

Міністерство освіти і науки України
Національний університет водного господарства
та природокористування
Навчально-науковий інститут автоматики,
кібернетики та обчислювальної техніки

Кафедра обчислювальної техніки

04-04-260М

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторних робіт з навчальної дисципліни
«Інформаційні системи реального часу» (частина 1)

для здобувачів вищої освіти першого (бакалаврського)
рівня за освітньо-професійною програмою
«Комп'ютерна інженерія» спеціальності 123
«Комп'ютерна інженерія» денної та заочної форм
навчання

Рекомендовано науково-
методичною радою з якості
ННІАКОТ
Протокол № 1 від 09.10.2023 р.

Рівне – 2023

Методичні вказівки до виконання лабораторних робіт з навчальної дисципліни «Інформаційні системи реального часу» (частина 1) для здобувачів вищої освіти першого (бакалаврського) рівня за освітньо-професійною програмою «Комп'ютерна інженерія» спеціальності 123 «Комп'ютерна інженерія» денної та заочної форм навчання [Електронне видання] / Шатна А. В., Шатний С. В., Бойчура М. В. – Рівне : НУВГП, 2023 – 62 с.

Укладачі: Шатна А. В., старший викладач кафедри обчислювальної техніки; Шатний С. В., к.т.н., доцент кафедри обчислювальної техніки; Бойчура М. В., к.т.н., старший викладач кафедри обчислювальної техніки.

Відповідальний за випуск: Круліковський Б. Б., к.т.н., доцент., зав. кафедри обчислювальної техніки.

Керівник (гарант) освітньої програми
«Комп'ютерна інженерія»
спеціальності 123 «Комп'ютерна інженерія» Сидор А. І.

© А. В. Шатна,
С. В. Шатний,
М. В. Бойчура, 2023
© НУВГП, 2023

ЗМІСТ

Вступ	4
Лабораторна робота №1. Вступ до системи реального часу LabVIEW	5
Лабораторна робота №2. Проєктування візуалізації та часові цикли в системі LabVIEW RealTime.....	15
Лабораторна робота №3. Проєктування автоматизованої системи керування конвеєрної стрічки в середовищі LabView Real-time	23
Лабораторна робота №4. Амплітудна модуляція за допомогою LabView RT	34
Лабораторна робота №5. Розробка та модернізація системи замовлення продукції в LabVIEW RT.....	43
Список використаної літератури	57
Додаток 1	58

Вступ

Дисципліна «Інформаційні системи реального часу» є складовою частиною підготовки студентів за спеціальністю 123 «Комп'ютерна інженерія». Метою викладання дисципліни «Інформаційні системи реального часу» є підготовка спеціалістів за освітньо-кваліфікаційним рівнем бакалавр відповідно до державних стандартів, встановлених освітньо-кваліфікаційною характеристикою (ОКХ) та освітньо-професійною програмою (ОПП) підготовки магістрів вищезазначеного фахового спрямування. Мета навчальної дисципліни – формування системи теоретичних і практичних знань студентами базових складових щодо побудови і функціонування систем реального часу та розробки програмного забезпечення у режимі реального часу.

Згідно з вимогами освітньо-професійної програми студенти після засвоєння дисципліни «Інформаційні системи реального часу» мають продемонструвати такі результати навчання як методи оцінки продуктивності обчислювальних систем, визначення та основні вимоги ,що пред'являються до систем реального часу, особливостей планування задач та типів планувальників, що використовують у системах реального час, проблем, що виникають при обміні інформацією між процесами і задачами, проблеми паралельної обробки та основних шляхів їх вирішення, методів оцінки часу виконання задач як у автономному режимі виконання, так і при роботі в конкретній програмній системі на різних стадіях проектування програмно-технічного комплексу реального часу.

Лабораторна робота №1

Вступ до системи реального часу LabVIEW

Мета: Навчитись працювати в системі реального часу LabView. Освоїти основні блоки та компоненти середовища LV, освоїти бібліотеки реального часу.

Короткі теоретичні відомості



LabVIEW — це програмне забезпечення для системної інженерії для додатків, які вимагають тестування, вимірювання та керування зі швидким доступом до апаратного забезпечення та даних. Середовище програмування LabVIEW спрощує інтеграцію апаратного забезпечення для інженерних додатків, щоб у вас був послідовний спосіб отримання даних від апаратного забезпечення NI та сторонніх виробників. LabVIEW зменшує складність програмування, тому ви можете зосередитися на своїй унікальній інженерній задачі. LabVIEW дозволяє негайно візуалізувати результати за допомогою вбудованого інженерного користувальницького інтерфейсу з функцією перетягування та інтегрованих засобів перегляду даних. Щоб перетворити отримані дані на реальні бізнес-результати, ви можете розробити алгоритми для аналізу даних і розширеного контролю за допомогою включеної математики та IP обробки сигналів або повторно використовувати власні бібліотеки з різноманітних інструментів. Щоб забезпечити сумісність з іншими інженерними інструментами, LabVIEW може взаємодіяти та повторно використовувати бібліотеки з іншого програмного забезпечення та мов з відкритим кодом.

LabVIEW має те, що вам потрібно для швидкого створення автоматизованих систем тестування, тисячі доступних функцій аналізу, настроювані інтерактивні елементи відображення, драйвери для автоматизації кожного приладу та обладнання для збору даних, підключення до інших мов і галузевих стандартних протоколів.

Порядок виконання роботи:

Частина 1: Логічні операції

1. Щоб створити новий проєкт необхідно зайти в меню File та вибрати NewVI або натиснути комбінацію клавіш Ctrl+N (рис.1.1)

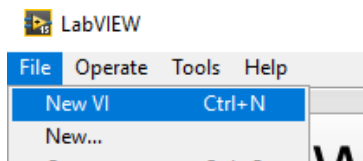


Рис.1.1. Створення проєкту

2. На робочому столі з'явиться два вікна: фронтальна панель (для роботи з графічною частиною) та блок-діаграма (для роботи з інструментами, циклами та іншим) .
3. Для того, щоб відкрити будь-яку панель інструментів необхідно на потрібному вікні клацнути ПКМ. Приклад панелі інструментів наведено нижче рис.1.2.

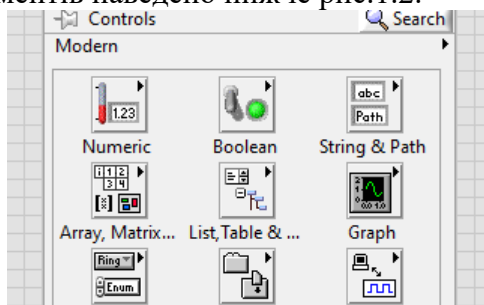


Рис.1.2. Панель інструментів для фронтальної панелі

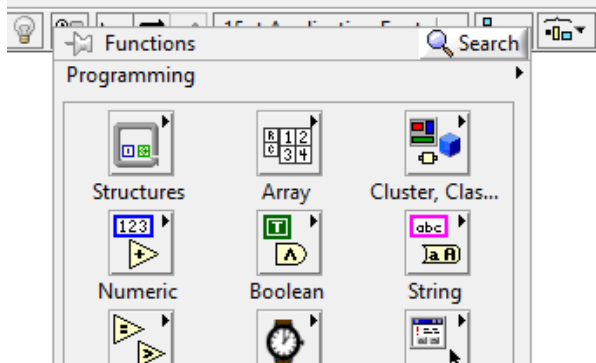


Рис.1.3. Панель блок-діаграм.

4. Перегляньте всі вкладки панелей.
5. Розв'язати задачу: Знайти суму двох значень X та Y. Відповідь присвоїти змінній Z.
6. Для того, щоб розв'язати задачу, спочатку необхідно в фронтальній панелі відобразити необхідні вікна куди буде записуватись значення. Приклад наведено нижче рис.1.4.

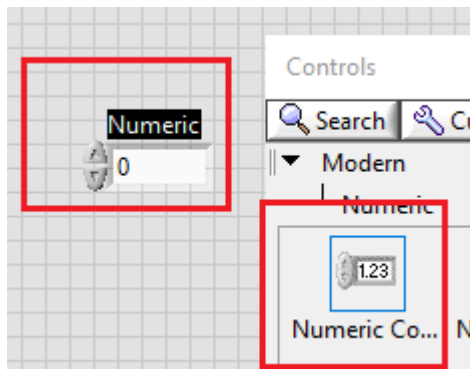


Рис.1.4. Розташування екрану

7. Розмістити на екрані три циферблати, та перейменувати їх на задані нам значення. Приклад наведено нижче рис.1.5.

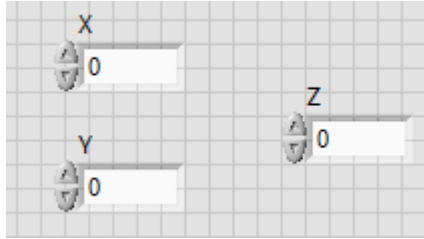


Рис.5. Розташування циферблату

8. На панелі діаграм, за допомогою математичних функцій з панелі інструментів знайти суму двох 2 чисел та здійснити прив'язки. Приклад наведено нижче рис.1.6.

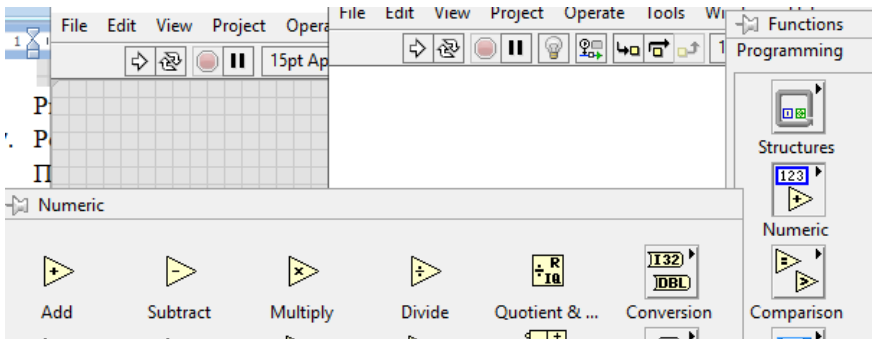


Рис.1.6. Математична панель

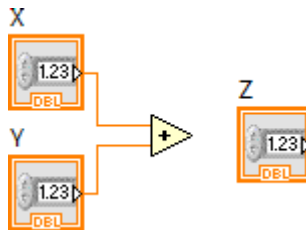


Рис.7. Під'єднання математичної функції додавання

9. Для того, що записати число в комірці, необхідно присвоїти їй індикатор. На даний момент всі комірki являються контрольними. Для цього необхідно клацнути по циферблату ПКЛ, та вибрати дії наведені нижче рис.1.8:

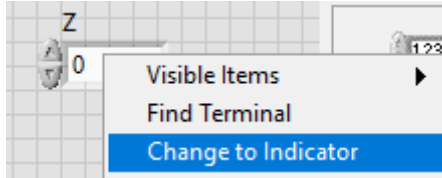


Рис.8. Перетворення контролю в індикацію.

10. Закінчити налаштування та запустити програму

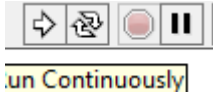


Рис.1.9. Запуск програми: 1 – раз, нескінченний цикл, пауза, стоп

11. Результат буде виглядати наступним чином:



Рис.1.10. Результат виконання програми

12. Всі значення можна змінювати та отримувати в режимі реального часу.

