підсумувати чотири величини.

Продовження таблиці 1.2

	Випадкова величина в діапазоні [0,1]. RND – блок. Цей блок генерує випадкову величину з рівномірним законом розподілення в діапазоні [0,1].
	НЕ (!X). $O = \text{NOT } X$. $O = 1$, якщо $X = 0$, у всіх інших випадках $O = 0$. Належить до логічних блоків.
	Одиниця із заданою вірогідністю. GP01 – блок. На виході цього функціонального блока генерується 0 або 1, причому вірогідність генерації одиниці задається значенням входу PRB. На вхід PRB подається ціле число в діапазоні від 0 до 1000. Цим границям відповідає значення випадковості 0 та 1. Блок належить до підрозділу генератори.
	Округлення величини. NDGT – блок. Належить до алгебраїчних блоків. На виході Q формується округлене значення входу INP. Значення входу NDG задає точність округлення: – 2 до сотень, – 1 до десятків, 0 до цілих чисел, 1 до першого розряду після коми, 2 до другого розряду після коми, 3 до третього розряду після коми, 4 до четвертого розряду після коми.
	Вихід з програми. EXIT – блок. Даний блок дозволяє вийти з програми до перерахунку всіх її функціональних блоків. Вихід проводиться при наявності значення на вході INP.

