

**МЕТОДИЧНІ ВКАЗІВКИ ДО
ЛАБОРАТОРНИХ РОБІТ**

з дисципліни «Комп'ютеризовані вимірювальні засоби»

для студентів за спеціальністю 175
«Інформаційно-вимірювальні технології»

затверджено
редакційно-видавничою
радою університета,
протокол №
від .

Харків НТУ «ХПІ» 2023

Методичні вказівки до лабораторних робіт з курсу «Комп'ютеризовані вимірювальні засоби» для студентів спеціальності 175 «Інформаційно-вимірювальні технології»/ За ред. В.М. Балєва, Частина 2 ,. – Харків: НТУ «ХПІ», 2023 – 39 с. –
Укр. мовою

Укладач: В.М. Балєв

Рецензент

Кафедра «Інформаційно-вимірювальні технології і системи»

ЛАБОРАТОРНА РОБОТА №9 ВІДОБРАЖЕННЯ ІНФОРМАЦІЇ НА ЕЛЕМЕНТАХ ІНДИКАЦІЇ СТЕНДУ «AVR – мікролаб»

1 Мета роботи: Отримання практичних навиків в організації обміну інформацією зі стендом «AVR – мікролаб».

2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6 та програмно-налагоджувальний стенд «AVR – мікролаб».

3 Основні характеристики стенду «AVR – мікролаб»

3.1 Багатоцільовий програмно-налагоджувальний стенд «AVR – мікролаб».

Стенд «AVR – мікролаб» призначений для моделювання та налагодження мікропроцесорних пристроїв на базі мікроконтролерів фірми Atmel і може використовуватися як для промислових розробок так і в учбових цілях.

В конструкції стенду реалізовано відкритий модульний принцип, тобто стенд являє собою набір найбільш поширених типів модулів різноманітних систем, які об'єднуються в різноманітні пристрої за допомогою плоских однотипних 10-жильних з'єднувальних кабелів. Кожен модуль має від одного до чотирьох рознімачів на які підключені усі функціональні зв'язки модулів.

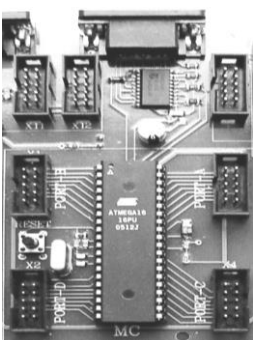
Така конструкція дозволяє конфігурувати інформаційно-вимірювальну або керуючу структуру як з використанням наявних на стенді модулів, так і шляхом підключення до них зовнішніх модулів або реальних об'єктів - сенсорів, виконуючих засобів, індикаторів, тощо. Більш того, такий принцип дозволяє моделювати і більш складні структури – мікропроцесорні системи різної конфігурації шляхом об'єднання декількох стендів.

До складу функціональних пристроїв стенду входять процесорний модуль, модуль статичної світлодіодної індикації, модуль динамічної світлодіодної індикації, модуль букво-цифрового дисплея LCD; модуль функціональної клавіатури, модуль аналого-цифрового перетворювача, модуль цифро-аналогового перетворювача, модуль драйвера послідовного інтерфейсу RS 232; модуль послідовного периферійного інтерфейсу SPI, модуль драйверів керування шаговим двигуном, модуль блоку вихідних силових ключів.

Підключення периферійних модулів до одного з чотирьох портів модуля мікроконтролера здійснюється за допомогою десяти контактних плоских кабелів (8 ліній портів та 2 проводи живлення).

3.2 Опис модулів програмно-налагоджувального стенду «AVR – мікролаб»

Модуль мікроконтролера є основою лабораторного стенду. Він складається з: 40-контактною панелі, в яку може бути встановлений будь який AVR мікроконтролер з ряду: ATmega16, ATmega8535, ATmega163, ATmega323 ATmega32 без будь яких комутацій та налаштувань, вбудованого стабілізатора напруги живлення, елементів програматора. Для підключення до чотирьох портів



вводу-виводу мікроконтролера периферійних пристроїв модуль має 4 уніфікованих рознімання. Окрім того, до складу модуля входять ланцюги синхронізації та скидання. Принципова схема модулю наведена на рисунку.

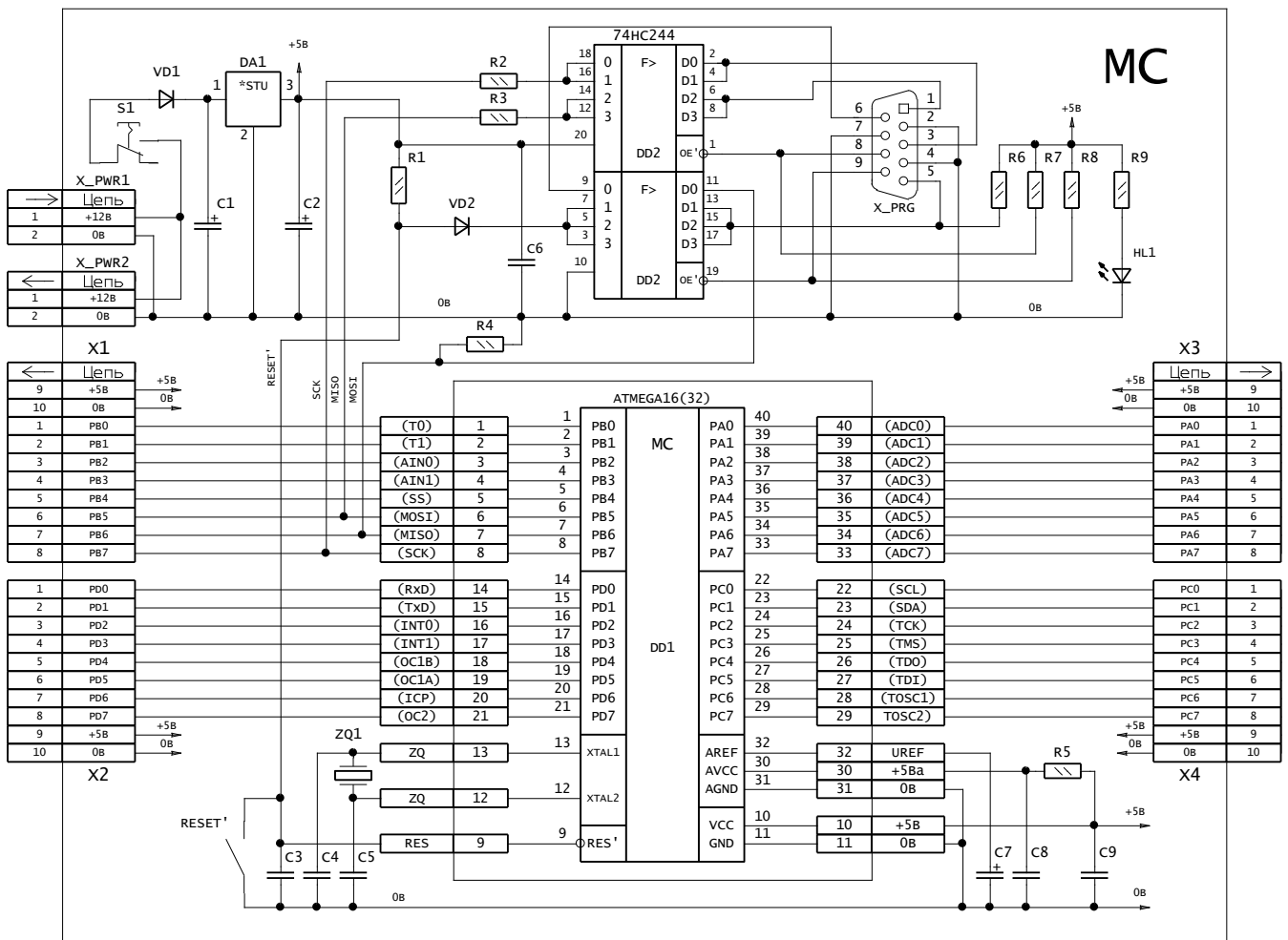


Схема електрична принципова модулю мікроконтролера.