Частина 2. Створення циклів та часових затримок

1. Створити дві структури WHILE

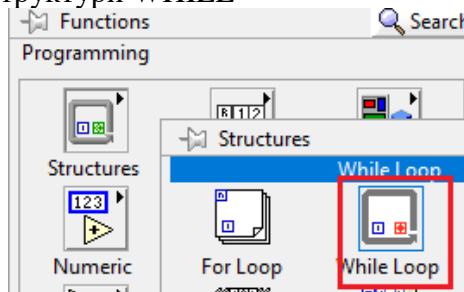


Рис.2.1. Цикл While

2. Додати до циклу кнопки для екстреного виходу

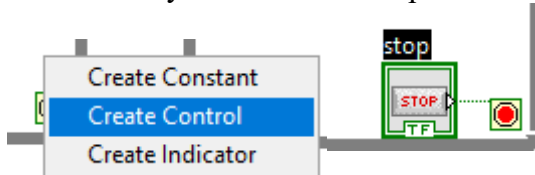


Рис.2.2. Кнопка стоп

3. Для зручності відстеження роботи дії в циклі додати індикатори

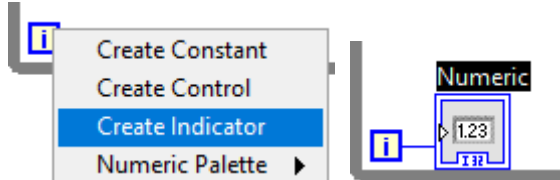


Рис.2.3. Індикатори

4. Для відслідковування затримки в часі, додати функцію по часу. Приклад нижче



Рис.2.4. Додання часу

5. Зліва від кожного значка створити константи, зі введеними затримками. Для першого циклу 300 мс, для другого 600 мс. Приклад нижче

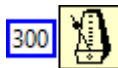


Рис.2.5. Встановлення затримки

6. Запустити програму. Числа на фронтальній панелі, будуть йти скоріше у верхньому виводі. Оскільки там менша затримка в часі.

7. Додати на фронтальну панель порівняння АБО



Рис.2. 6. Елемент АБО

8. Зробити порівняння з ітераціями, Для того щоб створити вихід із циклу по досягненню певного значення, АБО при натисненні кнопки стоп. Під'єднати схему наступним чином:

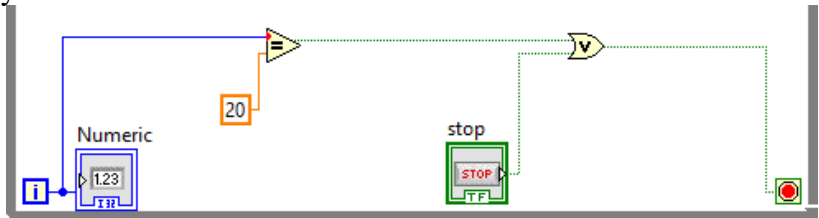


Рис.2.7. Схема підключення логічних елементів

9. Запустити програму. Зафіксувати результат.

10. Огляд циклу For. Додати цикл for. N- кількість ітерацій яку повинен пройти цикл. I – номер ітерації.

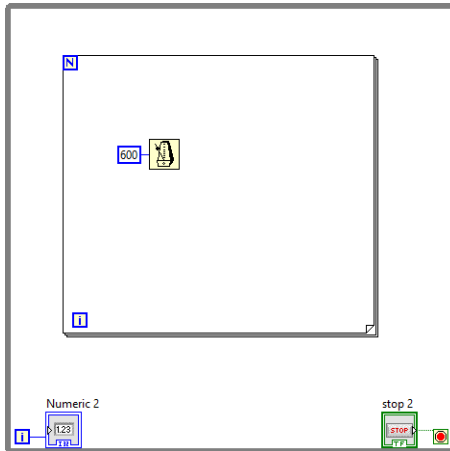


Рис.2.8. Цикл for

11. Виконати наступні налаштування:

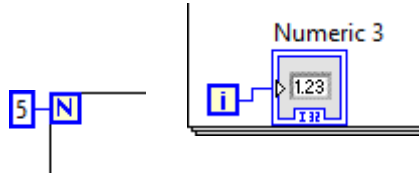


Рис.2.9. Налаштування циклу for

12. Додати взаємозв'язок . Додати константу та порівняння. При умові коли виконається константа буде вихід на індикатор. Приклад нижче

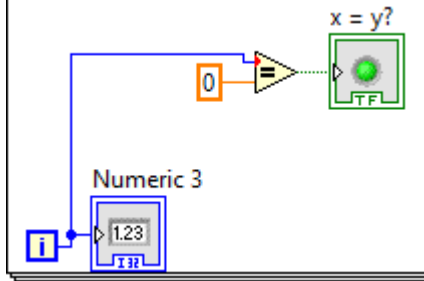


Рис.2.10. Порівняння

13. Розгляд структури Case. Створити новий проєкт. Додати структуру case аналогічно до циклів.

14. Дана структура містить в собі два значення true і false. Заповнимо структуру true. Для цього на фронтальну панель додамо булевий індикатор, та на панелі діаграм перемістимо його в case. Змінити його тип на контрол.

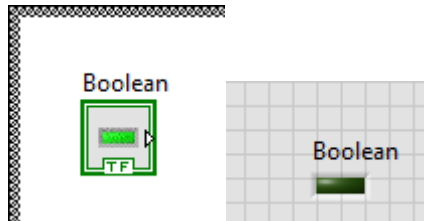
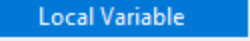


Рис.2.11. Булевий індикатор

16. До виходу індикатора додати булеве заперечення.

Додати на вихід за печення  (локальна змінна), клацнувши ПКЛ по індикатору.

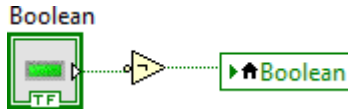


Рис.2.12. Налаштування індикатора

17. Далі опрацюємо вкладку false. Для цього необхідно на неї перейти. Далі встановити на графічній панелі круглий буле вий індикатор. Виконати наступні прив'язки

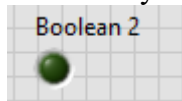


Рис.2.13. Опрацювання хибного значення

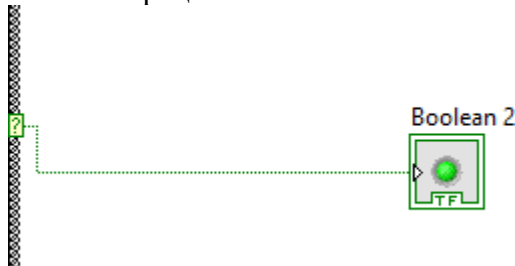


Рис.2.14. Підключення хибного значення

18. Далі на графічну частину додати вертикальний булевий контролер

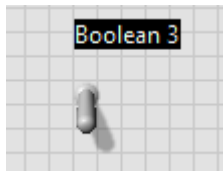


Рис.14. Булевий контролер

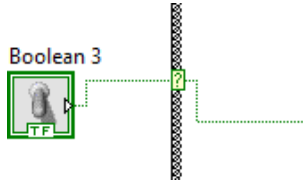


Рис.2. 15. Підключення булевого контролера

19. Коли контролер в стані false (вимкнений), на вхід надходить логічний нуль. Перевірити роботу програми.

Варіанти до виконання лабораторної роботи:

1. Розв'язати наступні рівняння, дані взяти згідно свого варіанту де X-номер в списку групи, Y-дата вашого народження, K-довільне значення. Оформити звіт. Зробити висновки.

2. $X+Y=J$

3. Розв'язати рівняння $X/K=V$

4. Розв'язати рівняння $K*Y=L$

5. $M=J+L-V$

Вміст звіту

1. Тема, мета роботи.
2. Хід роботи.
3. Результати розв'язання наведених задач в порядку виконання згідно варіанта (скрінні бло-діаграм та фронтальної панелі).
4. Висновки про результат виконаної роботи.

Контрольні питання:

1. Для чого призначене середовище LabVIEW?
2. Які вікна містяться в проекті NV?
3. Як створити новий проект в середовищі LV?
4. Які типи циклів можна будувати в середовищі LV?
5. Що містить в собі структура Case?

Лабораторна робота №2

Проектування візуалізації та часові цикли в системі LabVIEW RealTime

Мета: Навчитись створювати та програмувати візуальні об'єкти DSC module в середовищі LabVIEW RealTime. Навчитись та освоїти методику програмування та відтворення часових циклів у LabVIEW RealTime.

Короткі теоретичні відомості



DSC (The LabVIEW Datalogging and Supervisory Control) - Модуль реєстрації даних LabVIEW і контрольного керування. Модуль допомагає

розробляти HMI/SCADA або додатки для реєстрації даних із великою кількістю каналів.

Модуль LabVIEW Datalogging and Supervisory Control (DSC) є додатковим програмним забезпеченням для середовища програмування LabVIEW. Ви можете використовувати цей модуль для розробки розподіленої системи моніторингу та контролю з тегами від кількох десятків до десятків тисяч. Він містить інструменти для реєстрації даних у мережеву історичну базу даних, відстеження тенденцій у реальному часі та історичних тенденцій, керування сигналами тривоги та подіями, об'єднання цілей LabVIEW Real-Time та пристроїв OPC в одну повну систему та додавання безпеки для інтерфейсів користувача. Ви можете вибрати ліцензію на розробку, ліцензію на розгортання для розповсюдження коду розробки або ліцензію на налагодження/розгортання, які

можна використовувати для вирішення проблем із програмами, створеними за допомогою ліцензії на розробку.

В даній роботі описуються функціональні можливості, які надає модуль LabVIEW Datalogging and Supervisory Control (DSC) для полегшення впровадження безпеки для бібліотек проєктів LabVIEW, спільних змінних і об'єктів передньої панелі. Також обговорюється, як архітектура безпеки відокремлює керування користувачами та автентифікацію від дозволів на ресурси.

У LabVIEW 7.x і раніших версіях модуль LabVIEW DSC використовує базу даних облікових записів користувачів на кожному комп'ютері для автентифікації користувачів і забезпечення безпеки для різних ресурсів. Наявність єдиної бази даних облікового запису користувача створює проблему, оскільки комп'ютер, який виконує автентифікацію, має відомості лише про користувачів на комп'ютері, на якому знаходиться база даних. Комп'ютер, який виконує автентифікацію, не знає користувачів на інших комп'ютерах. Крім того, підключення бази даних облікових записів користувачів до кожного комп'ютера може бути неможливим для вбудованих цілей або цілей реального часу через їх обмежену обчислювальну потужність.

Порядок виконання роботи:

1. Створити новий бланк проєкту рис.2.1

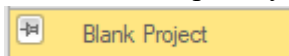


Рис.2.1. Бланк проєкту

2. Створити вікно контролю

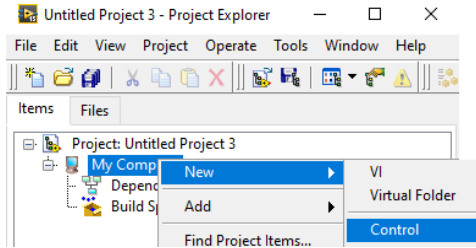


Рис.2.2. Вікно Control

3. Відкрити навігатор зображень Tools -> DSC Module -> Image Navigator

4. Додати два перемикачі

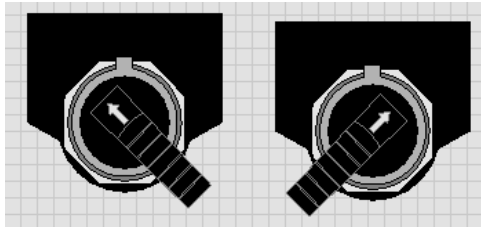


Рис.2.3. Перемикачі

5. Натиснути ПКЛ на вікні контролю -> Express -> LEDs -> Round LED

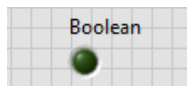


Рис.2.4. Лампочка

6. Виконати дії п.6.1.-6.6, рис.2.5-2.6

6.1. Виділити перший перемикач та натиснути



, з'явиться

6.2. Натиснути ПКЛ на виділений перемикач, та виконати наступну дію

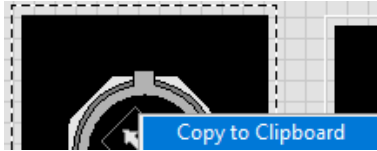


Рис.2.5. Копіювання компонента в буфер

6.3. Натиснути ПКЛ на Boolean, та виконати наступні дії

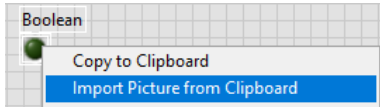





Рис.2.6. Імпортування компонента із буфера

6.4. Натиснути , з'явиться . Натиснути на , потім на скопійований перемикач. Елемент Boolean стане зеленим.