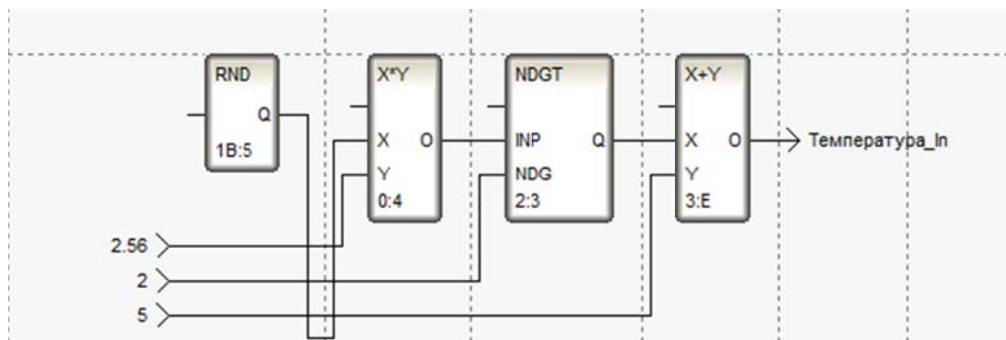


Рисунок 1.8 – Програма для визначення температури процесу

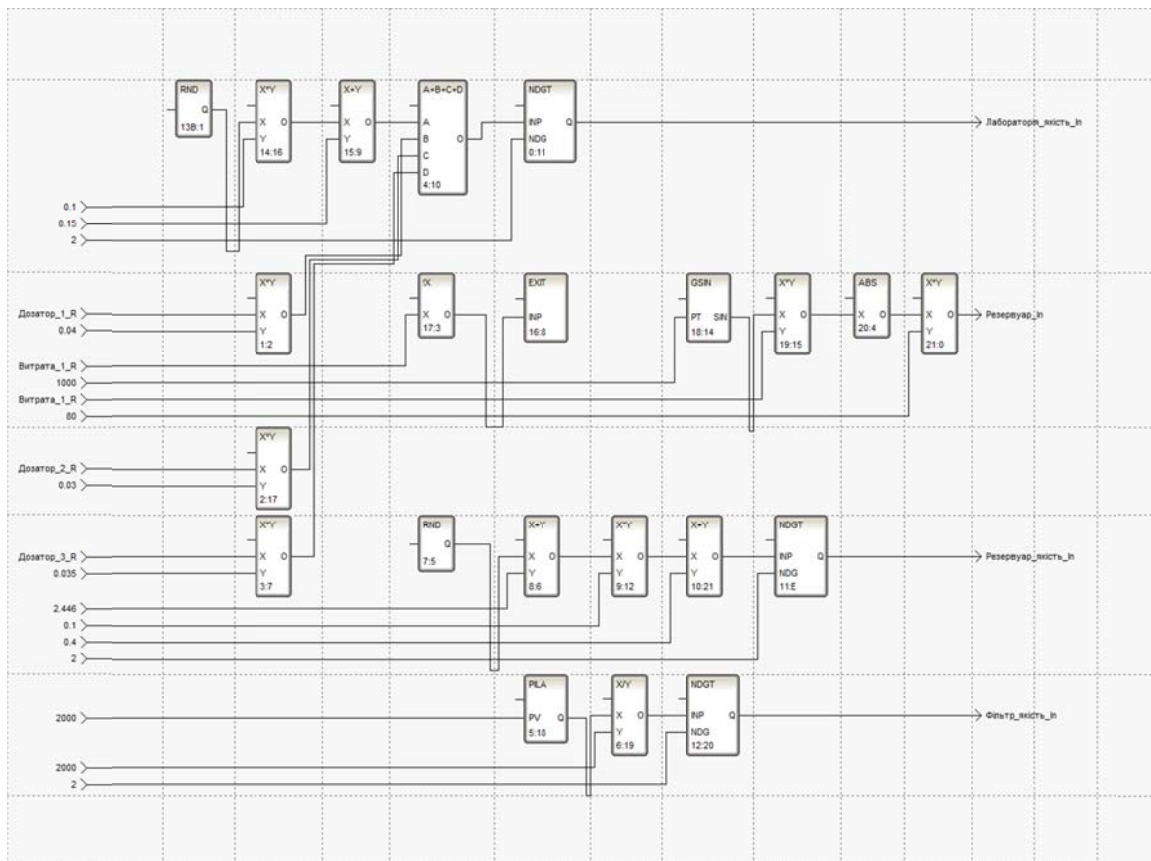


Рисунок 1.9 – Програма для визначення якості палива

В роботі використовуються декілька відмінних стандартних блоків Trase Mode від блоків попередньої програми «якість палива», описаних нижче в таблиці 1.3.

Для контролю тиску в системі написана програма за допомогою FBD-блоків. Дана програма зображена на рисунку 1.10.

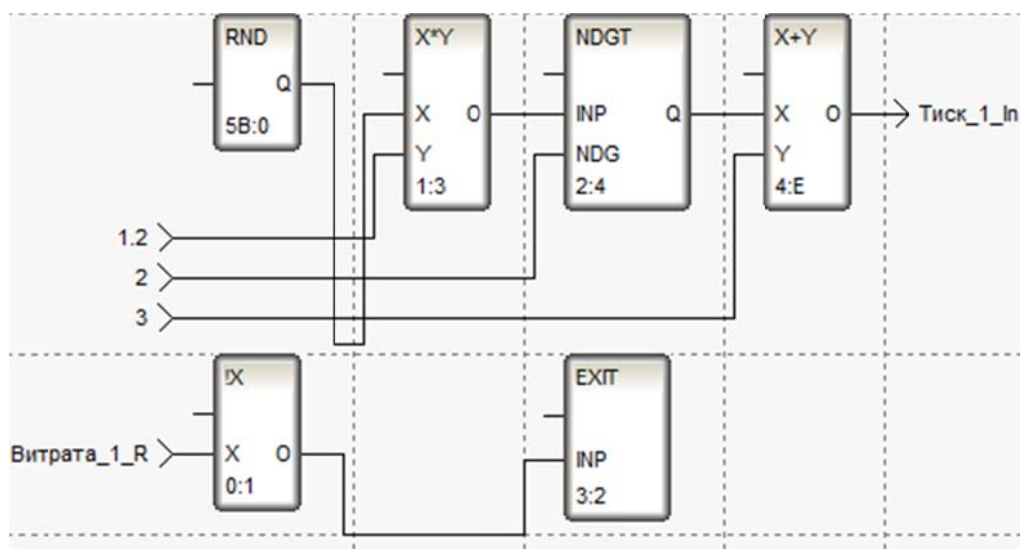
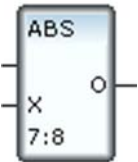
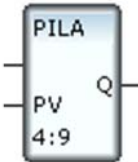
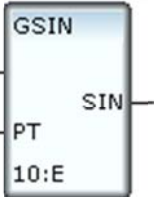


Рисунок 1.10 – Програма для визначення тиску в системі

У програмі для визначення тиску системи були використані стандартні блоки, які вже описані раніше.

Таблиця 1.3 – Описання FBD блоків для реалізації процесу «якість палива»

Графічне зображення	Назва блока та скорочений опис
1	2
	Абсолютне значення. ABS – блок. $O = X $ Належить даний блок до арифметичних.
	Пилоподібний сигнал. PILA – блок. Даний блок генерує пилоподібний сигнал з максимальним значенням, який задається входом PV. На кожному такті перерахунку вихід збільшується на 1. Даний блок належить до підрозділу генераторів.
	Синусоїдальний сигнал. GSIN – блок. Цей функціональний блок генерує синусоїдальний сигнал одиничної амплітуди. Період колювання (в секундах) задається значенням PT.

1.5 Подання даних

У системі Trace Mode подання даних відбувається різними способами. Редактор подання даних – інструмент розробки екранних форм операторського інтерфейсу SCADA/HMI. У ньому для кожного вузла проекту АСУТП створюються графічні екрани, формуються статичні мнемосхеми процесу, здійснюється їх динамізація шляхом векторної і AV-анімації. Створюються віртуальні органи управління, тренди реального часу та історичні, формуються звіти тривоги. Здійснюється вбудовування ActiveX-компонентів.

Редактор подання даних SCADA TRACE MODE містить графічні бібліотеки з більш ніж 600 технологічними об'єктами АСУТП (як статичними, так й анімованими), готовими до використання в прикладних проектах. Серед них: панелі управління, прилади, теплообмінники, ємності, сепаратори, колони, насоси, клапани, електричні елементи і т. д. Загальна кількість готових графічних об'єктів SCADA TRACE MODE перевищує 1000, що є одним з кращих показників у галузі.