Основою модуля є мікроконтролер DD1 з ланцюгами скидання R1,C3 та синхронізації ZQ1, C4, C5. Кварцовий резонатор ZQ1 визначає робочу частоту мікроконтролера – 8 МГц. Кнопка «RESET» дозволяє зробити перезапуск програми мікроконтролера в разі необхідності. Аналогова частина мікроконтролеру живиться від джерела живлення +5В через згладжувальний RC ланцюг R5,C8. Конденсатор C7 призначен для фільтрації напруги $U_{ref} = 2,56 \text{ В}$ на виході вбудованого в мікроконтролер джерела опорної напруги. На контакти гнізд X1...X4 виведен лінії чотирьох портів мікроконтролеру. Треба взяти до уваги що, молодшому розряду порту мікроконтролера відповідає молодший номер контакту рознімання. До старших контактів рознімань портів підведено напругу живлення +5В. За основу вбудованого в схему модуля мікроконтролера програматора взята рекомендована фірмою ATMEL схема внутрішньосхемного програматора (ISP) STK200/300 побудованого на мікросхемі шинного формувача DD2 т резисторах R2...R4, R6...R8. Резистори R4, R6...R8 підтягують «висячі» входи шинного формиувача DD2 до рівня напруги живлення, а резистори R2, R3 захищають виходи мікроконтролера

DD1 та шинного формувача DD2 при одночасній роботі програмтора та SPI інтерфейсу. Діод VD2 запобігає закорочуванню виходів шинного формувача DD2 на «землю» коли натискається кнопки «RESET». Живиться модуль мікроконтролера і усі модулі, які до нього можуть бути підключені, від вбудованого стабілізатора напруги, виконаного на мікросхемі DA1 та конденсаторах фільтра C1, C2. Діод VD1 захищає схему від випадкової переполюсовки напруги живлення на контактах роз'ємів живлення X_PWR1 або X_PWR2.

Модуль статичної світлодіодної індикації можна використовувати для індикації стану виводів портів мікроконтролера. До складу модуля входять дві групи з 8 (за кількістю виводів портів) світлодіодів. При цьому одна група світлодіодів HL2...HL9 (зелених) підключається через роз'єм X5 та струмообмежуючі резистори R10...R17 безпосередньо до виводів порту, а друга HL10...HL17 (червоних) через буфер DD3 – для зменшення навантаження на лінії порту. Друга група світлодіодів може бути підключена до виводів порту контролера паралельно до іншого пристрою. З цією метою в схему модуля введено два «запаралелених» роз'єми X6 і X7.

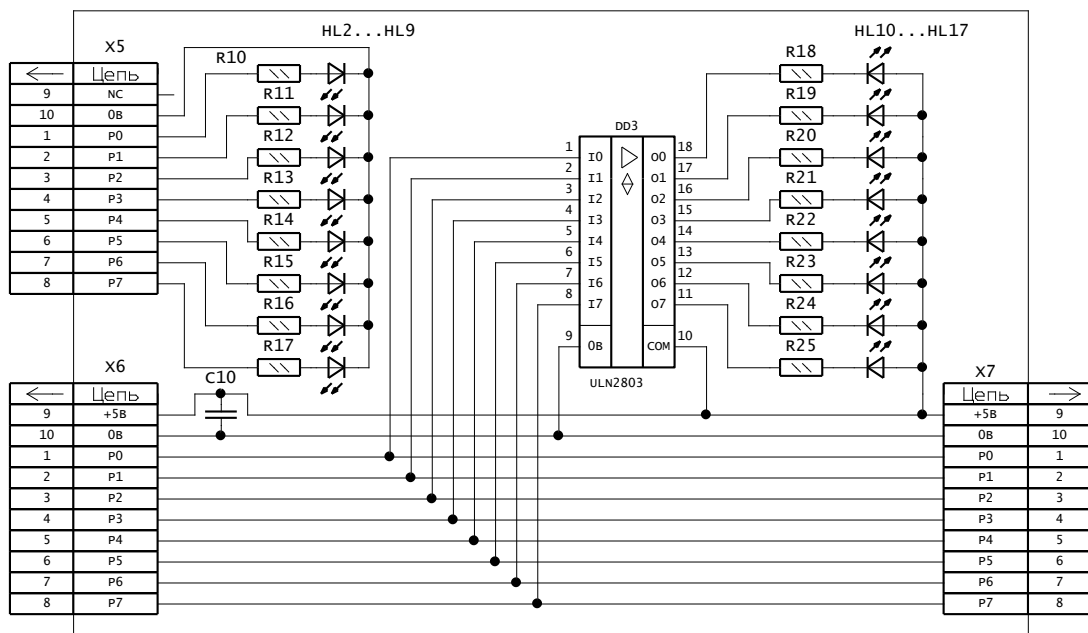
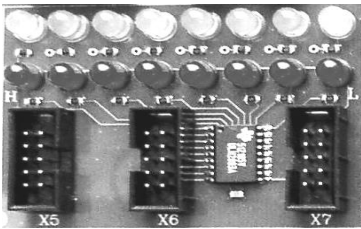
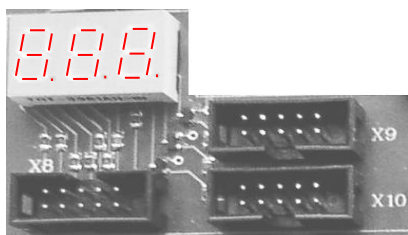


Схема електрична принципова модуля статичної світлодіодної



Модуль динамічної світлодіодної індикації є 3-х розрядним семисегментним дисплеєм з ключами управління, що дозволяють відображення 3-х розрядних чисел або стилізованих символів. Модуль має три роз'єми для підключення до портів контролера. Основою модуля є трирозрядний семисегментний індикатор HL19. Світлодіоди індикатора включені за схемою із загальним катодом. Таким чином, для засвічення символу на індикаторі необхідно через роз'єм X8 подати на відповідні світлодіоди напругу високого рівня від одного з портів

мікроконтролера, а на катодні виводи індикатора рівень логічного нуля від іншого порту мікроконтролера. Три виведення катодів індикатора HL19 можуть бути підключені до «нуля» через роз'єм X9 або подачею потенціалу високого рівня, прикладеного до одного з транзисторів VT1...VT3 через роз'єм X10. Резистори R26...R33 призначені обмеження струму через світлодіоди індикатора HL19, а резистори R34...R36 визначають струми баз транзисторів VT1...VT3. Для забезпечення повного використання портів мікроконтролера старші п'ять сигнальних ліній роз'ємів X9 і X10 запаралелені.