6.5. Аналогічним чином зробити з другим елементом.

6.6. Стерти два лишніх елементи. Зберегти файл під назвою L1.

7. Створити вікно контролю

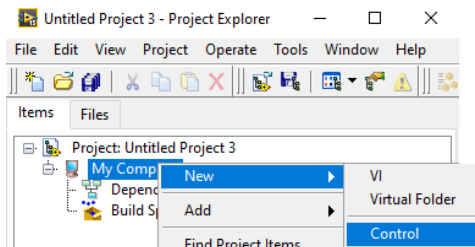


Рис.2.7. Вікно Control

8. Натиснути ПКЛ на вікні контролю -> Express -> LEDs -> Round LED

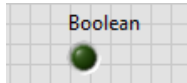


Рис.2.8. Лампочка

9. Відкрити навігатор зображень Tools -> DSC Module -> Image Navigator

10. В меню Blowers Etc, вибрати два кулера рис.2.9

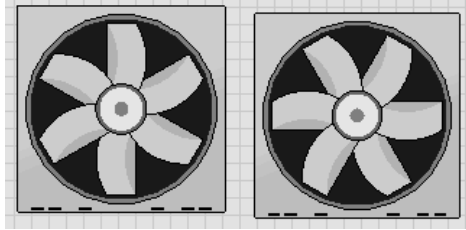


Рис.2.9. Кулер

11. Виконати дії з кулером відповідно до пункту 6.

12. Зберегти файл під назвою F2.

13. Створити вікно контролю

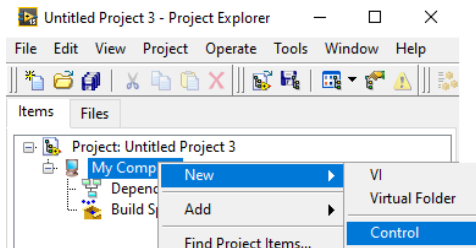


Рис.2.10. Вікно Control

14. Натиснути ПКЛ на вікні контролю -> Express -> LEDs -> Round LED

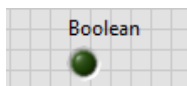


Рис.2.11. Лампочка

15. Відкрити навігатор зображень Tools -> DSC Module -> Image Navigator

16. В меню Nature, вибрати два Fire 6 та Fire 7 рис.2.12

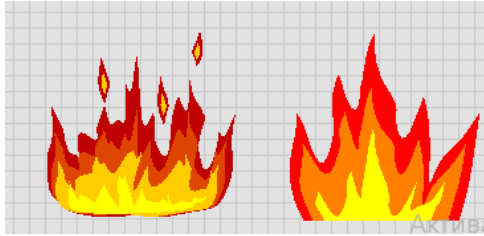


Рис.2.12. Вогонь

17. Виконати дії відповідно до пунктів 6 та 11.

18. Зберегти файл під назвою F3

19. Створити новий проєкт VI

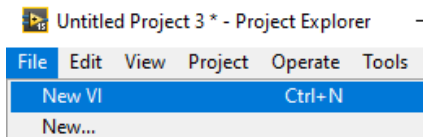


Рис.2.13. Новий проєкт

20. Додати на фронтальну панель файл F1, F2, F3 перетягнувши їх

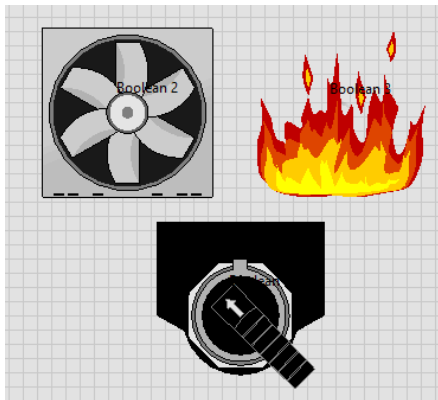


Рис.2.14. Розміщення елементів на фронтальній панелі

21. Додати на фронтальну панель кнопку із меню Express-> Buttons -> Push Button



Рис.15. Кнопка

22. Додати на блок діаграму цикл та структура Case. Зробити кнопку екстреного виходу з циклу. Підключити компоненти як показано на рис.2.15.

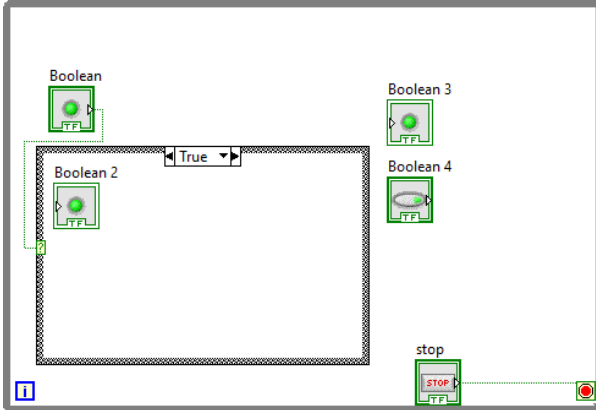


Рис.2.15. Розміщення елементів зі структурою

23. Додати функцію Not, додати Local Variable, ПКМ по Boolean2 -> Create -> Local Variable

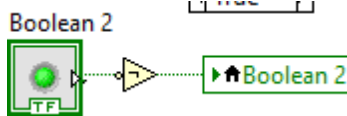


Рис.2.16. Локальна змінна

24. Запустити програму. При натисканні на перемикач буде прокручуватись кулер.

25. Додати ще одну структуру Case і цикл. Виконати дії для Boolean3, Boolean4 аналогічно як з Boolean1 та Boolean2.(пункт 23)

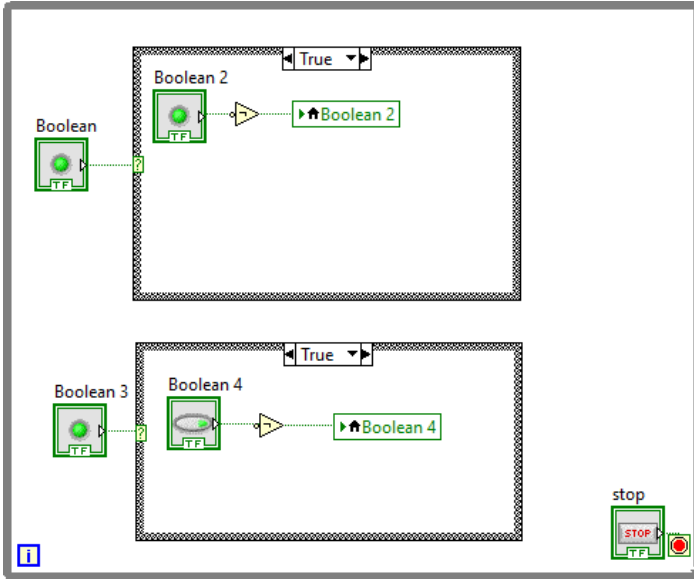


Рис.2.17 Блок-діаграма проекту

26. Запустити програму. При натисканні на кнопку горить вогонь.

Варіанти до виконання лабораторної роботи:

Згідно порядку виконання лабораторної роботи зафіксувати для звіту вже пройдений та виконаний проєкт, змість файлу F3 (вогонь) вставити будь-яке інше зображення для анімації з бібліотеки моделей. Оформити звіт, зробити висновки.

Вміст звіту

1. Тема, мета роботи.
2. Хід роботи.
3. Результати розв'язання наведених задач в порядку виконання згідно варіанта (скрини бло-діаграм та фронтальної панелі).

4. Висновки про результат виконаної роботи.

Контрольні питання:

1. Що таке модуль DSC?
2. Як створити вікно контролю?
3. Як скопіювати компонент в буфер?
4. Як імпортувати компонент з буфера?
5. Як додати екстрену зупинку циклу?

Лабораторна робота №3

Проектування автоматизованої системи керування конвеєрної стрічки в середовищі LabView Real-time

Мета: навчитись створювати анімовані функціональні схеми автоматизації в режимі реального часу в системі LabView.

Короткі теоретичні відомості

На відміну від інструментів загального призначення, LabVIEW призначений для тестування та вимірювання. Він використовує G, графічну мову програмування високого рівня, яка допомагає інженерам перекладати їхні завдання в код. Економте час розробки, програмуючи так, як ви думаєте.

Основні функції для графічного програмування

1. Програмування потоку даних. Графічне програмування пропонує більш інтуїтивно зрозумілий

досвід, ніж традиційні мови сценаріїв. LabVIEW абстрагує синтаксис і більше нагадує фізичні моделі.

2. Паралельне програмування. Інженери підвищують продуктивність, виконуючи завдання паралельно. Паралелізм, властивий LabVIEW, дозволяє виконувати багатозадачність і багатопотоковість без складного кодування.

3. Інтерактивне налагодження. LabVIEW автоматично компілює код на кожному кроці та має інтерактивні засоби налагодження. Користувачі отримують попередження про проблеми та можуть виконати покрокове виконання, щоб вирішити їх.

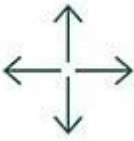
LabVIEW для інтуїтивного програмування



Програма як ви думаєте. LabVIEW допомагає скоротити час розробки, обслуговуючи інженерів і науковців, які часто є «візуальними мислителями», які описують свої системи за допомогою блок-схем і діаграм.



Інтегрувати інший код. Інженери інтегрують код із програмного забезпечення Python, MathWorks, MATLAB, C і .NET у програми LabVIEW для створення гнучкої системи тестування, яка поєднує в собі найкраще з усіх програмних засобів.



Масштаб у майбутньому. NI має набір програмного забезпечення для тестування, і LabVIEW є основою. Клієнти можуть масштабувати продукти, які працюють разом і відповідають їхнім потребам.



Залучайтеся до спільноти. LabVIEW використовується сотнями тисяч інженерів у всьому світі. Розробники покладаються на спільноту NI, щоб створити мережу, навчатися або наймати консультантів і системних інтеграторів.

Порядок виконання роботи:

1. Створити новий проєкт

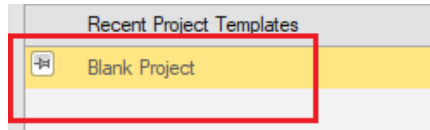


Рис.3.1. Створення проєкту

2. В панелі що з'явилась додати контроль

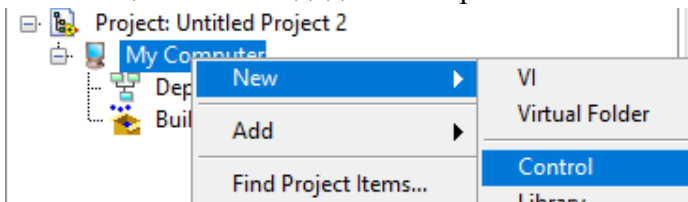


Рис.3.2. Панель контролю

3. Додати на графічну частину повзунок Modern -> Numeric -> Horizontal Pointer Slide.
4. Встановити максимальну величину повзунка 100 та вирівняти його на екрані. Перейменувати, як транспортер_1.