У редакторі подання даних був розроблений графічний інтерфейс користувача, пов'язаний зі структурною моделлю системи управління заправної станцією за допомогою інформаційних каналів.

Для запуску редактора подання даних у меню «Пуск» ПК вибираємо пункт «TRACE MODE-6», потім пункт «Редактор подання даних».

У редакторі подання даних необхідно відкрити проект. Для цього в меню «Проект» вибираємо пункт «Відкрити», де необхідно вказати назву файлу проекту і вказати шлях до нього.

Після появи в лівому верхньому кутку вікна з навігацією проекту двома, активуємо в системі блок RTM_1, натиснувши на нього 1 раз правою кнопкою миші. При цьому RTM_1 стає синім (рис. 1.11).

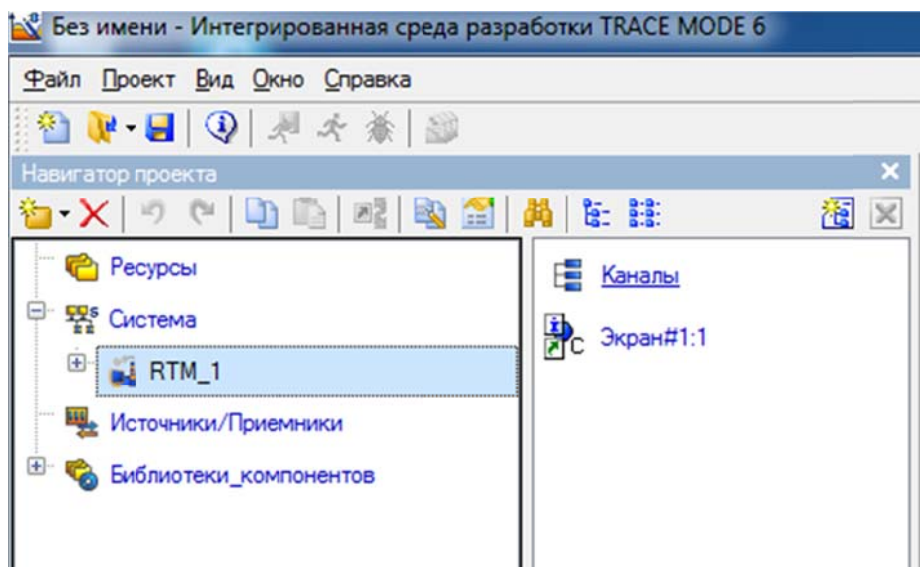


Рисунок 1.11 – Відкриття проекту в редакторі подання даних

На отриманому екрані розміщені динамічні та статичні елементи, які використовуються для візуалізації проекту.

У вузлі RTM_1 створимо групу каналів та дамо їй назву «Програми». Додамо до групи компонент, для цього натиснемо лівою кнопкою миші на створену групу та оберемо компонент – «Програма». Таких програм у групі «Канали» створимо 4. Перейменуємо їх на: «Програма_якості», «Програма_витрат», «Програма_тиск», «Програма_температури».

Розглянемо приклад побудови FBD-блоків на прикладі «Програми_якості». В «навігаторі проекту» натискаємо двічі на створену програму. З правого боку від навігатора з'являється нова вкладка редагування створеної програми. Щоб обрати мову програмування, в цьому вікні потрібно натиснути в структурі програми на підрозділ «Програма» лівою кнопкою миші, відкриється вікно «Вибір мови програмування» (рис. 1.12).