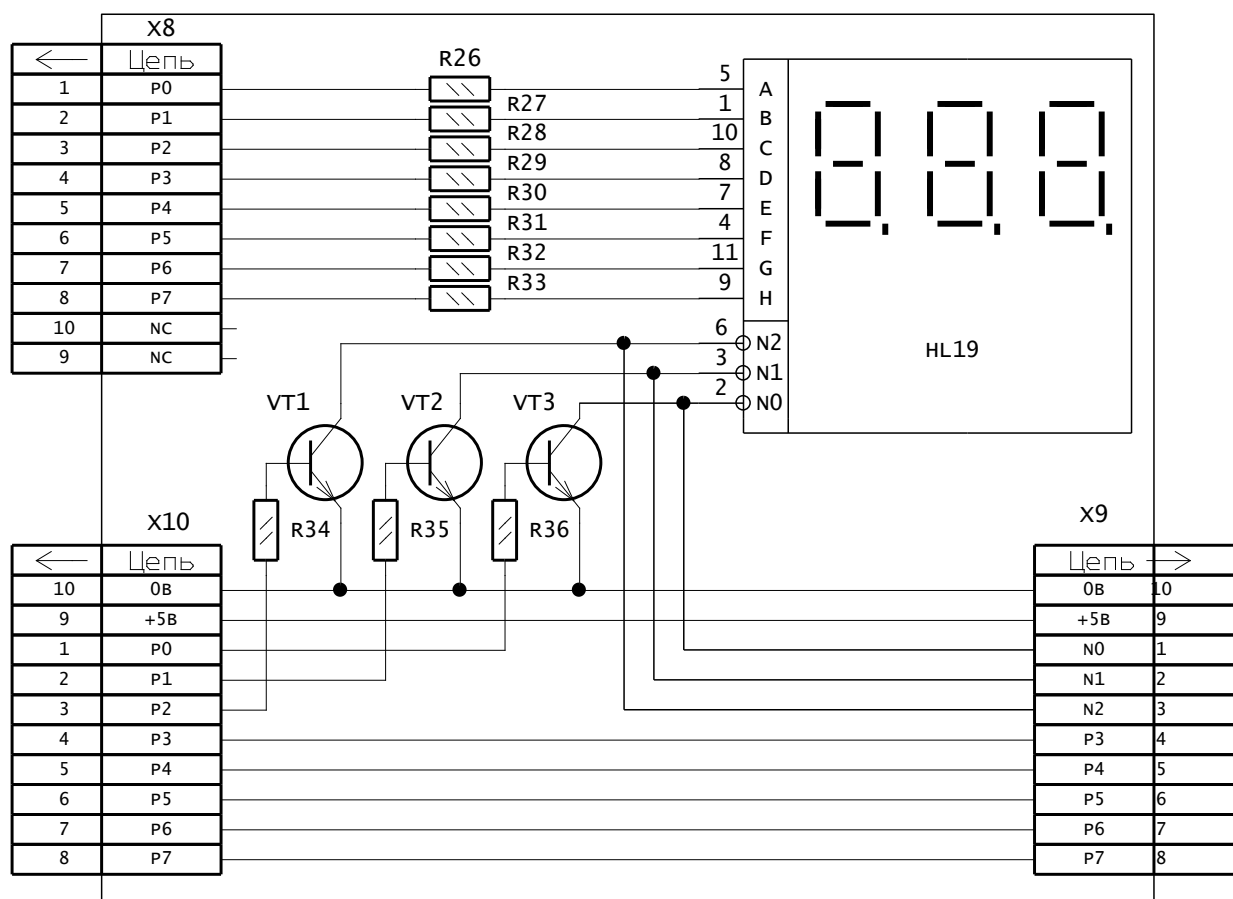


Схема електрична принципова модуля динамічної світлодіодної індикації

3.3 Перелік команд, що використовуються при спілкуванні зі стендом «AVR – мікролаб»

У мікроконтролер стенду «защита» програма, що дозволяє отримати доступ до портів вводу/виводу, вбудованого АЦП та таймерів-лічильників. За замовчуванням порти мікроконтролера налаштовані: порт А входи (аналогові), порт В виходи (цифрові), порт С входи (цифрові), порт D виходи (цифрові).

Обмін даними з мікроконтролером стенду здійснюється за інтерфейсом RS232. Швидкість передачі становить 38400 біт/с, формат даних що передаються – 8 біт, біт парності не використовується, стоповий біт 1. Для спілкування з мікроконтролером використовується система двобайтних команд, наприклад, \$BB\$AC, де числа представлені в шістнадцятковому кодуванні.

Для установки ліній порту В необхідно передати першим байт виду \$BB (у шістнадцятковому коді, в десятковому коді це еквівалентно 187). Якщо перший прийнятий байт \$BB, то значення другого байта записується в порт В, після 10 мкс очікування зчитується стан вхідних ліній порту і результат повертається у вигляді двобайтного коду де перший байт \$BB, а другий байт стан вхідних ліній порту В (\$BB \$xx). Таким чином, проаналізувавши відповідь, можна судити про виконання відправленої команди.

Для встановлення ліній порту С необхідно передати першим байт виду \$CC (у шістнадцятковому коді, у десятковому коді це еквівалентно 204). Якщо перший прийнятий байт \$CC, то значення другого байта (записується в порт С, після 10 мкс очікування зчитується стан вхідних ліній порту С і результат повертається у вигляді двобайтного коду, де перший байт \$CC, а другий байт стан вхідних ліній порту С (\$CC\$xx) Таким чином, проаналізувавши відповідь, можна зробити судження про виконання відправленої команди.

Для читання ліній порту С необхідно передати першим байт виду \$CA (читання ліній порту С), а другим довільне значення. Якщо перший прийнятий мікроконтролером байт \$CA, то значення другого байта ігнорується, зчитується стан вхідних ліній порту С і результат повертається у вигляді двобайтного коду де перший байт \$CA, а другий байт стан вхідних ліній порту С (\$CA\$xx).

4. Хід роботи

1) Створіть віртуальні прилади згідно з індивідуальними завданнями. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:
- Лицьову панель розроблених приладів, з відображеною інформацією згідно з індивідуальним завданням;

- Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

1) Чим відрізняється статичний та динамічний режим індикації?

2) Розкажіть про систему команд, яка використовується для обміну даними зі стендом «AVR – мікролаб»?

3) Поясніть обмеження по діапазону при відображенні чисел на індикаторі з трьома розрядами, як можна розширити діапазон значень, що відображаються?

6. Варіанти завдань

Завдання

1. Розробити програму що дозволяє реалізувати 2 різних режими роботи лінійки з 8 світлодіодів розташованих на лицьовій панелі віртуального приладу та в

модулі статичної індикації стенду, режим роботи вибирається на лицьовій панелі віртуального приладу.

Перший режим – режим ручного керування, увімкнення або вимкнення будь-якого зі світлодіодів призводить до увімкнення/вимкнення відповідного світлодіода на стенді «AVR – мікролаб». Світлодіоди (роз'єм X5) на стенді повинні бути підключені до порту С мікроконтролера.

Другий режим – послідовне включення (парний номер за списком у журналі) світлодіодів зі стану усі вимкнені, або послідовне вимкнення (непарний номер за списком у журналі) зі стану увімкнено. Увімкнення/вимкнення світлодіодів у даному режимі має імітувати збільшення/зменшення суми вагових коефіцієнтів від 0 до 255 (від 255 до 0). Найправіший світлодіод має вагу 2^0 , а найлівіший вагу 2^7 .

2. Розробити програму керування трьома розрядами динамічного світлодіодного індикатора стенду «AVR – мікролаб» що дозволяє відображати числа в діапазоні від 0,01 до 999. Три розряди семисегментного індикатора також розмістити на передній панелі приладу, використовуючи раніше розроблену підпрограму. Входи управління сегментами індикатора стенду «AVR – мікролаб» (роз'єм X8) повинні бути підключені до порту мікроконтролера, входи управління розрядами індикатора (роз'єм X10) підключити до порту С мікроконтролера. Команди управління динамічним індикатором повинні послідовно включати один із розрядів трирозрядного індикатора і подавати на входи управління сегментами інформацію, що відповідає активному, у кожний конкретний момент, розряду.

Для спрощення програми слід доопрацювати раніше розроблену підпрограму з метою додавання додаткового виходу, що дозволяє виводити код, відповідний сумі вагових коефіцієнтів включених світлодіодів семисегментного індикатора для передачі на стенд «AVR – мікролаб». Включену десяткову точку слід враховувати шляхом збільшення сформованого коду на 128, що відповідає вазі десяткової точки на семисегментному індикаторі.

Значення коду, що відображається, задавати за допомогою цифрового контролю розміщеного на лицьовій панелі приладу.

3. Налаштувати програми та навести у звіті лицьову панель та діаграму.

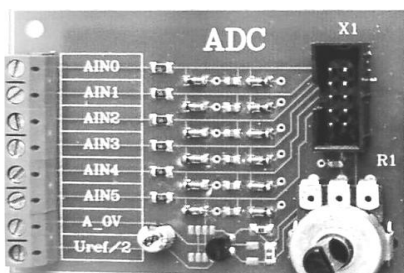
ЛАБОРАТОРНА РОБОТА №10 РОЗРОБКА ТРИХАНАЛЬНОГО ВОЛЬТМЕТРА НА БАЗІ СТЕНДА «AVR – мікролаб»

1 Мета роботи: Отримання практичних навиків зі створення комп'ютеризованих засобів вимірювань на базі стенда «AVR – мікролаб».

2 2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6 та програмно-налагоджувальний стенд «AVR – мікролаб»

3 Необхідні довідкові дані по роботі з вбудованим АЦП

3.1 Аналого-цифровий перетворювач стенда «AVR – мікролаб».



Модуль аналого-цифрового перетворювача організований у вигляді восьми каналів десятирозрядного аналого-цифрового перетворення з використанням наявного в контролері АЦП порозрядного врівноваження. При цьому зовнішні напруги можуть бути наступними стандартними нормованими рівнями 0..5В; 0..2,5В.