5. Додати контейнер та конвеєрну стрічку за допомогою меню Tools -> DSC Module -> Image Navigator. В меню Image Navigator, спочатку вибрати підменю Containers, та витягти ящик



Рис.3.3. Зображення ящика

6. В меню Image Navigator, вибрати підменю Conveyors, Misc, та витягти конвеєрну стрічку

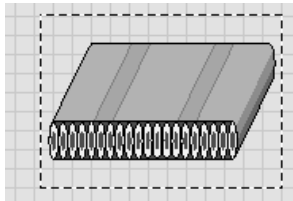


Рис.3.4. Рисунок конвеєрної стрічки

7. Аналогічно з попередньою лабораторною роботою, додати ящик на повзунок. Або вирізати ящик, та вставити його на початкову точку повзунка.

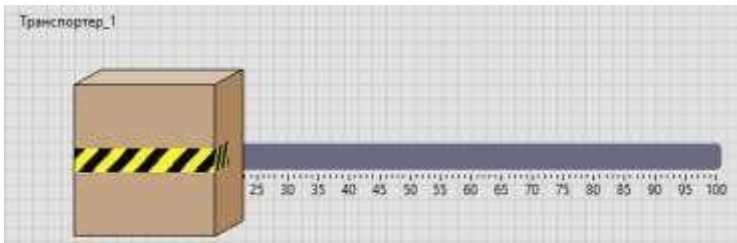



Рис.3.5. Встановлення ящика

8. Якщо вибрати налаштування курсору  , та потягти за ящик має відбуватися наступна дія:

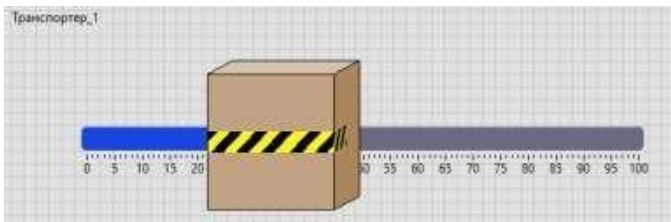


Рис.3.6. Анімація ящика

9. Аналогічно до встановлення ящика, встановити конвеєрну стрічку на повзунк. Результат буде виглядати наступним чином:



Рис.3.7. Встановлення ящику на стрічку


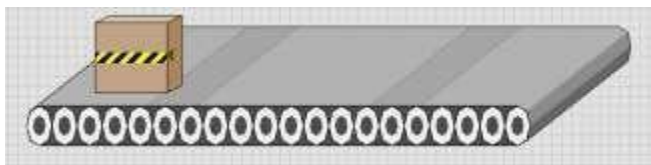
10. За допомогою меню  , зробити вирівнювання. Результат буде виглядати наступним чином

Рис.3.8. Встановлення ящику на стрічку



11. Зберегти файл як транспортер_1.
12. Додати ще один контроль.

13. Додати буле вий перемикач за допомогою меню Modern -> Boolean -> HorizontalToggle Switch



Рис.3.9. Булевий перемикач

14. Із меню Tools -> DSC Module -> Image Navigator вибрати 3-D Pushbuttons Etc ->Selector Switch1(left) та Selector Switch1(right)

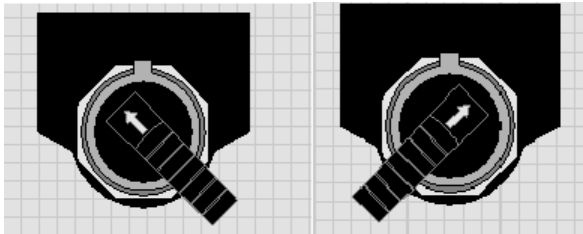


Рис.3.10. Розподільвач

15. Аналогічно до імпортування ящика на повзунок (а також лабораторна робота 2), імпортувати перемикачі на буле вий перемикач. Має спрацювати анімація, як в лабораторній роботі №2.
16. Зберегти файл, як перемикач.
17. Створити новий проєкт VI
18. Додати в графічну частину файли Транспортер_1 та перемикач.
19. Додати назви Старт та Стоп.

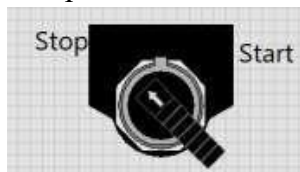



Рис.3.11. Опис розподільника

20. Згрупувати елементи перемикача за допомогою кнопки меню .
21. За допомогою меню Modern -> Decoration - > додати Vertical Smooth Box. Розмістити бокс спочатку на весь малюнок.



Рис.3.12. Розміщення вертикального боксу

22. Далі за допомогою меню  Move to Back, помістити його на задній фон.
23. Скопіювати конвеєр та вставити ідентичний внизу. З перемикачем необхідно виконати ті самі дії.

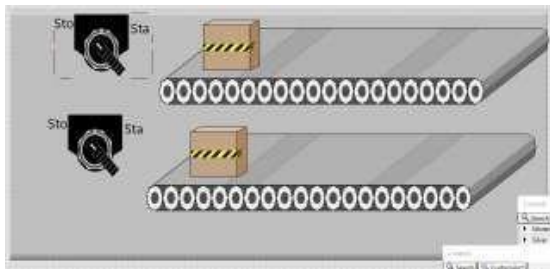


Рис.3.13. Схема проєкту

24. Вставити горизонтальний бокс, та помістити його спочатку на перемикачі. Modern -> Decoration - > додати Horizontal Smooth Box., а потім на задній фон перемикачів, аналогічно до вертикального боксу. Потім помістити на задній фон вертикальний бокс. Результат буде виглядати наступним чином.

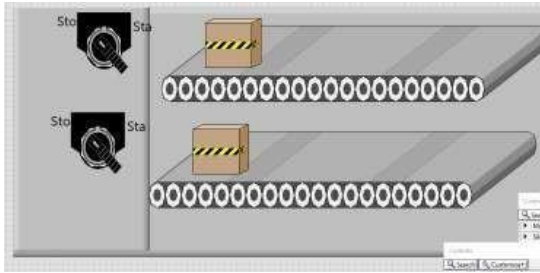


Рис.3.14. Меню проєкту

25. Додати із пункту Modern -> Decoration -> Thick лінею та нею розділити конвеєри.
26. Додати кнопку зупинки Silver -> Boolean -> Stop Button. Та дві кнопки скидання (перезавантаження) Silver -> Boolean -> Push Button.



Рис.3.15. Встановлення горизонтального боксу

27. Тепер перейдемо до вікна прив'язок , та запрограмуємо схему.
28. Помістити в цикл While всю схему.

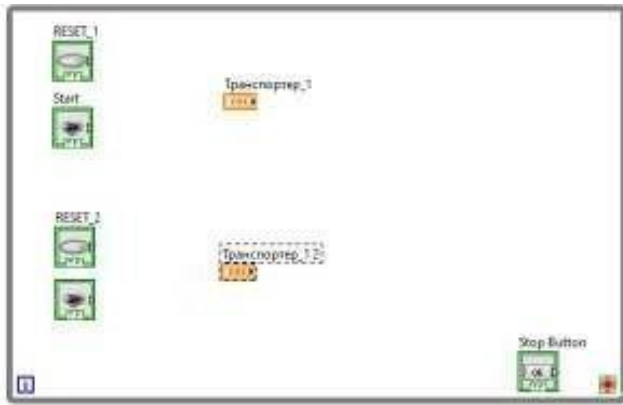


Рис.3.16. Встановлення циклу

29. Під'єднати кнопку стоп до стопової кнопки циклу.

30. Додати дві константи Programming -> Numeric -> DBL Numeric Constant, та встановити їх поза циклом. Прикріпити їх до циклу. Та виконати наступні дії:

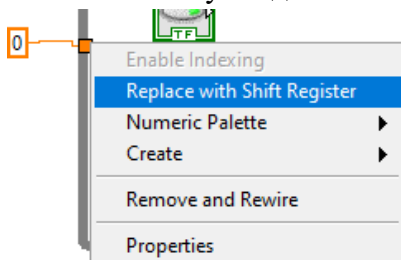


Рис.3.17. Прив'язка констант

31. Додати наступну функцію в цикл = Comprasion -> Select. На середній вхід функції подати сигнал з виходу скидання1, на верхній вхід константу, нижній підключити до константи циклу. Приклад виконання нижче.

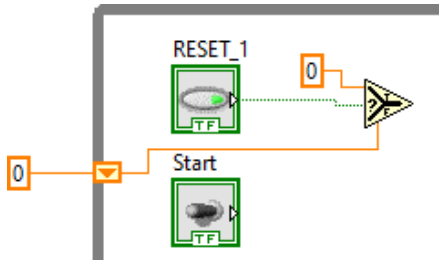


Рис.3.18. Встановлення вибірки

32. Виконати наступні дії, які зображено на схемі:

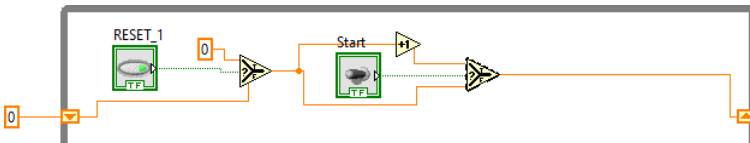


Рис.3.19. З'єднання всіх функцій першої конвєрсної стрічки

33. Виконати аналогічні дії з кнопками ресет2 та старт 2.

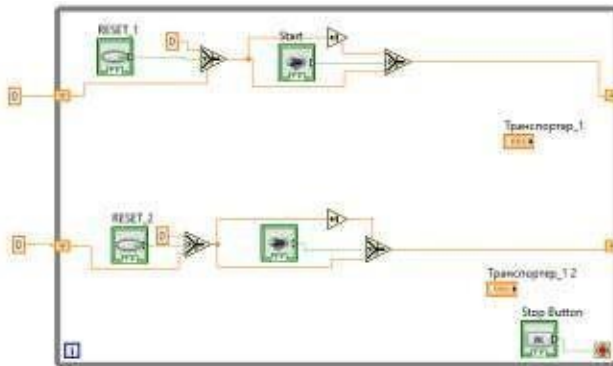


Рис.3.20. Підключення всієї системи

34. Перетворити транспортер_1 в індикатор, та під'єднати до лінії, аналогічно зробити з транспортером 2.

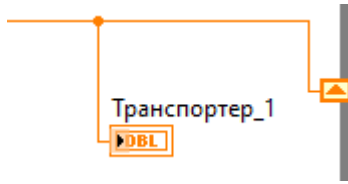


Рис.3.21. Підключення транспортера
Додати таймер часу, для встановлення затримок, та додати на його вхід константу в 200 мс.

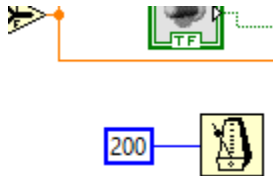


Рис.3.22. Встановлення часової затримки

35. Запустити програму на виконання. Верхній ящик повинен двигатись швидше нижнього.

Варіанти до виконання лабораторної роботи:

Згідно порядкового номера в списку групи вибрати часову затримку до схеми. Змінити кольорове оформлення лінії на ваш розсуд.