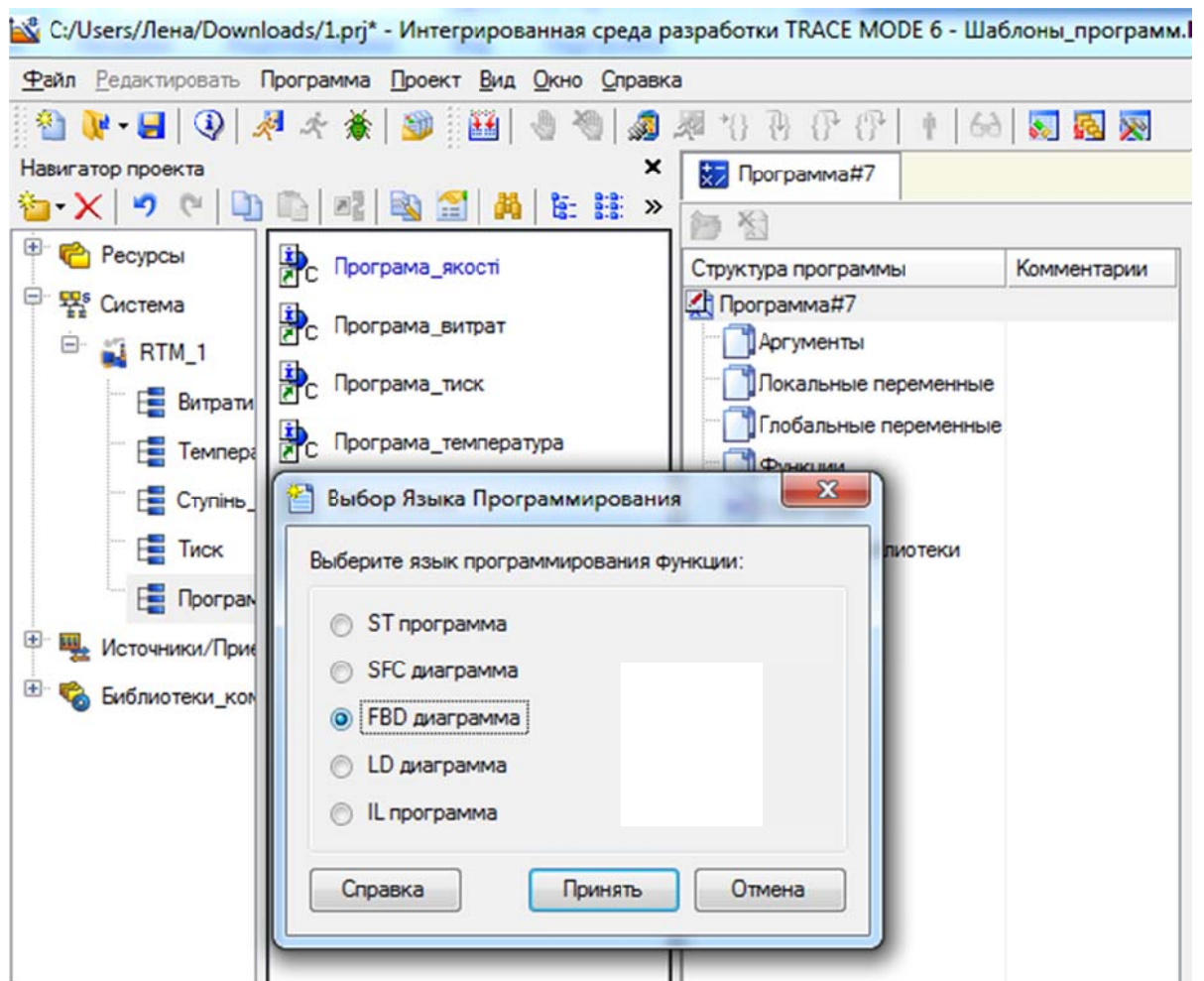


Рисунок 1.12 – Вибір мови програмування в програмному середовищі TRACE MODE

Обираємо мову «FBD-діаграма». Після чого додамо аргументи програми. У вікні «структура програми» натискаємо на вкладку «Аргументи», з права з'являється таблиця для введення аргументів, яка складається з колонок: ім'я, тип, тип даних, прив'язка. Заповнюємо таблицю та робимо прив'язки до раніше заготовлених каналів «Витрати» (рис. 1.13).

Имя	Тип	Тип данных	Значение по умолчанию	Привязка
Витрата_1_R	IN	REAL		Витрата_1:Реальное значение (Сист
Дозатор_1_R	IN	REAL		Дозатор_1:Реальное значение (Сист
Дозатор_2_R	IN	REAL		Дозатор_2:Реальное значение (Сист
Дозатор_3_R	IN	REAL		Дозатор_3:Реальное значение (Сист
Фильтр_якість_In	OUT	REAL		Фильтр_якість:Входное значение (С
Резервуар_якість_In	OUT	REAL		Резервуар_якість:Входное значение
Лаборатория_якість_In	OUT	REAL		Лаборатория_якість:Входное значен
Резервуар_In	OUT	REAL		Резервуар:Входное значение (Систе

Рисунок 1.13 – Зразок заповнення таблиці аргументів на прикладі «Програми_якості»

Далі будуюмо діаграму з блоків – FBD. Для цього в розділі «структура програми» натискаємо правою кнопкою миші на вкладку «Програма». Відразу відкривається сітка для побудови програми з блоків, а з правого боку TRACE MODE з'являються FBD-блоки, які розділені на групи, що спрощує їх пошук (рис. 1.14). Створюємо програму з тих блоків, які були описані раніше.