Для зв'язку з портом контролера модуль має роз'єм X16, а для підключення зовнішніх сигналів використовуються 8 гвинтових затискачів. Принципова схема модуля наведено на рисунку.

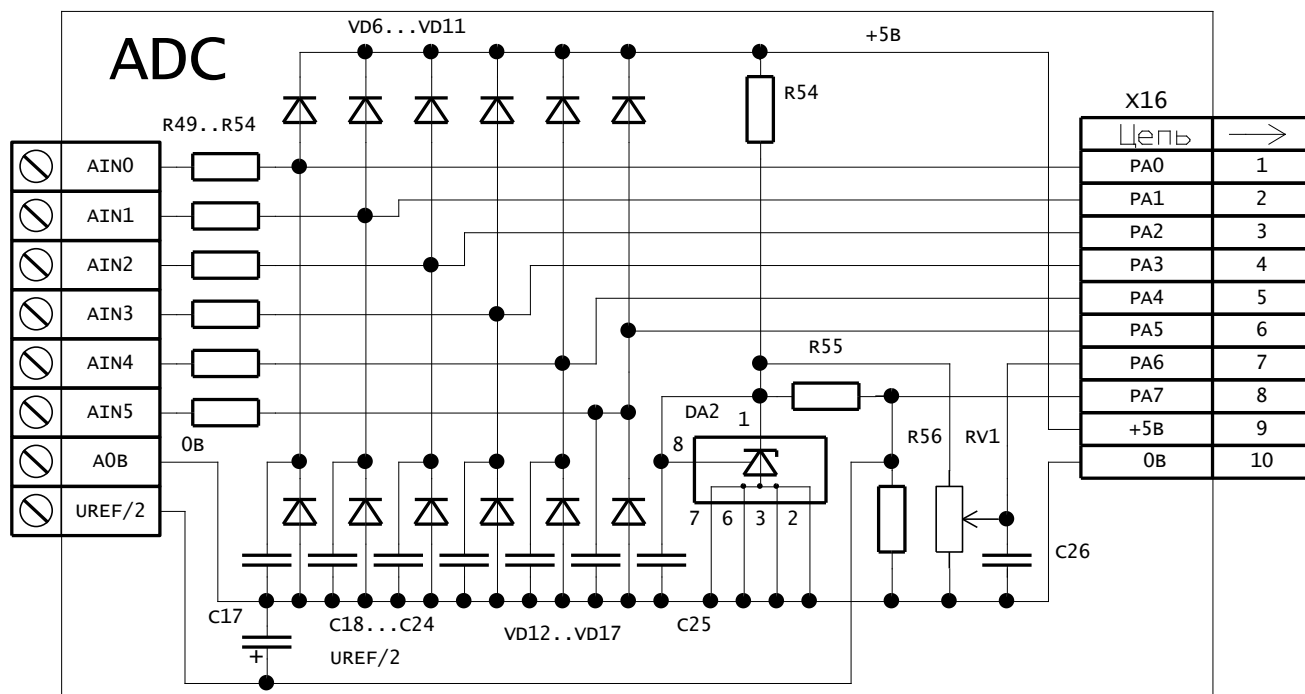


Схема електрична принципова модуля аналого-цифрового перетворювача

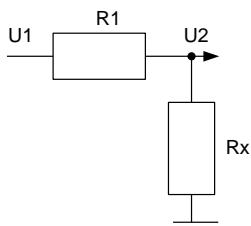
Схема модуля організована таким чином, що шість молодших виводів АЦП підключені через резистори R49...R54 призначені для обмеження вхідного струму безпосередньо до гвинтових затискачів модуля. На сьомий вхід АЦП (РА6) подано

напругу з повзунка змінного резистора RV1, а до восьмого входу підведена напруга UREF/2.

Модуль складається з дванадцяти діодів VD6...VD17, конденсаторів C17...C26, резисторів R49...R56 та мікросхеми джерела опорної напруги DA2. Діоди VD6...VD17 спільно з резисторами R49...R54 які обмежують струм оберігають входи АЦП від переполюсування та перевищення рівня вхідного сигналу ($U_{вх} < 5В$). Крім того, резистори R49...R54 та конденсатори C18...C24 утворюють фільтри низьких частот, що знижують рівень високочастотних перешкод на входах АЦП. Джерело опорної напруги ($U_{REF} = 2,5В$) реалізовано на мікросхемі DA2, баластному резисторі R54 та конденсаторі фільтра C25. Введення в схему джерела опорної напруги обумовлено необхідністю отримання стабільної напруги для живлення змінного резистора RV1 та отримання напруги зсуву $U_{CM} = U_{REF}/2 = 1,25В$. Підключення до будь-якого входу АЦП джерела змінної напруги щодо напруги зсуву дає можливість «оцифровувати» змінну напругу, амплітуда якої не перевищує 1,25 В.

3.2 Використання вольтметра для визначення значення невідомого опору

Знання закону Ома дозволяє скористатися вольтметром з метою визначення значення невідомого опору R_x за умови, що у нашому розпорядженні є джерело живлення і відомий опір R_1 . Для цього потрібно зібрати схему дільника напруги та виміряти значення напруги U_1 і U_2 , а потім, знаючи значення відомого опору R_1 , розрахувати значення невідомого опору R_x . Струм у ланцюгу дільника визначається за формулою $I = \frac{U_1}{R_1 + R_x}$,




напруги на невідомому опорі складе $U_2 = R_x I$. Звідси можна записати вираз значення невідомого опору, $R_x = \frac{U_2 R_1}{U_1 - U_2}$. Таким чином, вимірявши значення напруги U_1 і U_2 , і знаючи значення опору R_1 можна визначити значення невідомого опору, використовуючи метод непрямих вимірювань.

3.3 Перелік команд які використовуються при обмінні інформацією зі стендом «AVR – мікролаб» для доступу к вбудованому АЦП

Обмін даними з мікроконтролером стенду здійснюється з використанням інтерфейсу RS232. Швидкість передачі становить 38400 біт/с, формат даних що передаються – 8 біт, біт парності не використовується, стоповий біт 1. Для спілкування з мікроконтролером використовується система двобайтних команд, наприклад, \$BB\$AC, де числа представлені в шістнадцятковому кодуванні.


Модуль аналого-цифрового перетворювача стенда має бути підключений перемичкою до порту мікроконтролера А. Для доступу до вбудованого мікроконтролера АЦП використовується команда виду \$AA\$nn, де nn номер каналу АЦП, що приймає значення від 0 до 7.

На функціональній діаграмі запуск АЦП на перетворення виглядає як запис двобайтної символічної змінної до послідовного порту (на малюнку представлений

 варіант звернення до сьомого каналу АЦП), при цьому слід пам'ятати, що команди повинні бути представлені в шістнадцятковій системі числення, і, отже, має бути обраний режим показу символічних даних Hex Display.

Якщо перший прийнятий мікроконтролером байт дорівнює \$AA, перевіряється другий прийнятий байт. Якщо він не перевищує 7, то провадиться запуск АЦП каналом номер nn і десятибітний результат перетворення повертається у вигляді двобайтного коду виду \$mm\$nn.

Оскільки у відповідь АЦП відправляє двобайтну відповідь, то наступним оператором після запуску АЦП на перетворення має бути оператор зчитування відповіді, який виглядає як оператор читання з послідовного порту. Кількість зчитуваних байт має дорівнювати кількості байт у відповіді АЦП – 2 байти.

 Відповідь АЦП із послідовного порту буде отримано у вигляді двох символів рядкової змінної. Для подальшого обчислення отримані дані слід перетворити на числові дані з використанням оператора String To Byte Array, який перетворює символічну змінну в масив байт.

Відповідь АЦП приходить у вигляді двох байт (10 біт): у молодшому байті значущі всі біти (nnnnnnnn), а у старшому байті значущі лише молодші два біти (xxxxxxnn), тут n приймає значення 0 або 1. Для доступу до елементів масиву з двох байт слід використовувати оператор Index Array, за нульовою адресою розташовується молодший байт, а по одиничній адресі розташовується старший байт відповіді АЦП. Для визначення результату перетворення як десяткового числа слід пам'ятати, що одиниця молодшого розряду нульового байта має вагу $2^0=1$, а одиниця молодшого розряду першого байта має вагу $2^8=256$. Результат виходить підсумовуванням значень нульового та першого байта перетворення з урахуванням вагових коефіцієнтів.