Варіант	01	02	03	04	05	06	07	08	09
Час.зат.	210	220	230	240	250	260	240	230	210

Вміст звіту

1. Тема, мета роботи.
2. Хід роботи.
3. Результати розв'язання наведених задач в порядку виконання згідно варіанта (скріни бло-діаграм та фронтальної панелі).
4. Висновки про результат виконаної роботи.

Контрольні питання:

1. Як встановити константу в циклі?
2. Як згрупувати елементи в схеми?
3. Як імпортувати елементи у вікні?
4. Як в даній лабораторній роботі виконувалась прив'язка ящика до стрічки?
5. В якому меню знаходяться розподільники?

Лабораторна робота №4 Амплітудна модуляція за допомогою LabView RT

Мета: Навчитись будувати та отримувати результати графіків амплітудної модуляції середовищі LabVIEW RealTime в режимі реального часу. Освоїти різні типи виведення сигналів.

Короткі теоретичні відомості

Системи амплітудної модуляції

У системах зв'язку нам часто потрібно проектувати та аналізувати системи, у яких багато незалежних повідомлення може бути передано одночасно через той самий фізичний канал передачі. Це є це можливо за допомогою техніки, що називається мультиплексуванням із частотним поділом, за якої кожне повідомлення транслюється частоти, щоб зайняти інший діапазон спектру. Це включає допоміжний сигнал, який називається несучою який визначає величину частотного перекладу. Це

вимагає модуляції, в якій або амплітуда, частота або фаза несучого сигналу змінюється відповідно до миттєвого значення сигнал повідомлення. Тоді результуючий сигнал називається модульованим сигналом. Коли амплітуда в несуча змінюється відповідно до миттєвого значення повідомлення/сигналу основної смуги, це призводить до

Переваги частотного перетворення такі:

1. Частотне мультиплексування: у випадку, коли є більше ніж одне джерело, яке виробляє сигнали з обмеженою смугою, які лежать в одній смузі частот. Такі сигнали передаються як такі одночасно через канал передачі, вони будуть заважати один одному і не можуть бути повернуто до призначеного отримувача. Але якщо кожен сигнал переводиться в частоту таку що вони охоплюють різні діапазони частот, не заважаючи іншим спектрам сигналу, тоді кожен сигнал можна відокремити назад у приймачі за допомогою відповідних фільтрів. Вихід фільтрів, потім можна відповідним чином обробити, щоб повернути вихідний сигнал повідомлення.

2. Практичність антени: у бездротовому середовищі антени використовуються для випромінювання та прийому сигнали. Антена працює ефективно лише тоді, коли розмір антени дорівнює порядку величини довжини хвилі відповідного сигналу. На низьких частотах основної смуги, довжина хвилі велика, тому необхідний розмір антени є непрактичним. За частотою трансляції сигнал може бути зміщений за частотою до більш високого діапазону частот. Звідси відповідна довжина хвилі є малою настільки, що необхідний розмір антени є достатнім маленький і практичний.

Типи амплітудної модуляції:

1. Двостороння смуга з носієм (DSB+C)
2. Двостороння смуга придушеної несучої (DSB-SC)
3. Односмугова придушена несуча (SSB-SC)
4. Рудиментарна бічна смуга (VSB)

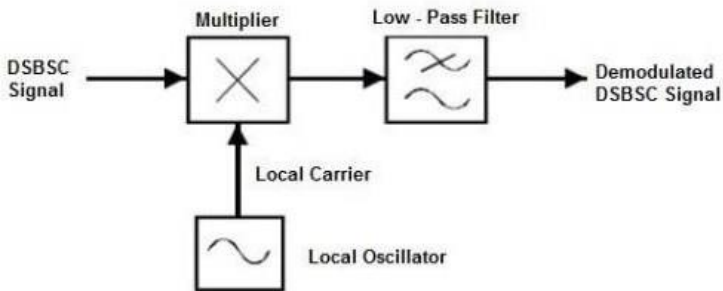


Рис.4.1. Синхронне виявлення DSBSC

Порядок виконання роботи:

1. В середовищі LabView створити новий проект з VI;
2. На основі простої функції $y(t) = [1 + M \sin(wmt)]. A \sin(wct)$

побудувати контрольну панель та відобразити три типи сигналів: сигнал повідомлень, сигнал модуляції та сигнал носія відповідно до заданої функції.

3. Додати на панель блок діаграми 2 симулятори сигналу з меню Signal Processing – Waveform Generation – Simulate Signal. Назвати їх «Сигнал повідомлень» та «Сигнал носія» відповідно. Встановити налаштування, які показано на рис.4.2.

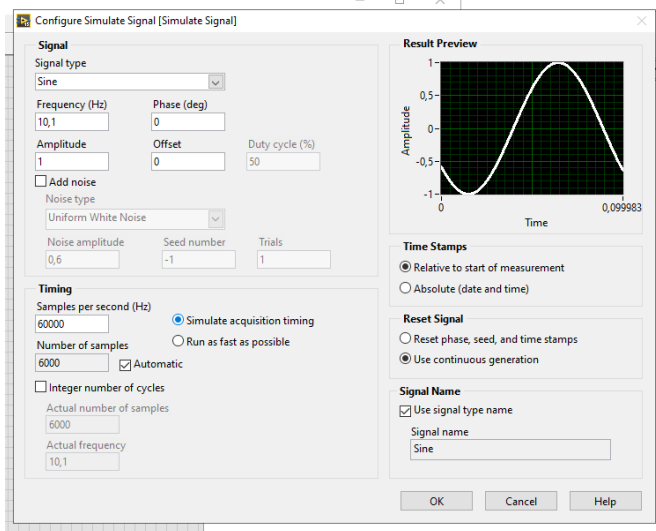


Рис.4.2. Вікно налаштувань симулятора сигналів

4. На графічну панель додати 3 ручки для перемикання (перемикачі) рис.4.3. із меню Numeric - Knob. Перший перемикач «Частота повідомлень», шкалу встановити на 100 макс. Другий перемикач «Частота носія», шкалу встановити на 1000 макс., третій перемикач «Індекс модуляції» шкалу встановити на 3 макс.

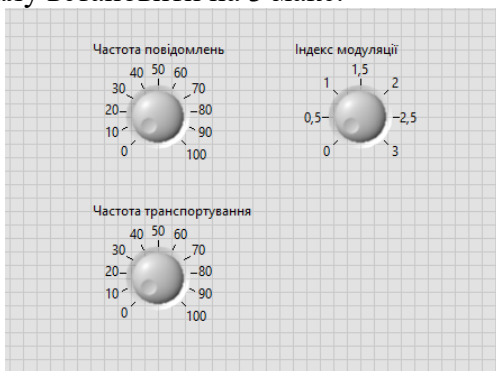


Рис.4.3. Графічне зображення перемикачів

5. В блок-діаграмі підключити перемикачі до симуляторів сигналу, 2 вхід «Frequency», рис.4.4.

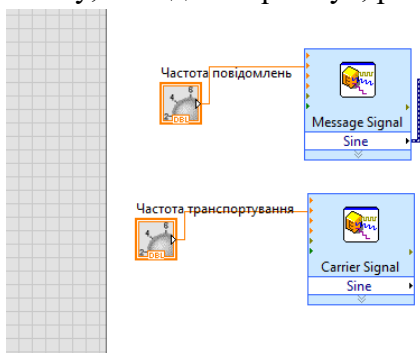


Рис.4.4. Підключення симуляторів сигналів та перемикачів

6. Додати 2 множники (Multiply) та одне додавання (Add), нижній вхід додавання встановити константу 1. та підключити все як показано на рис.4.5.

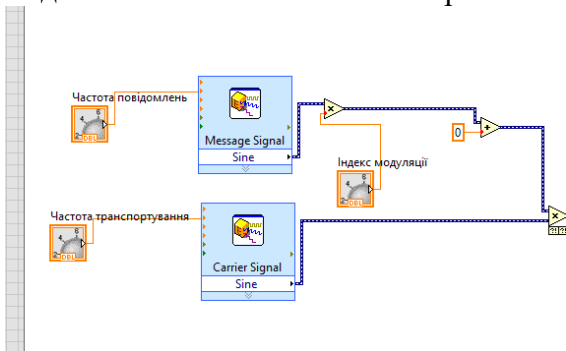


Рис.4.5. Схема підключення

7. На графічну панель додати 3 графіки з панелі Graph – Waveforms Graph. Перший «Сигнал повідомлень», другий «Сигнал носія», третій «Сигнал модуляції» відповідно рис.4.6.

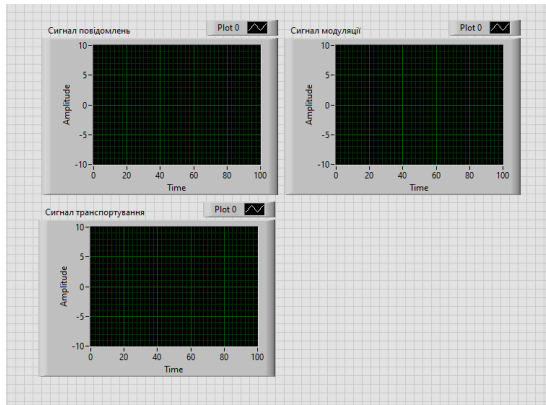


Рис.4.6. Відображення графіків

8. Підключити додані інструменти в блок-діаграмі рис.4.7.

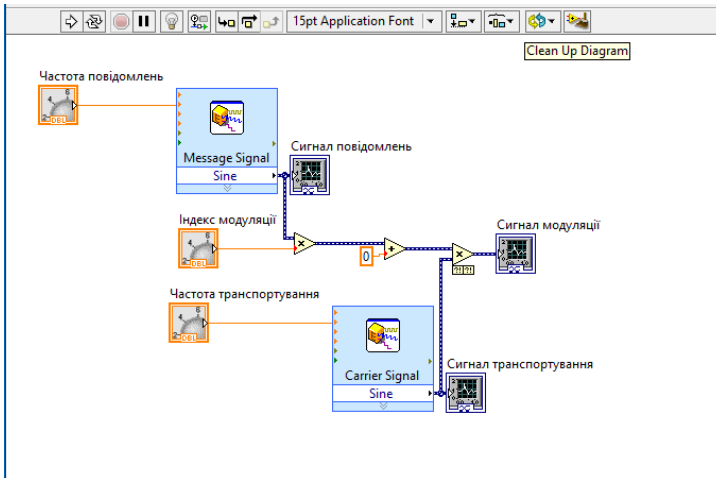


Рис.4.7. Підключення сигналів графіків

9. Запустити та змінивши значення перемикача отримати результат зміни графіків рис.4.8.

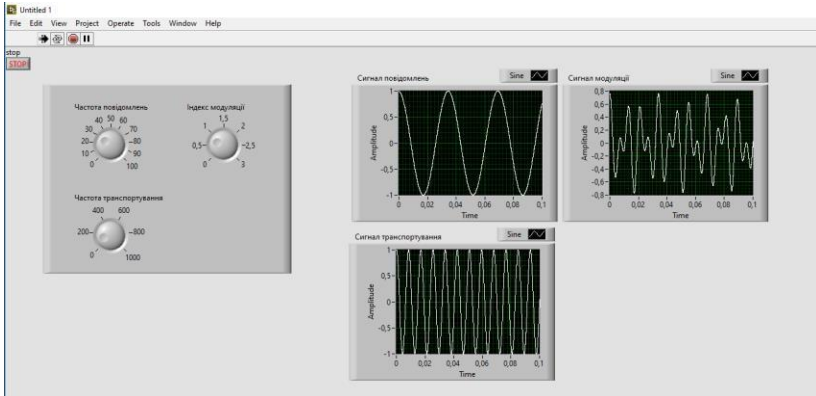


Рис.4.8. результат опрацювання графіків

10. Додати структуру While loop, та додати в блок діаграмі до частоти носія та до частоти повідомлень індикатори та підключити їх як показано на рис.4.9.