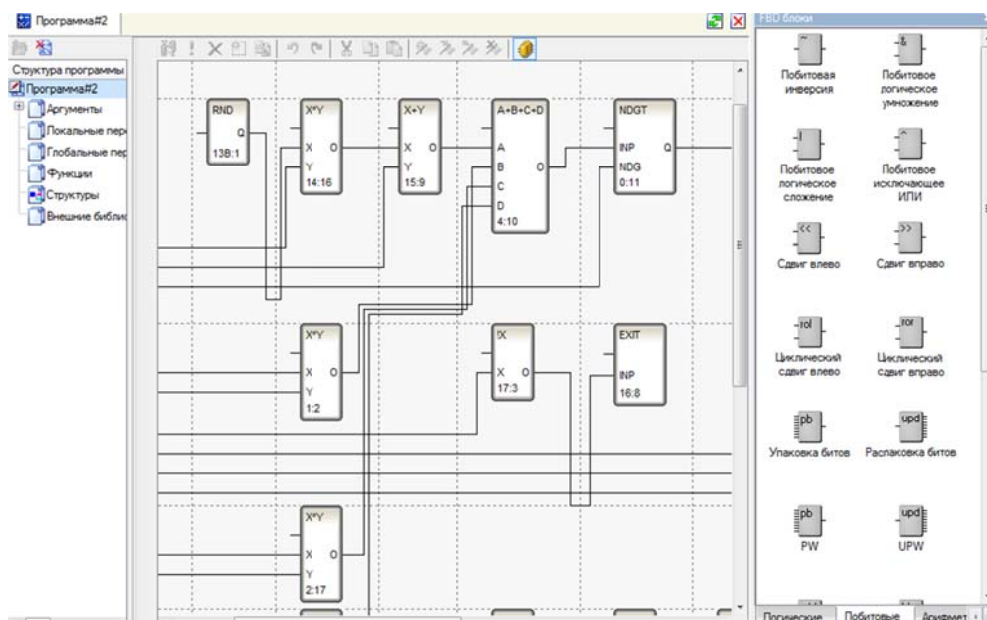


Рисунок 1.14 – Створення програми за допомогою FBD-блоків на прикладі «Програми_якості»

Повний лістинг програми наведено на рисунку 1.6.

Таким чином розробляємо FBD-діаграму та заповнюємо таблицю «Аргументи» для програм: «Програма_витрат», «Програма_тиск», «Програма_температури».

Для виконання графічної частини проекту були використані наступні динамічні елементи:

- кнопки (для завдання поточного значення каналів);
- динамічний текст (для відстежування поточного значення сигналу в каналах);
- кнопки (для завдання поточного значення каналів);
- динамічний текст (для відстежування поточного значення сигналу в каналах);
- гістограми (для реалізації в графічному вигляді сегментів цифрового індикатора).

Для надання даних проекту були створені декілька екранів роботи процесу:

- головний екран являє собою саму централізовану систему в цілому (рис. 1.2), також на ньому зображено стрілкові пристрої (рис. 1.15), які показують витрату бензину до та після змішування, що допомагає оператору

при заправці літака чітко бачити скільки бензину заливається. Подання даних про паливо в ємності забезпечується повзунком (рис. 1.16), приладом, який показує на скільки відсотків заповнена ємність. Для виводу даних про температуру, мутність та витрату використовується графічний елемент Текст (рис. 1.16). Про забруднення фільтра та злив палива з ємності 8, сповіщає графічний елемент «Вимикач».

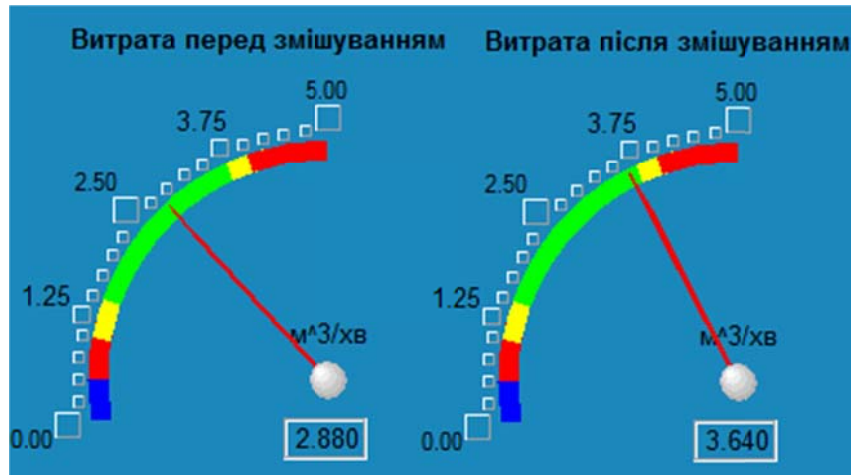


Рисунок 1.15 – Приклад подання графічного елемента «Стрілкові пристрої»

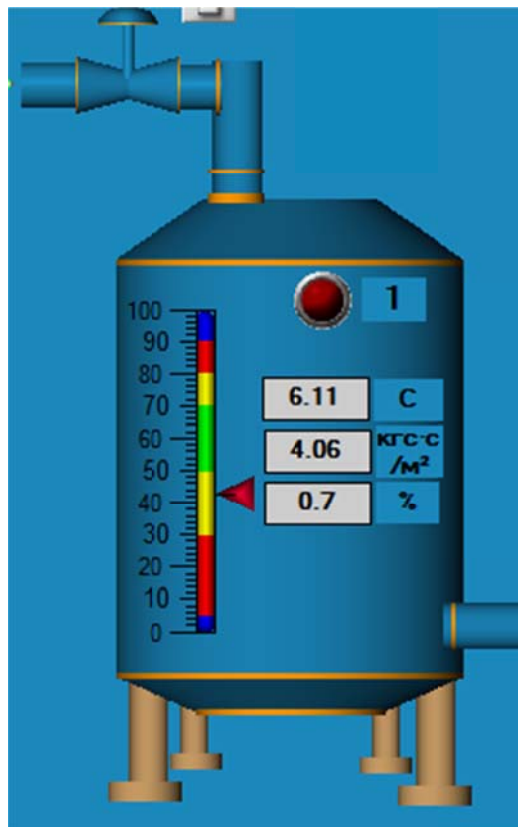


Рисунок 1.16 – Приклад подання даних графічними елементами «Повзунок» та «Текст»

– на екранах № 2, № 3, № 4 (рис. 1.3, 1.4, 1.5 відповідно) подання даних відбувається за допомогою графічного елемента «тренд». Приклад подання

зображений на рисунку 1.17. На ньому можна збудувати криві процесів та порівнювати отримані значення.