Для перерахунку десяткового коду АЦП в значення напруги необхідно визначити крок квантування, який залежить від діапазону перетворення, в нашому випадку (0-5) В і розрядності АЦП (10 розрядів).

4. Хід роботи

1) Створіть віртуальні прилади згідно з індивідуальними завданнями. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:
- Лицьову панель розроблених приладів, з відображуваною інформацією згідно з індивідуальним завданням;
- Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

1) Як можна ініціювати перетворення АЦП вбудованого в мікроконтролер?

2) Скільки каналів має вбудований в мікроконтролер АЦП, скільки їх доступні для вимірювання довільних сигналів, яка розрядність АЦП?

3) Який діапазон вимірюваної постійної напруги АЦП? Чи можна вимірювати змінну напругу і якщо так, то в якому діапазоні?

6. Варіанти завдань

1. Розробити триканальний вольтметр що дозволяє виміряти значення напруги джерела опорної напруги підключеного до сьомого каналу АЦП, виміряти значення напруги на движку змінного опору RV1, підключеного до шостого каналу АЦП і значення напруги U2 (перший канал АЦП) використовуваного для знаходження значення невідомого опору. Як напруга U1 (живлення дільника) використовується джерело опорної напруги.

Результати вимірювань напруги по шостому та сьомому каналу АЦП, а також різницю напруги на сьомому та шостому каналах, відобразити на лицьовій панелі віртуального приладу з використанням аналогових індикаторів (3 індикатори).

Результат непрямого вимірювання опору відобразити за допомогою цифрового індикатора (відомим виступає опір R1 номіналом 3,3 кОм).

Чотири старші розряди (наприклад, 1.235) результату вимірювання по шостому каналу АЦП відобразити на трирозрядному динамічному індикаторі стенду «AVR – мікролаб» у вигляді рядка символів, що біжать.

2. Налаштувати програму та навести у звіті лицьову панель та діаграму.

ЛАБОРАТОРНА РОБОТА №11

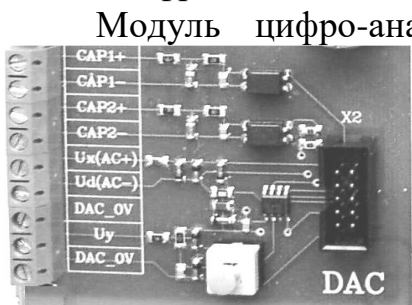
РОЗРОБКА ГЕНЕРАТОРІВ СИНУСОІДАЛЬНОГО, ТРИКУТНОГО ТА ПИЛКОПОДІБНОГО СИГНАЛІВ НА БАЗІ СТЕНДУ «AVR – мікролаб»

1 Мета роботи: Отримання практичних навиків зі створення комп'ютеризованих засобів формування різноманітних сигналів спеціальної форми на базі стенда «AVR – мікролаб».

2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6 та програмно-налагоджувальний стенд «AVR – мікролаб».

3 Необхідні довідкові дані по роботі з вбудованим АЦП

3.1 Цифро-аналоговий перетворювач стенду «AVR – мікролаб»



Модуль цифро-аналогового перетворювача побудовано на інтегральній мікросхемі DA3 типу TLC5615 фірми Texas Instruments, який представляє собою десяти розрядний ЦАП з послідовним SPI-інтерфейсом, який може бути підключено до порту «В» контролера. Схема може працювати в режимі ЦАП зі стандартною вихідною напругою 0..5 В або в режимі керуемого дільника зовнішньої напруги. Вибір режиму здійснюється за

допомогою перемикача S2, встановленого на модулі. Окрім того, можливе використання модуля в якості джерела опорної напруги при моделюванні аналого-цифрових структур з використанням аналогового компаратора та таймера лічильника, вбудованих в контролер. Модуль має один роз'єм X17 для підключення до порту контролера, а для зовнішніх сигналів – 9 гвинтових затискачів. К затискачам модуля також підведені два канали вводу дискретних (імпульсних) сигналів з гальванічною розв'язкою, що дозволяє безпечно використовувати функцію захоплення таймерів-лічильників, вбудованих в мікроконтролер. Принципова схема модуля приведена на рисунку нижче.

Канали вводу дискретних сигналів складаються з транзисторних оптронів VU1, VU2, стабілітронів VD18, VD19, резисторів R57...R62 и конденсаторів C27, C28. Оптрони VU1, VU2 здійснюють гальванічну розв'язку входних імпульсних сигналів від входів мікроконтролера. Стабілітрони VD18, VD19 з резисторами R57, R58 обмежують входні імпульсні сигнали на рівні 4,7В, а резистори визначають вхідний струм світлодіодів оптронів. Резистори R61, R62 являють собою колекторне навантаження транзисторів оптронів.

Елементи VD20...VD23, R63 и C29 призначені для захисту входів аналогового компаратора мікроконтролера від небезпечних рівнів входних сигналів, які можуть бути підключені до гвинтових затискачів Ux(AC+) и Ux(AC-).

Резистивний дільник напруги R65, R66 формує з напругі живлення ($U_{\text{жив}} = 5\text{В}$) один рівнів опорної напруги ЦАП ($U_{\text{REF}} = 2,5\text{В}$), який визначається положенням

перемикача S2. В якості другого рівня може використовуватися напруга, прикладена до гвинтового затискача U_y ($0 < U_y < 2,5\text{В}$). В цьому випадку ЦАП перетворюється в керуємий дільник напруги U_y. Клема U_x (АС-) є виходом ЦАП.

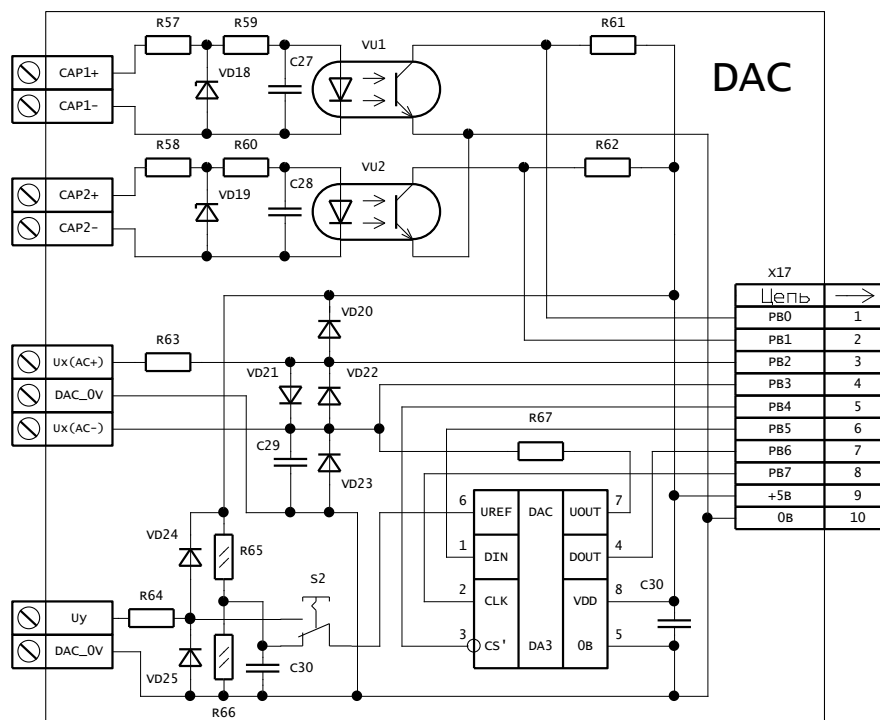
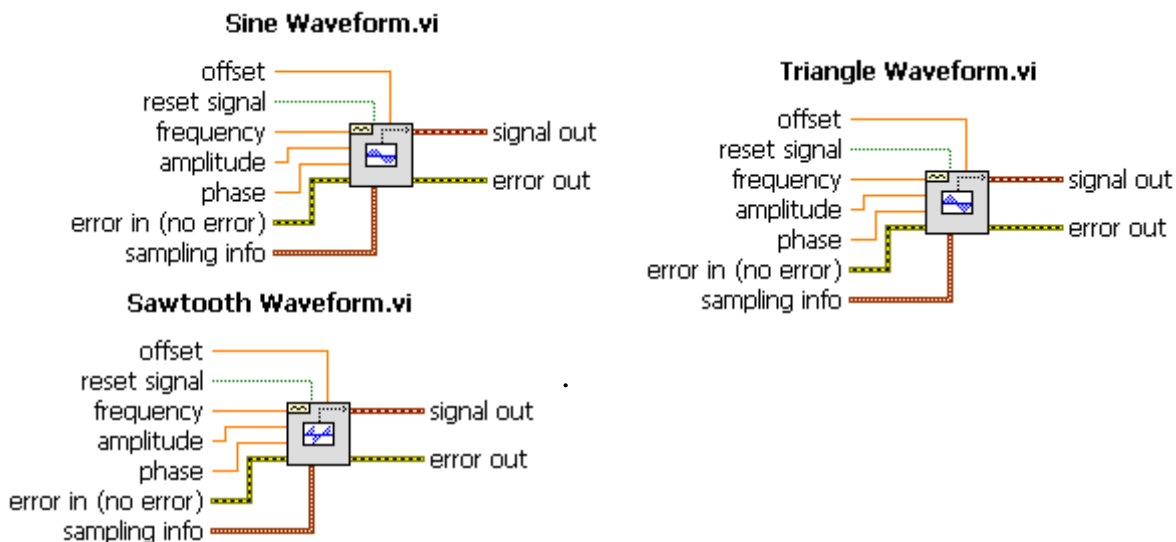