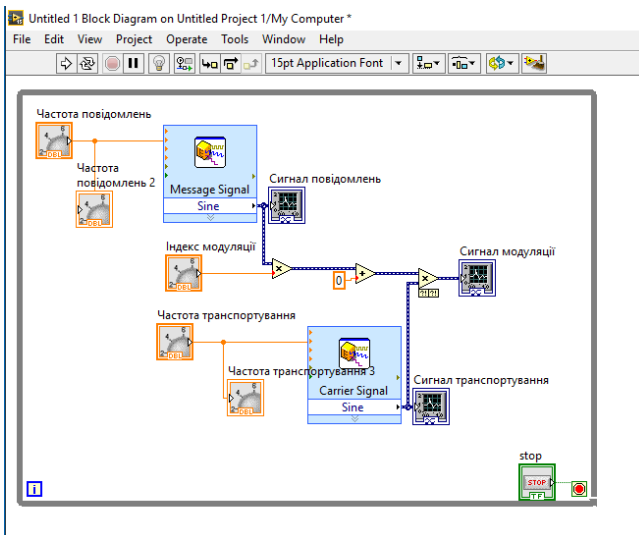


Рис.4.9. Підключення індикаторів та структури While

11. Встановити індикатори на відповідні місця рис.4.10. Запустити програму та отримати графіки.

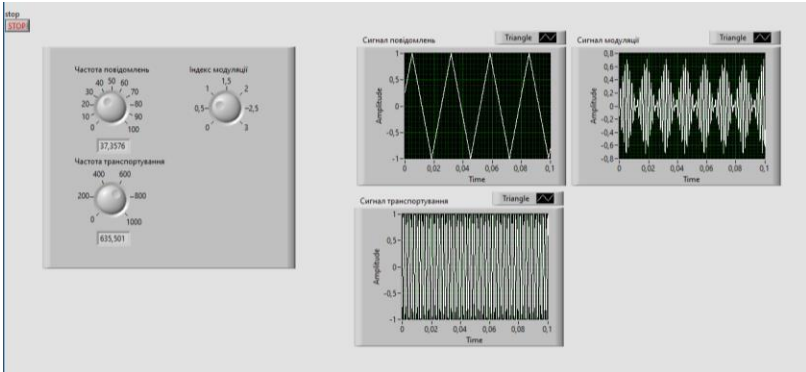


Рис.4.10. Графічна панель

12. Змінити налаштування блоку сигналу повідомлень на відображення графіків трикутної форми, для цього натиснути на ньому подвійник кліком та обрати відображення Triangle рис.4.11.

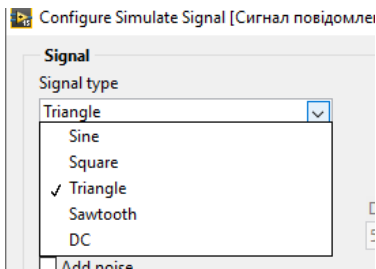


Рис.4.11. Налаштування відображення сигналу

13. Запустити програму та отримати результат. Переглянути всі види виведення сигналів та отриманий результат з таблиці варіантів вставити в звіт.

Варіанти до виконання лабораторної роботи:

Згідно порядкового номера в списку групи вибрати тип сигналів до схеми. Змінити кольорове оформлення на ваш розсуд.

Варіант	01	02	03	04	05
Сигнал	Sine, Triangle	Square, DC	Sawtooth, DC	Sine, DC	Square, Triangle
Варіант	06	07	08	09	10
Сигнал	Sine, Square	Square, Sawtooth,	Sine, Sawtooth	Sawtooth , Triangle	Triangle, DC

Вміст звіту

1. Тема, мета роботи.
2. Хід роботи.
3. Результати розв'язання наведених задач в порядку виконання згідно варіанта (скріни блок-діаграм та фронтальної панелі).
4. Висновки про результат виконаної роботи.

Контрольні питання:

1. Як підключити індикатор до циклу?
2. Як змінити тип графіка в програмі?
3. В якому меню знаходяться симулятори сигналів?
4. Які існують типи амплітудної модуляції?
5. Назвіть одну з переваг амплітудної модуляції?

Лабораторна робота №5

Розробка та модернізація системи замовлення продукції в LabVIEW RT

Мета: Навчитись створювати системи для передачі та підрахунку інформації в системах обслуговування в режимі реального часу. Модернізувати розроблений проект згідно свого варіанту.

Короткі теоретичні відомості

Інформаційні технології сьогодні можна класифікувати за такими ознаками: способом реалізації в інформаційній системі, ступенем охоплення завдань управління, класами реалізованих технологічних операцій, типом користувацького інтерфейсу, варіантами використання мережі, що обслуговує предметну область.

Управління – найважливіша функція, без якої неможлива цілеспрямована діяльність будьякої соціально-економічної, організаційно-виробничої системи (підприємства, організації, території). Систему, що реалізує функції управління, називають системою управління. Найважливішими функціями, реалізованими цією системою, є прогнозування, планування, облік, аналіз, контроль і регулювання.

Отже, інформаційну систему можна визначити з технічного погляду як набір взаємопов'язаних компонентів, які збирають, опрацьовують, накопичують і розподіляють інформацію, щоб підтримати прийняття рішень і

управління в організації. На додаток до підтримки прийняття рішень, координації й управління інформаційні системи можуть також допомагати менеджерам аналізувати проблеми, роблять видимими комплексні об'єкти й створюють нові вироби.

Нові інформаційні системи втілюють передовий досвід управлінських технологій. При цьому вони оптимізують бізнес-процеси відповідно до останніх досягнень теорії та практики менеджменту. У зв'язку з використанням таких технологій інформація стає важливою складовою виробничого процесу і витісняє з нього традиційні компоненти: природні ресурси, працю і капітал.

Порядок виконання роботи:

1. Створити новий проєкт VI
2. На фронтальну панель додати Boolean – Square Lad
3. Додати Numeric – numeric control (встановити розмір 36)
4. Додати індикатор

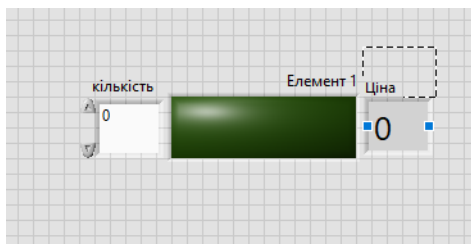


Рис.5.1. Виконання пунктів 1,2,3

5. Зайти в налаштування Lad. Змінити налаштування як показано на рис.5.2.

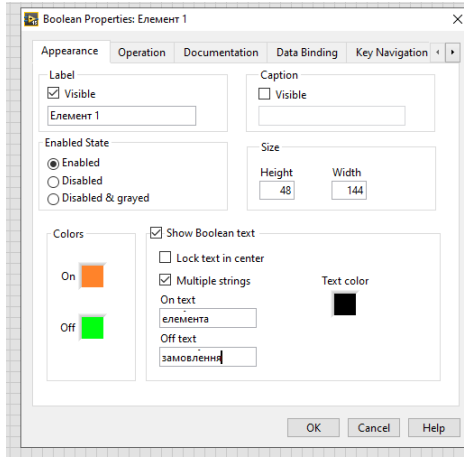


Рис.5.2. Налаштування Lad

6. Перейти в блок діаграму. Елемент 1 змінити на «контроль», вставити функцію Bool (0,1) меню Boolean – Bool to (0,1), В елементі «Ціна» встановити константу «5» рис.5.3.

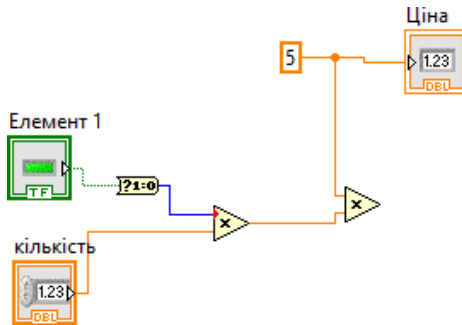


Рис.5.3. Підключення доданих елементів

7. Для того, щоб встановити мінімальну кількість 1 та максимальну 10 (Ви можете вводити свою максимальну кількість) натиснути ПКЛ на елементі фронтальної панелі «Кількість» та виставити налаштування, як показано на рис.5.4

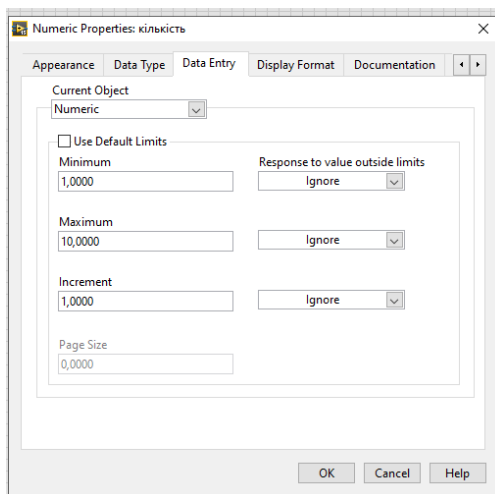


Рис.5.4. Налаштування елемента «Кількість»

8. Далі необхідно додати елементи для ще мінімум 4 товарів. Для цього скопіюйте елементи на фронтальній панелі та елементи на блок діаграмі, як показано на рис.5.5.

9. Для того, щоб об'єднати між собою ціни елементів додати на блок діаграму Compound Arithmetic із меню Boolean, та змінити його налаштування на «Add», приклад зображено на рис.5.6.

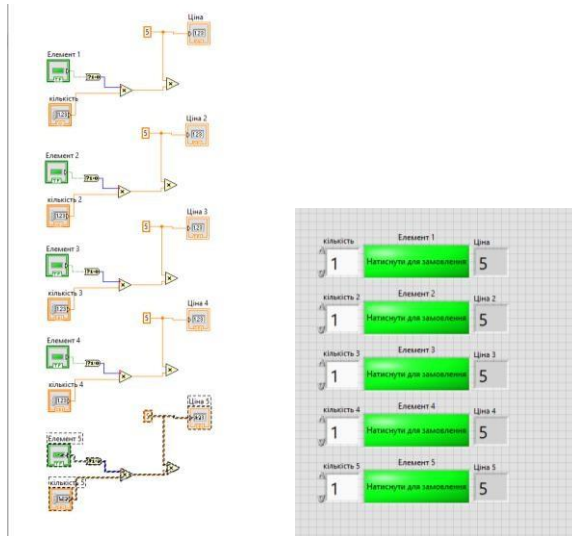


Рис.5.5. Відображення 5 різних елементів на панелях VI

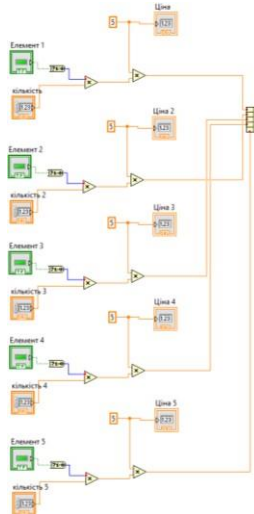


Рис.5.6. Підключення цін елементів в загальний блок

10. Наступний крок це створення індикатора, для цього натисніть ПКЛ Replace – Silver – Numeric – Numeric Indicator (Silver) Рис.5.7. Розмір встановити 36.