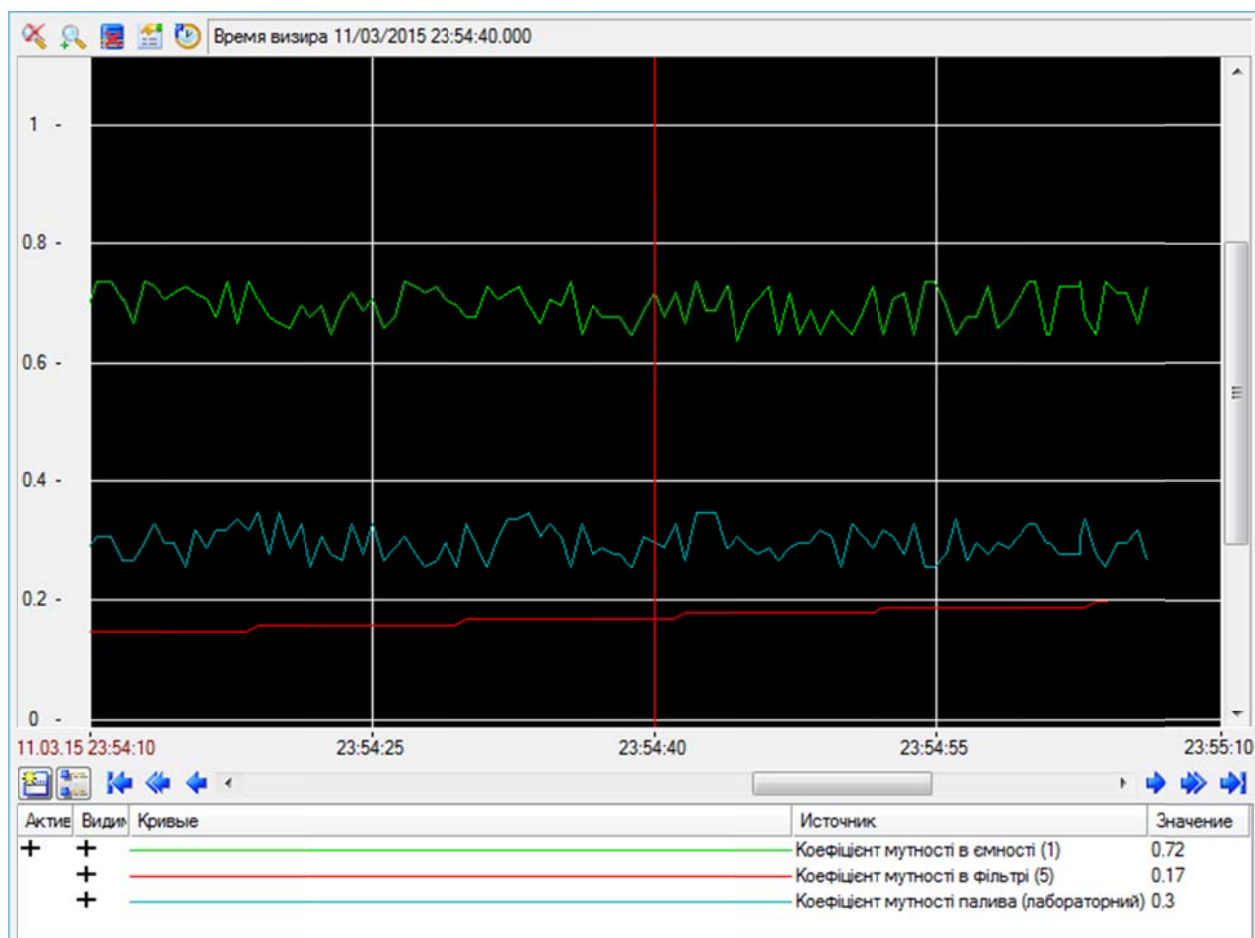


Рисунок 1.17 – Приклад подання даних графічним елементом «Тренд»

TRACE MODE дозволяє автоматично генерувати повну звітну документацію про проект АСУ ТП, не вимагаючи від користувача ніяких додаткових зусиль.

Автоматичне документування є основою для оформлення опису АСУ ТП на базі TRACE MODE 6 відповідно до стандартів ГОСТ 24.207-80, ГОСТ 19.402-78 та ГОСТ 19.502-78.