Схема електрична принципова модуля цифро-аналогового перетворювача

3.2 Функції генерування сигналів

Для генерування потрібних детермінованих та випадкових сигналів із заданим набором параметрів використовуються функції які розташовані в Functions-Signal Processing- Waveform Generation.



На рисунках наведені іконки синусоїдального, трикутного та пилкоподібного сигналів. Вони однотипні і розрізняються тільки формою вихідного сигналу. В

якості основних даних використовуються частота, амплітуда, фаза та інформація щодо частоти та кількості відліків які будуть генеровані.

Для знайомства з цими функціями створіть новий віртуальний прилад і розмістіть на блок-діаграмі три різних функції для генерування сигналів.

Для кожного з видів сигналів створіть на лицьовій панелі приладу зону, в якій будуть задаватися параметри одного з сигналів, а відповідні іконки на диграмі підключить до функції генерування сигналу. Частоту дискретизації та кількість відліків встановіть рівними 500 для усіх типів сигналів.

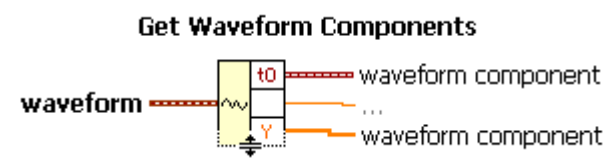


Результат генерування сигналів відобразить за допомогою одного зі індикаторів Waveform Graph.

Помістіть елементи діаграми в структуру While Loop запустіть програму в однократному режимі і спробуйте змінювати частоту та амплітуду сигналів що генеруються.

Впевнитесь що картинка на індикаторі змінюється. Розроблений віртуальний прилад дозволив

сформуванню віртуальні сигнали які живуть тільки в комп'ютері. Для того, щоб "оживити" ці сигнали ми можемо використати модуль цифроаналогового перетворювача стенду «AVR-мікролаб». Оскільки модуль ЦАП на стенді одноканальний, то ми можемо "оживити" в кожний конкретний момент часу тільки один сигнал, за нашим вибором. Для цього нам необхідно знати значення амплітуди конкретних відліків сгенерованих функціями обраними нами сигналів. Вихідний сигнал функції несе в собі інформацію про початковий



момент часу t_0 функції яка генерується, про період дискретизації dt та значеннях амплітуди відліків Y . Доступ до цієї інформації ми можемо отримати за

допомогою функції Get Waveform Components, розміщених в Functions-Programming-Waveform. Значення амплітуди відліків надається у вигляді масиву чисел, доступ до елементів масиву здійснюється за допомогою оператора Index Array. Для "оживлення" отриманих значень сигналу, який був згенерований, необхідно його перетворити до вигляду зрозумілому для ЦАП, а саме в однобайтний код в діапазоні від 0 до 255. Тоді мінімальному значенню сигналу має відповідати код 0, а максимальному значенню має відповідати код 255. При цьому слід пам'ятати, що амплітуда сигналу який «оживлюється» не має перевищувати 2,5 В. Процес "оживлення" будемо проводити шляхом передачі інформації про значення амплітуди згенерованого сигналу через COM порт.

3.3 Перелік команд які використовуються під час спілкування зі стендом «AVR – мікролаб» для доступу до модулю ЦАП.

Обмін інформацією з мікроконтролером стенду здійснюється по інтерфейсу RS232. Швидкість передачі складає 38400 біт/с, формат даних що передаються – 8 біт, біт парності не використовується, стоповий біт 1. Для спілкування з

мікроконтролером використовується система двобайтних команд, наприклад, \$BB\$AC, де числа представлені у вигляді шістнадцятиричного коду.

Для передачі даних до ЦАП його слід підключити до порту В. Перший байт даних який слід передавати має бути \$BD. Отримавши цей байт, мікроконтролер переконфігурує порт для роботи з ЦАП, а значення другого байту \$nn записується в 10 розрядний ЦАП. 8 біт другого байту будуть використані як старші біт 10 розрядного ЦАП (два молодші біти не використовуються, тобто рівні нулю).

4. Хід роботи

1) Створіть віртуальні прилади згідно з індивідуальними завданнями. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:

- Лицьову панель розроблених приладів, з відображеною інформацією згідно з індивідуальним завданням;

- Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

1) Яка розрядність ЦАП та у яких режимах він може працювати?

2) Які параметри можливо задавати для сигналу який буде згенеровано?

3) Як визначити розмір масиву та отримати доступ до довільного елемента масиву?

6. Варіанти завдань

1. Розробити прилад який дозволяє генерувати три види сигналів: синусоїдальний, трикутний та пилкоподібний і відобразити усі сигнали на лицьовій панелі приладу усі сигнали. Один зі згенерованих сигналів, за вибором оператора, необхідно «оживити» за допомогою стенда «AVR – мікролаб».

Вибір сигналу який буде «живим» можливо виконати з використанням елементів лицьової панелі, наприклад, Text Ring, Menu Ring або Enum. Перевірити правильність роботи приладу необхідно з використанням осцилографу. Подивіться що станеться якщо ми спробуємо встановити значення амплітуди сигналу який оживлюється більш ніж 2,5 В.

2. Налагодити програму та навести у звіті лицьову панель та діаграму розробленого приладу.

ЛАБОРАТОРНА РОБОТА №12 РОЗРОБКА ОСЦИЛОГРАФА НА БАЗІ СТЕНДА «AVR – мікролаб»

1 Мета роботи: Отримання практичних навиків зі створення комп'ютеризованих засобів вимірювання на базі стенда «AVR – мікролаб».

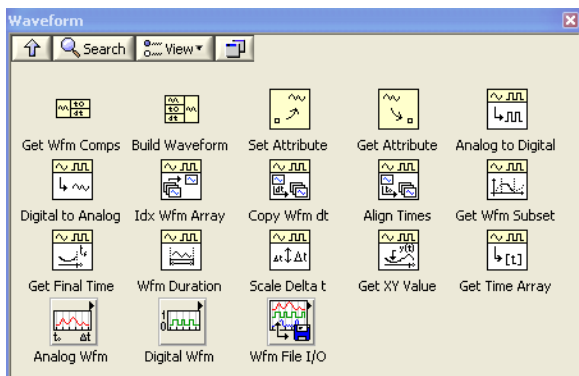
2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6 та програмно-налагоджувальний стенд «AVR – мікролаб».

3 Необхідні довідкові дані по роботі з вбудованим АЦП

3.1 Функції генерації та вимірювання параметрів осцилограм.

В більшості функцій збору даних для їх наступної передачі використовується спеціальний тип даних – осцилограма. Тип даних осцилограма (waveform) являє собою особливий тип кластера і використовується в LabVIEW для роботи з даними які залежать від часу. Осцилограми дозволяють зберегти як значення даних, так і позначку часу отримання первинного значення, значення інтервалу дискретизації та коментарі до даних. Подібно до масивів і кластерів, осцилограми можна складати, віднімати, масштабувати і нормувати.