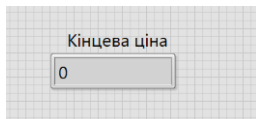


Рис.5.7. Вигляд індикатора

11. Змінити ціни в елементах. Для цього встановіть константи та введіть в них ціни, в методичних вказівках це 10,15,20 та 25. Ви можете ввести свої ціни на товар. Також додати на блок діаграму три кондиції для розгалуження, для цього натисніть ПКЛ Comparison – Select та три елементи Creator? Із того ж меню та підключити їх рис.5.8.

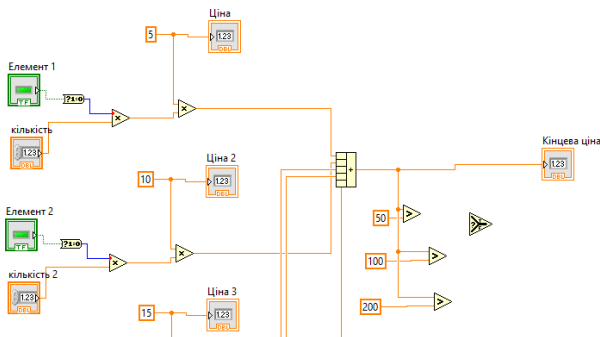


Рис.5.8 Схема підключення розгалуження

12. Додати структуру Case, підключити до неї елемент «Кінцева ціна» та підключити її як показано на рис.5.9.

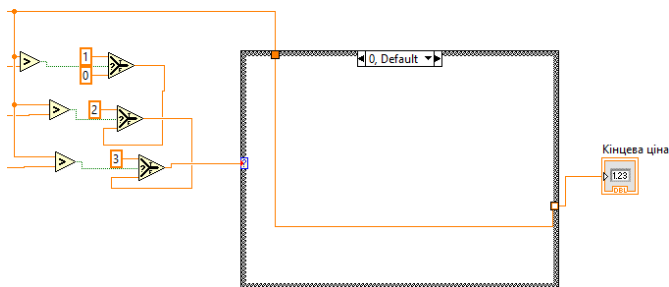


Рис.5.9 Створення структури для множинного вибору

13. Виконати підключення рис.5.10.

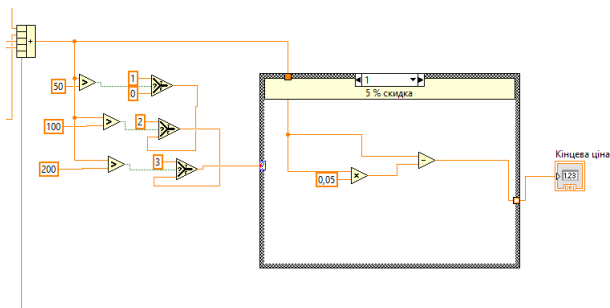


Рис.5.10 Вікно налаштування вибору «1»

14. Додати панель для відображення відсотків, для цього натиснути на ПКЛ «1» - Visible Items – Subdiagram Label, та підключити і налаштувати всі 4 вибори. Додати на фронтальну панель індикатор із меню Silver – String & Path – String Indicator (Silver), це буде

відображення знижки, підключити елемент «Знижка» та налаштувати його для кожного вікна вибору Рис.5.11.

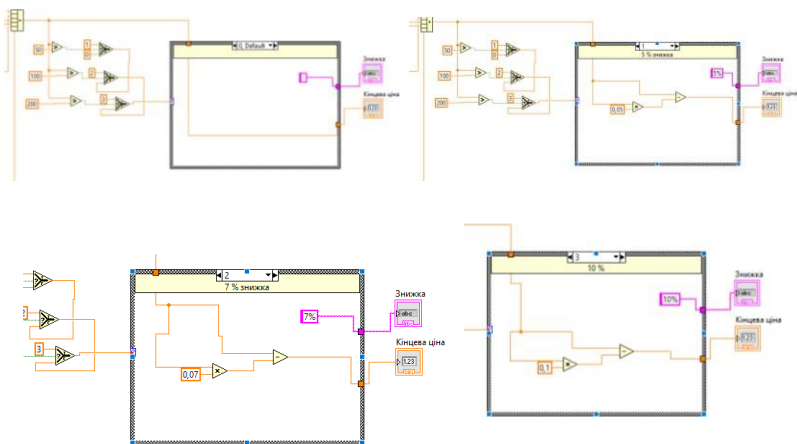


Рис.5.11. Налаштування вікон множинного вибору

15. Додати на блок діаграму фінальну кнопку із меню Silver – Boolean – Buttons – OK Button (silver) та налаштувати її як показано на рис.5.12.

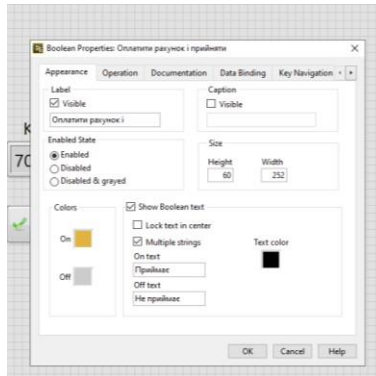
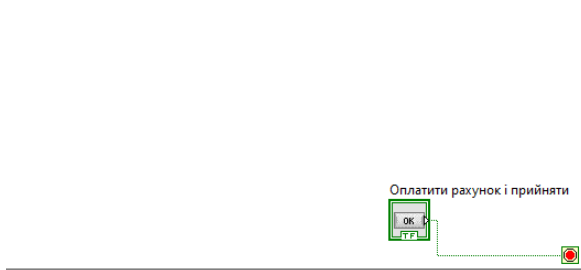


Рис.5.12. Налаштування кнопки

16. Додати створений проєкт в цикл While Loop, підключити кнопку із п.15 до циклу рис.5.13.



5.13. Підключення фінальної кнопки

17. Додати структуру Flat Sequence Structure навколо циклу While. Натиснути на ній ПКЛ Add Frame After та створити у вікні що з'явилося ще один While рис.5.14.

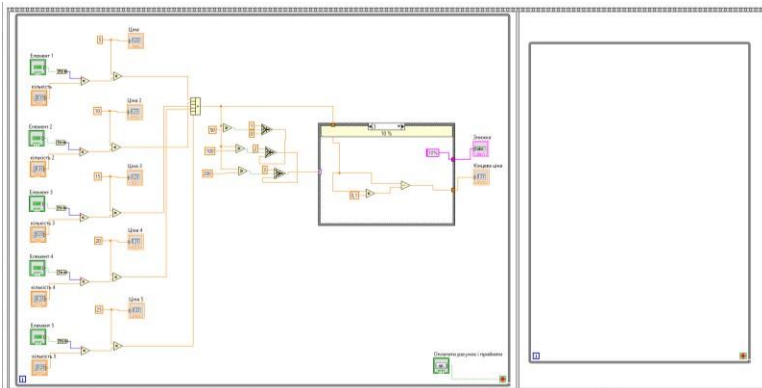


Рис.5.14. Створення Структур програми

18. Налаштувати новий цикл для відображення та зв'язку з фінальною кнопкою рис.5.15.

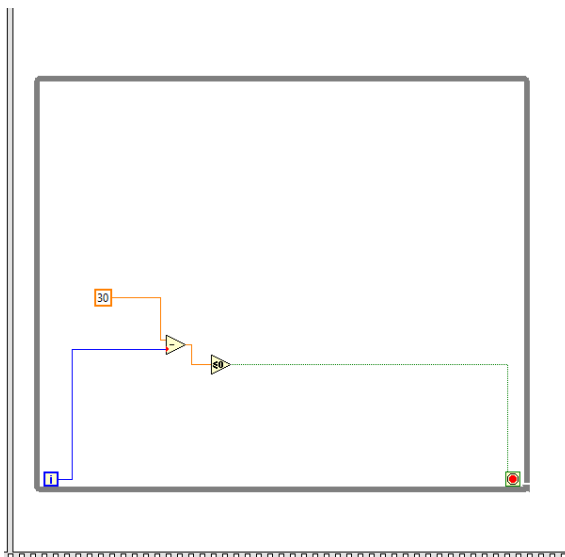


Рис.5.15 Налаштування циклу

19. Натиснути ПКЛ по елементу віднімання та створити індикатор рис.5.16.

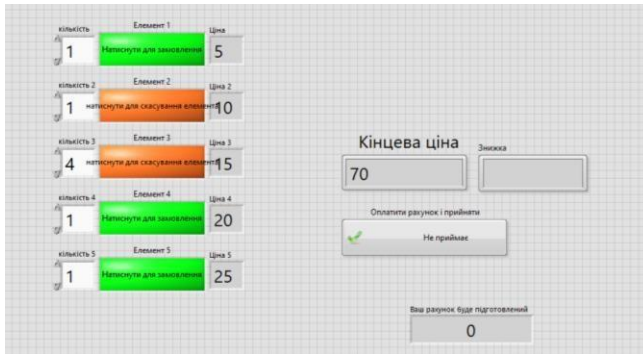


Рис.5.16. Відображення фронтальної панелі з індикатором

20. Додати час з меню Timing – Wait (ms), та встановити константи. Час додати в обидва цикли.

21. Творити «Local Variable» до кожного «Елементу» рис.5.17.

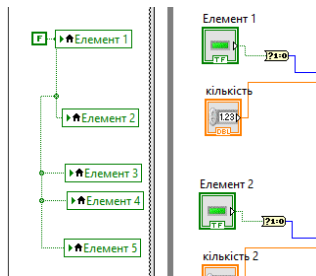


Рис.5.17. Відображення Local Variable

22. Для кнопки «Ваш рахунок» створити Local Variable та залишити константу 0 рис.5.20.

23. Для кнопки «Знижка» створити Create – property Node – Visible, ПКЛ- Visible – Change to Write , створити константу «Т» рис.5.18, те саме зробити з кожним вікном case.

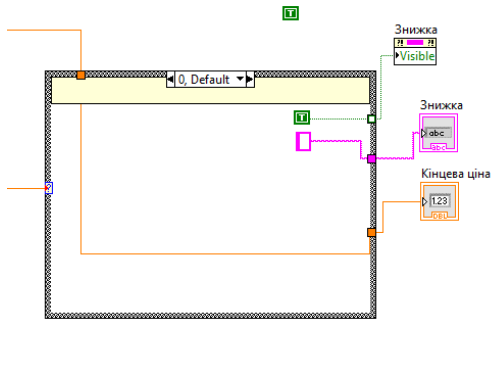


Рис.5.18. Підключення кнопки «Знижка»

24. Запустити та перевірити програму. Вікна кінцевого результату можна побачити на рис.5.19 та 5.20.



Рис.5.19. Фронтальна панель

25. Панель блок-діаграма виконаної програми знаходиться в в Додатку 1 Приклад А.

Варіанти до виконання лабораторної роботи:

Замінити назви «Елемент» на якісь товари, для прикладу ноутбуки, планшети, ОЗП чи інше на Ваш вибір , змінити ціни в блоках «Ціна», додати дизайн, та отримавши результат програми зробити знімок вікна фронтального та блок-діаграми, оформити звіт.

Вміст звіту

1. Тема, мета роботи.
2. Хід роботи.
3. Результати розв'язання наведених задач в порядку виконання згідно варіанта (скрін блоку-діаграм та фронтальної панелі).
4. Висновки про результат виконаної роботи.