Автоматичне документування проекту TRACE MODE 6 дозволяє істотно полегшити працю автоматизації праці розробника АСУ ТП. Для будь-якого проекту TRACE MODE автоматично генерує його повний опис у вигляді гіпертекстового документа формату HTML, який можна вивести до друку, або експортувати в MS Word.

Для документування проекту АСУ ТП досить завантажити його в інтегроване середовище розробки TRACE MODE 6 IDE і в меню Файл скористатися пунктом «Документувати проект». Процес документування може зайняти від декількох секунд до декількох хвилин у залежності від розміру проекту АСУ ТП.

Опис містить як загальні відомості про проект АСУ ТП, так і його докладний зміст у вигляді таблиць, тексту і, за потребою, рисунків. Наприклад, для кожного шаблону програми опис включає:

- ім'я шаблону програми;
- кодування шаблону;
- коментар до шаблону;
- перелік аргументів шаблону із зазначенням значень за замовчуванням, прив'язок, прапорів та ін.;
- список викликів даної програми із зазначенням їх розміщення;
- повний текст програми (а в разі використання графічної мови – та її графічне надання).

Аналогічно описані й інші шаблони проекту АСУ ТП.

Для кожного шаблону графічного екрана автоматично створюється графічна ілюстрація. Опис проекту забезпечено зручними гіперпосиланнями, які спрощують пошук інформації. Наприклад, натиснувши на "Шаблони програм" можна перейти до списку програм. У свою чергу, за натисканням по назві шаблону програми можна перейти до його докладного опису. Те ж саме справедливе і для екранів, зв'язків з СУБД та інших елементів проекту. При цьому користувачеві доступні всі переваги гіпертексту: закладки, пошук. Наприклад, можна за допомогою пошуку знайти всі випадки лінковки на даний канал, всі виклики конкретної програми і т. д.

Прив'язки каналів до аргументів шаблону і список викликів даного шаблону – це теж гіперпосилання, що дозволяють перейти до опису відповідного каналу. З іншого боку опис каналу може бути викликаний з опису відповідного вузла. Кожна група каналів документована у вигляді списку гіперпосилань на канали і вкладені групи. А з опису каналу можна перейти на опис шаблону, який він викликає.

Автоматичне документування проекту TRACE MODE 6 – це зручна інтерактивна система, що дозволяє створювати документи, готові для передачі замовнику АСУ ТП.

Автодокументування – економить час розробника, скорочує число помилок, підвищує продуктивність праці і позбавляє інженера від рутинної роботи, дозволяючи концентруватися на головному – як і функціональності АСУ.

2. РЕАЛІЗАЦІЯ ДІАГНОСТУВАННЯ ЗА ДОПОМОГОЮ SCADA TRACE MODE

Для системи функціонального діагностування, СК заправною системою була реалізована у програмному середовищі TRACE MODE.

Розроблено інтерфейс системи функціонального діагностування. Для запуску діагностування на головному екрані передбачено додаткові клавіші діагностики (рис. 2.1), які знаходяться в нижньому лівому куті.