Вигляд основної палітри функцій осцилограма (Waveform), розташована на

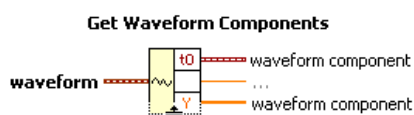


Functions-Programming-Waveform, показаний на малюнку зліва. Сюди також входять Аналогова Осцилограма (Analog Waveform), Цифрова осцилограма (Digital Waveform) та Введення/виведення осцилограм в/з файлу (Waveform File I/O). На вкладці Аналогова осцилограма є підменю: Генерація осцилограм (Waveform Generation) та Вимірювання параметрів осцилограм (Waveform

Measurements)

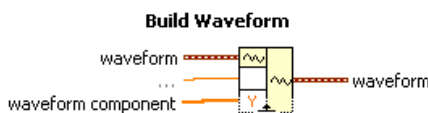
Розглянемо деякі з функцій роботи з осцилограмами.

3.2 Базові функції аналогових та цифрових осцилограм



Отримати компоненти осцилограми (Get Waveforms Components) – функція повертає компоненти осцилограми, визначені користувачем.

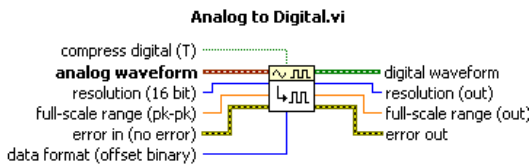
Вхід осцилограма (waveform) є осцилограмою, з якої вилучаються компоненти. Вихід t_0 представляє час (відповідно до системного годинника) отримання першої точки масиву Y . Вихід dt представляє інтервал часу між точками масиву Y . Вихід Y являє значення даних осцилограми. Вихід атрибути (attributes) дозволяє передавати та вводити додаткові дані.



Функція Build Waveform створює осцилограму або модифікує існуючу осцилограму. Якщо вхід осцилограма (waveform) не підключений, функція

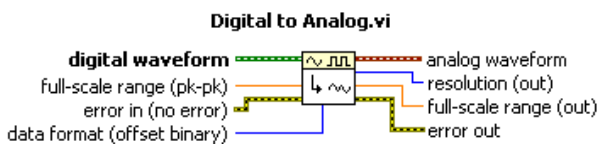
створює нову осцилограму на основі підключених компонентів, а якщо він підключений, то вона модифікується на основі підключених компонентів.

Функція Аналогову Осцилограму на Цифрову (Analog to Digital Waveform)



перетворює аналогову осцилограму на цифрову або цифрові дані. Вхід стискати цифрові дані (compress digital) визначає опцію стиснення цифрової осцилограми чи цифрових даних. Вхід роздільна здатність (resolution) визначає число

бітів, представлених у цифровій осцилограмі або цифрових даних. Вхід повний діапазон (пік-пік) (full-scale range (pk-pk)) задає повний діапазон від піку до піку або різницю між мінімумом і максимумом для цифрової осцилограми або цифрових даних. Вхід Формат даних (data format) визначає необхідне двійкове подання для цифрової осцилограми або цифрових даних.

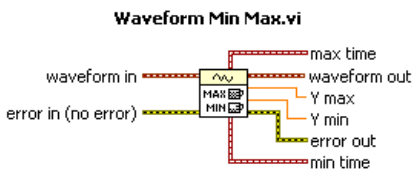


Функція Цифрову Осцилограму на Аналогову (Digital to Analog Waveform)

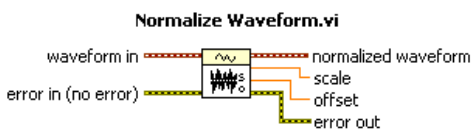
перетворює цифрову осцилограму на аналогову. На вхід цифрова осцилограма

(digital waveform) подається осцилограма, що перетворюється. Вхід повний діапазон перетворення (full-scale range (pk-pk)) визначає різницю між максимальним та мінімальним цифровим значенням. На виході аналогова осцилограма (analog waveform) отримуємо перетворену цифрову осцилограму. Вихід роздільна здатність (resolution) повертає кількість біт представлених у цифровій осцилограмі. Вихід full-scale range (out) повертає повний діапазон цифрової осцилограми.

3.3 Аналогова осцилограма (Analog Waveform) Функція Мінімум та Максимум



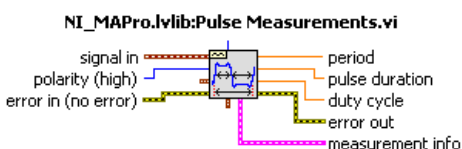
Осцилограми (Waveform Min Max) визначає максимальне та мінімальне значення осцилограми та відповідні значення часу.



Функція Нормалізувати Осцилограму (Normalize Waveform) визначає масштаб (scale) і зміщення (offset), необхідні перетворення вхідної осцилограми до діапазону зміни від -1,0 до 1,0. Осцилограма, наведена до такого діапазону, передається на вихід нормалізована осцилограма.

3.4 Вимірювання параметрів осцилограм

Функція Амплітуда та Рівні (Amplitude and Levels) повертає амплітуду, верхній рівень (high state level) та нижній рівень (low state level) осцилограми або масиву осцилограм. Вихід амплітуда визначається як різниця між верхнім та нижнім рівнями стану. Виходи верхній рівень стану (high state level) та нижній рівень стану (low state level) визначають рівні, на яких осцилограма знаходиться у високому або низькому стані.

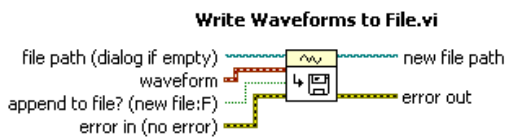


Функція Вимірювання імпульсів (Pulse Measurements) приймає вхідний сигнал окремої осцилограми та повертає період (period), тривалість імпульсу (pulse duration), заповнення циклу (duty cycle)

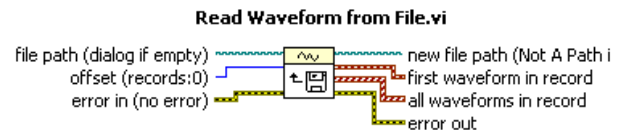
та центр вибраного імпульсу (pulse center). Вхід полярність (polarity) визначає імпульс як високий (high) чи низький (low).

3.5 Введення/виведення осцилограм в/з файлу

Функція Записати Осцилограму у файл (Write Waveforms to File) створює новий файл або додає до існуючого файлу, записує задану кількість записів у файл і закриває файл, перевіряючи наявність помилок.



Функція Зчитування Осцилограми з Файлу (Read Waveform from File) відкриває файл, створений за допомогою Write Waveforms to File, та читає один запис файлу, який може містити одну або кілька осцилограм. Функція повертає на різні виходи усі осцилограми (all waveforms in record) і першу осцилограму в запису (first waveform in record). Вхід зміщення (offset) визначає, наскільки нижче початку файлу буде початок його зчитування, виражене числом байтів.



3.6 Перелік команд, які використовуються при спілкуванні зі стендом «AVR – мікролаб» для доступу до модуля АЦП (циклічний режим)

Обмін даними з мікроконтролером стенду здійснюється за інтерфейсом RS232. Швидкість передачі становить 38400 біт/с, формат переданих даних – 8 біт, біт парності не використовується, стоповий біт 1. Для спілкування з мікроконтролером використовується система двобайтних команд, наприклад, \$BB\$AC, де числа представлені в шістнадцятковому кодуванні.

Модуль аналого-цифрового перетворювача стенда має бути підключений перемичкою до порту А мікроконтролера. Для доступу до вбудованого в мікроконтролер АЦП в циклічному режимі використовується команда виду \$AC\$nn, де nn номер каналу АЦП, що приймає значення від 0 до 7.