Контрольні питання:

1. За якими ознаками можна класифікувати інформаційні технології ?

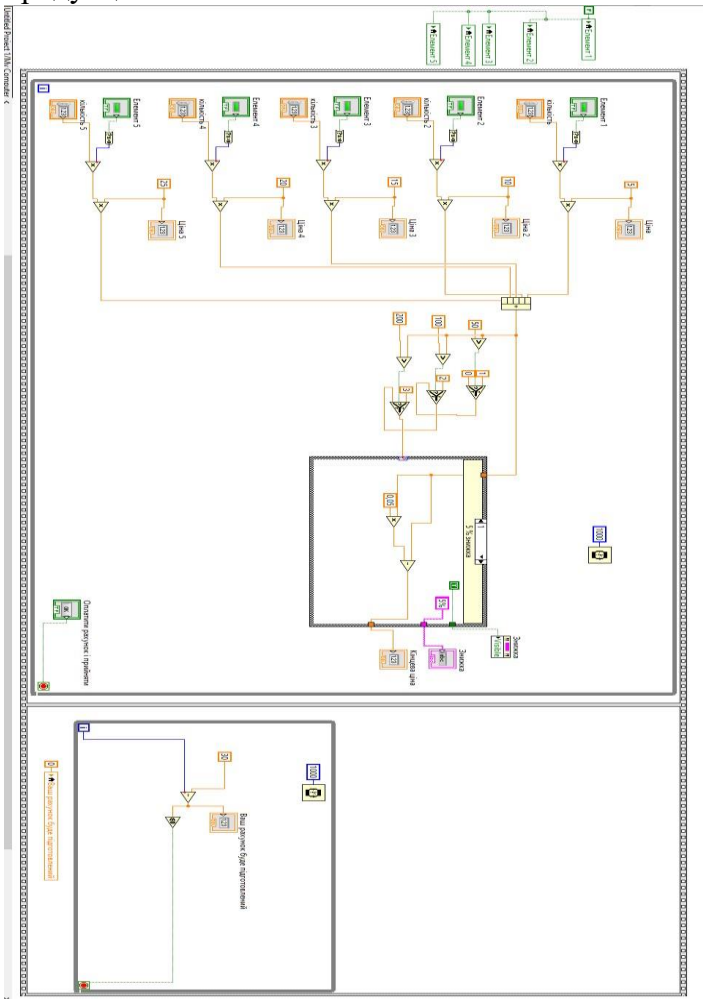
2. Що таке інформаційна система?
3. Що таке функція Bool ?
4. За допомогою якого елемента можна об'єднати між собою елемент «Ціни» ?
5. Як налаштувати вікна множинного вибору?

Список використаної літератури

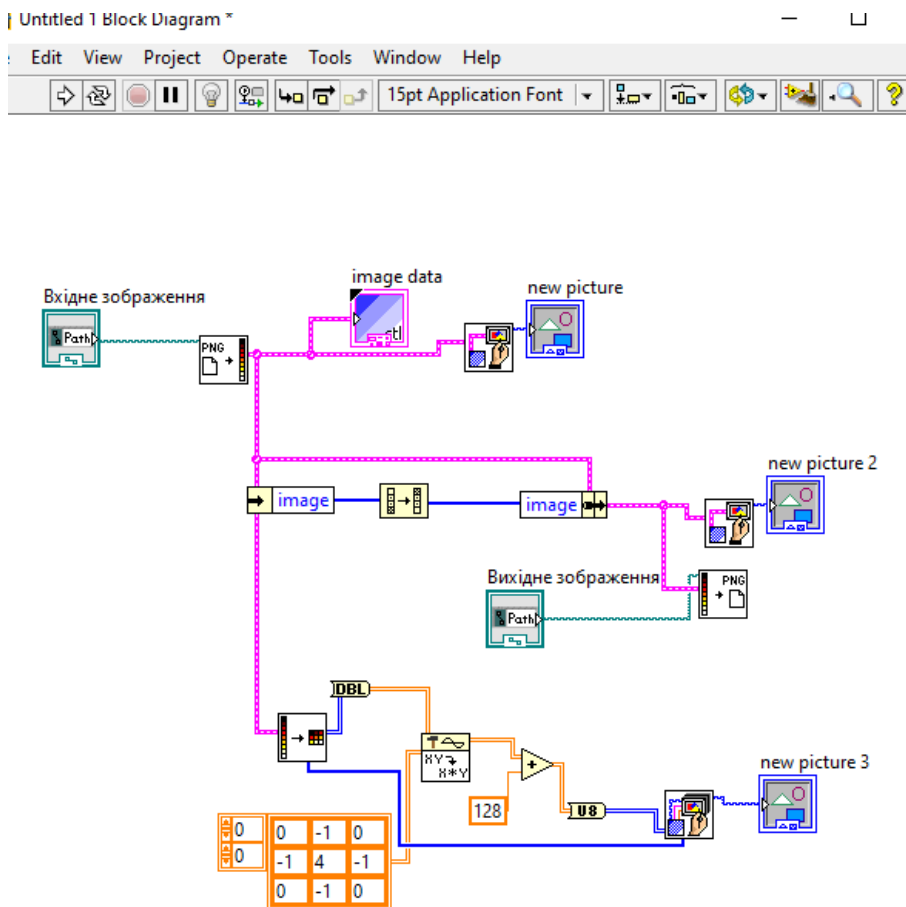
1. Harel D., Politi M. Modeling Reactive Systems with Statecharts. New York : McGraw Hill, 1998.
2. Jacobson I., Booch G., Rumbaugh J. The Unified Software Development Process. Reading. Mass. : Addison-Wesley, 1999.
3. Karampogias K. An In-Memory RDMA-Based Architecture for the Distributed Filesystem Computer Engineering and Networks Laboratory. Zurich : Computer Engineering and Networks Laboratory; ETII Zurich; IBM Research, 2012.
- 4.<https://www.ni.com/en-us/support/downloads/software/products/download.labview.html#487445>
5. Hard Real-Time Computing Systems Predictable Scheduling Algorithms and Applications. Authors: Giorgio C Buttazzo. Copyright: 2011.
6. Real-Time Systems Design Principles for Distributed Embedded Applications Authors: Hermann Kopetz. Copyright: 2011.
7. Hard Real-Time Computing Systems Predictable Scheduling Algorithms and Applications Authors: Giorgio C Buttazzo. Copyright: 2005.
8. Event-Triggered and Time-Triggered Control Paradigms. Authors: Roman Obermaisser. Copyright: 2005.

Додаток 1

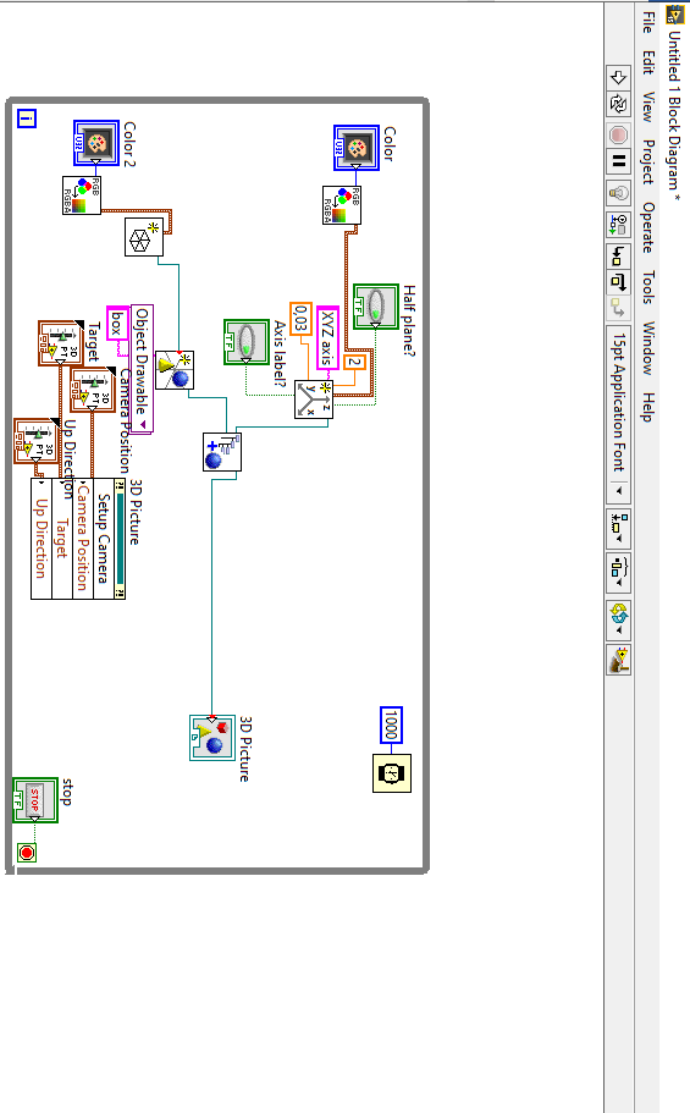
Приклад А. Панель блок-діаграми системи замовлення продукції в LabVIEW RT



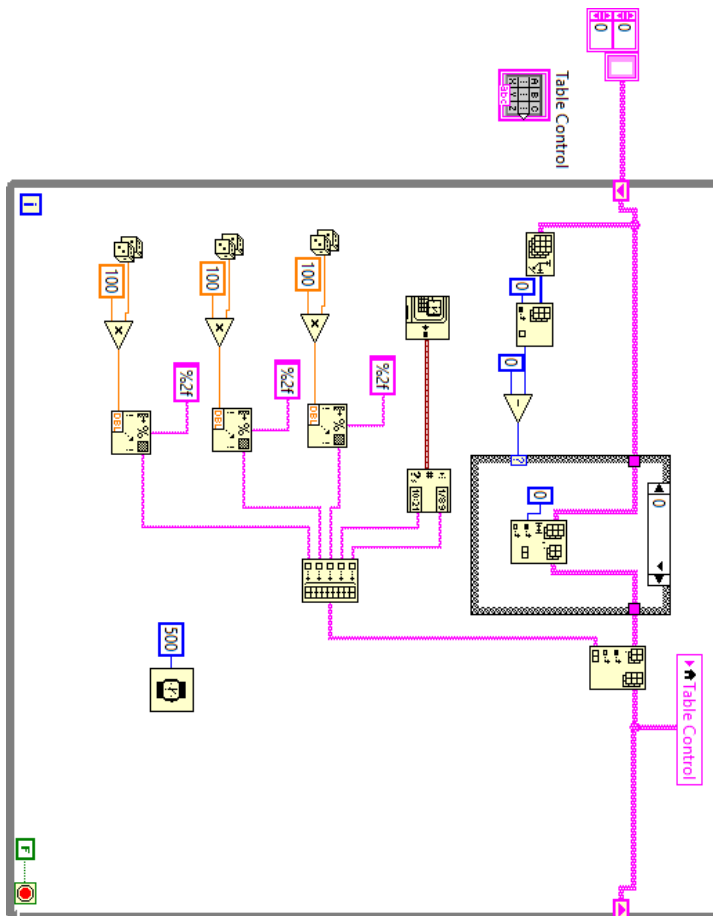
Приклад Б. Панель блок-діаграми обробки зображень в LabView.



Приклад С1. Панель блок-діаграма «Робота з тривимірними зображеннями та визначення координат в LabVIEW»



Приклад Д1. Панель блок-діаграма «Аналіз табличних даних для інформаційних систем в середовищі LabView»



Приклад К1. Панель блок-діаграма «Робота зі звуковими потоковими даними в режимі реального часу в середовищі LabView»