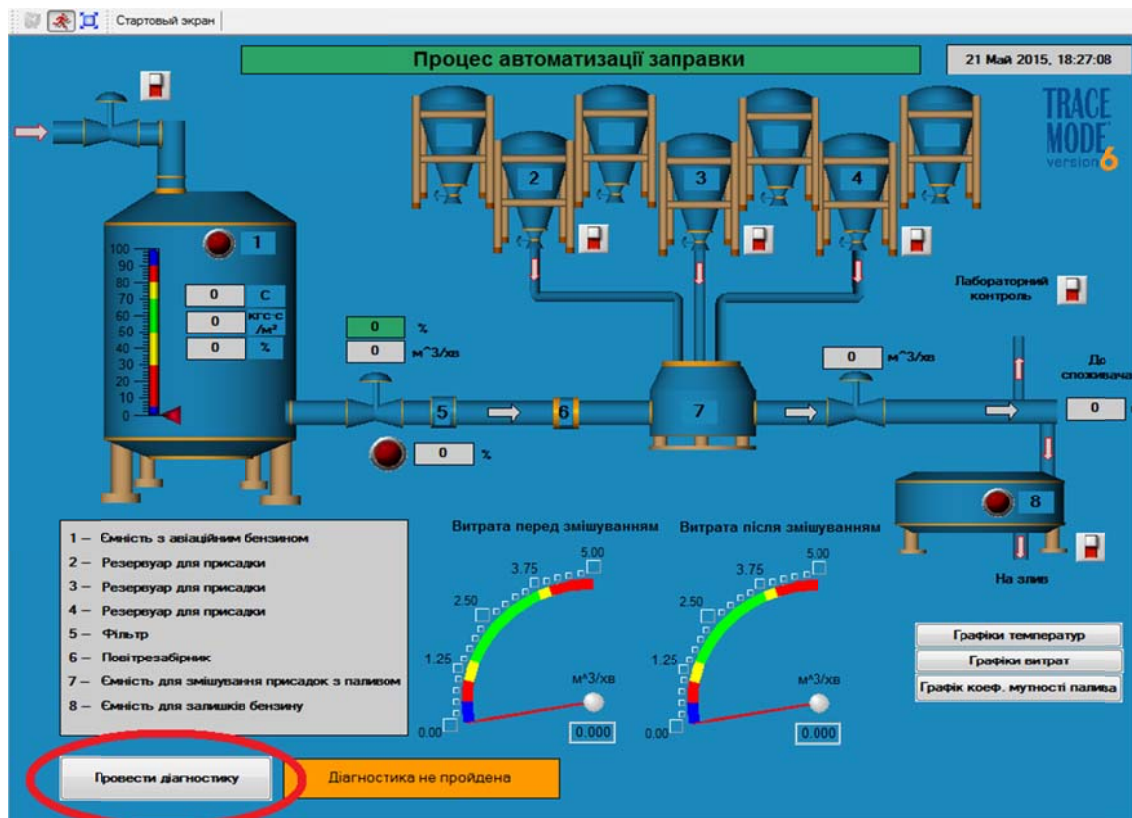


Рисунок 2.1 – Головний екран: кнопки діагностики

```

PROGRAM
VAR_INPUT T1 : REAL; END_VAR
VAR_OUTPUT T1_out : REAL; END_VAR
VAR_OUTPUT cons : REAL; END_VAR
VAR_INPUT запуск : REAL; END_VAR
VAR_OUTPUT error : REAL; END_VAR
VAR_INPUT Витрата_2_R : REAL; END_VAR
VAR_INPUT Дозатор_1_R : REAL; END_VAR
VAR_INPUT Дозатор_2_R : REAL; END_VAR
VAR_INPUT Дозатор_3_R : REAL; END_VAR
VAR_INPUT T2 : REAL; END_VAR
VAR_OUTPUT T2_out : REAL; END_VAR
VAR_INPUT T3 : REAL; END_VAR
VAR_OUTPUT T3_out : REAL; END_VAR
VAR_INPUT T4 : REAL; END_VAR
VAR_OUTPUT T4_out : REAL; END_VAR
VAR_INPUT T5 : REAL; END_VAR
VAR_OUTPUT T5_out : REAL; END_VAR
VAR_INPUT T6 : REAL; END_VAR
VAR_OUTPUT T6_out : REAL; END_VAR
VAR_INPUT T7 : REAL; END_VAR
VAR_OUTPUT T7_out : REAL; END_VAR
VAR_OUTPUT Діагностика_C : REAL; END_VAR

if запуск == 1 then
    T1_out:=T1_out-Check_Calc()/1000;
    if T1_out<= 0 then
        cons = 1;
        запуск = 2;
        error=0;
    end if;
end if;

```

Рисунок 2.2 – Частковий лістинг програми для реалізації функціональної діагностики

Так як ми продовжуємо роботу з проектом, у нас певна кількість програм описана раніше. Додамо до групи ще одну програму, для цього натискаємо правою кнопкою миші на канал «Програми» та створюємо компонент «Програма». Перейменуємо цей канал у «Діагностика». В «навігаторі проекту» натискаємо двічі на створену програму, з правого боку від навігатора з'являється нова вкладка редагування створеної програми.

Функціонування даного діагностування відбувається за допомогою мови ST, що забезпечує повну діагностику системи за алгоритмом, зображеним на рис. 2.1.

Частковий лістинг програми функціонального діагностування зображено на рисунку 2.2.

СПИСОК ЛІТЕРАТУРИ

1. Жан М. Рабаи, Ананта Чандракасан, Боривож Николич. Цифровые интегральные схемы. Методология проектирования = Digital Integrated Circuits / Жан М. Рабаи, Ананта Чандракасан, Боривож Николич. – 2-ое изд. – М.: Вильямс, 2007. – 912 с.
2. Островский Г.М., Волин Ю.М. Технические системы в условиях неопределенности. Анализ гибкости и оптимизации / Г.М. Островский, Ю.М. Волин. – М.: Бинوم. Лаборатория знаний, 2008. – 308 с.
3. Петров И.В. Программируемые контроллеры. Стандартные языки и приемы прикладного программирования / И.В. Петров; под ред. В.П. Дьяконова. – М.: Салон-Пресс, 2004. – 410 с.
4. John Tukey, 85, Statistician; Coined the Word 'Software' [Електронний ресурс] // bell-labs. – 2003. – Режим доступу до ресурсу: <http://cm.bell-labs.com/cm/ms/departments/sia/tukey/nytimes.html>.
5. Автоматическое документирование проекта АСУ ТП и АСУП [Електронний ресурс] // AdAstrAResearchGroup, Ltd. – Режим доступу до ресурсу: http://www.tracemode.ua/products/overview/proj_doc/.
6. Руководство пользователя Трейс Моуд. Версия 5.0. – М.: Ad Astra Research Group, Ltd. 2006. – 814 с.
7. Пьявченко Т.А. Автоматизированные системы управления технологическими процессами и техническими объектами: учебное пособие / Т. А. Пьявченко. – Таганрог: ТРТУ, 1997. – 128 с.