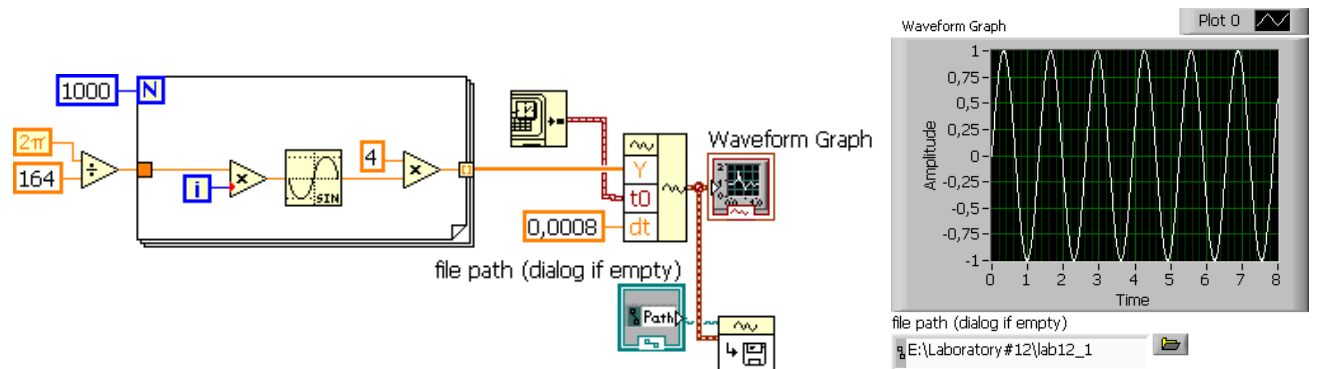
Якщо перший прийнятий мікроконтролером байт дорівнює \$AC, перевіряється другий прийнятий байт. Якщо він не перевищує 7, то провадиться запуск АЦП каналом номер nn і десятибітний результат перетворення повертається у вигляді двобайтного коду виду \$mm\$nn. АЦП переходить в автоматичний режим і постійно здійснює перетворення та передачу даних до того часу, поки знову прийде команда виду \$AC\$nn, що розглядається як завершення циклічного режиму роботи АЦП.

Для перерахунку десяткового коду АЦП значення напруги необхідно визначити крок квантування, який залежить від діапазону перетворення, в нашому випадку (0-5) В і розрядності АЦП (10 розрядів).

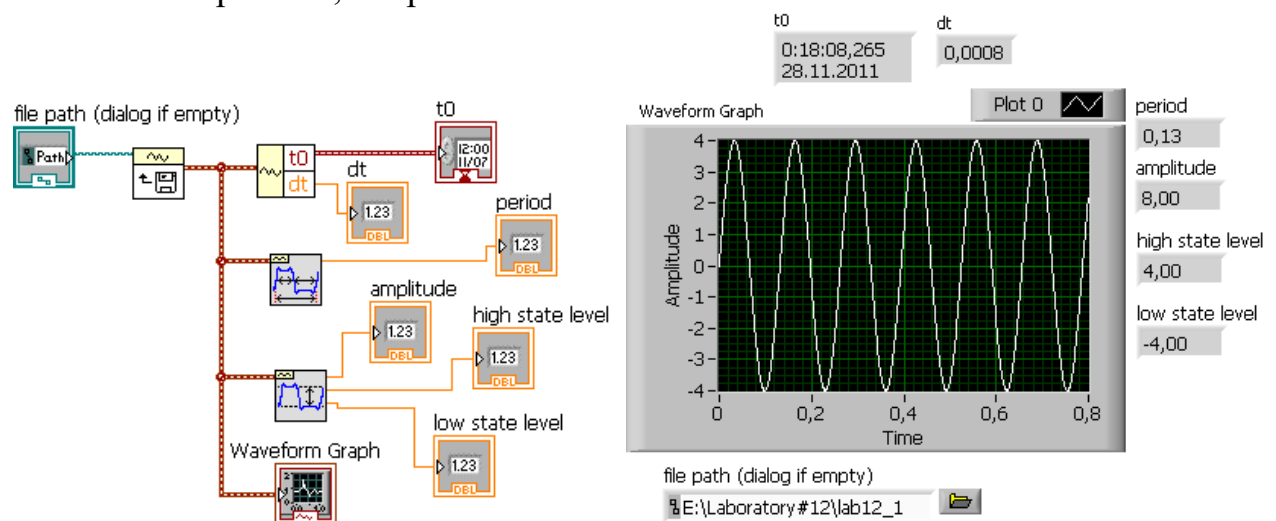
3.7 Поосцилограмимо ...?!!!

Створіть новий віртуальний пристрій, який сформує осцилограму. Для цього нам знадобиться функція косинуса, яку ми помістимо в цикл з кількістю 1000 ітерацій і скористаємося функцією Build Waveform. Початковий час організуємо з допомогою функції Get Date/Time in Seconds. Крок дискретизації встановить з

діапазону 0,0005-0,0025. Частоту сигналу, що формується, визначимо за допомогою дільного значення від числа π , а амплітуду визначимо за допомогою множника до результату розрахунку значення синуса. Результат генерації осцилограми відобразимо на індикаторі Waveform Graph і запишемо у файл за допомогою функції Write Waveforms to File. В результаті отримуємо приблизно таку картину лицьової панелі та блок діаграми.



Створіть ще один віртуальний прилад, який прочитає записаний файл, відобразить осцилограму на індикаторі, визначить її характеристики (момент отримання першого відліку та крок дискретизації), а також виміряти параметри сигналу: період сигналу, мінімальний і максимальний рівні сигналу, повний розмах сигналу. Для цього скористаємося функціями Read Waveform from File, Get Waveforms Components, Amplitude and Levels та Pulse Measurements.



Додайте в пристрій можливість визначати частоту сигналу.

4. Хід роботи

1) Створіть віртуальні прилади згідно з індивідуальними завданнями. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:
 - Лицьову панель розроблених приладів, з відображуваною інформацією згідно з індивідуальним завданням;

- Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

1) За допомогою яких функцій створюється осцилограма та визначаються її компоненти?

2) Чим відрізняється циклічний режим роботи АЦП від використаного у лабораторній роботі №10, у чому вираш?

3) Які параметри осцилограм можуть бути виміряні відомими вам функціями?

6. Варіанти завдань

Завдання

1. Розробити програму, яка дозволяє отримати задану викладачем кількість точок (від 500 до 10 тисяч) сигналу від джерела, підключеного до стенду «AVR – мікролаб», АЦП стенду (2 канал) використовувати в циклічному режимі. Значення кодів, отриманих від АЦП, перетворити в напругу в діапазоні (0-5) В. З отриманих точок сформувати осцилограму з кроком дискретизації 530 мкс. Початковий момент осцилограми визначити за допомогою годинника комп'ютера. Отриману осцилограму записати у файл.

2. Розробити програму для аналізу характеристик отриманої осцилограми: повного розмаху, мінімального та максимального рівнів, періоду та частоти сигналу, моменту отримання першого відліку та кроку дискретизації. Відобразити осцилограму що аналізується на індикаторі. Виконайте нормалізацію осцилограми, визначте масштаб та зсув, необхідні для нормалізації осцилограми. Відобразіть нормалізовану осцилограму на окремому індикаторі.

3. Налаштувати програму та навести у звіті лицьову панель та діаграму розробленого приладу.

ЛАБОРАТОРНА РОБОТА №13 ПРОЄКТИ В LABVIEW

1 Мета роботи: Отримання практичних навиків зі створення проєктів в середовищі LabVIEW та опанування віртуальними приладами обробки інформації від пристроїв вводу інформації (клавіатура, мишка).

2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6 та програмно-налагоджувальний стенд «AVR – мікролаб».

3 Необхідні довідкові дані по роботі з проєктами в LabVIEW

3.1 Проєкти в LabVIEW

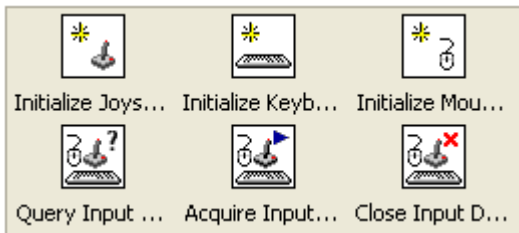
Однією з потужних особливостей LabVIEW є можливість створювати файли які можуть виконуватися на інших комп'ютерах. Для цього необхідно, щоб віртуальний прилад, що розробляється, був створений в рамках проєкту. Для цього ви повинні в меню Файл середовища LabVIEW вибрати новий проєкт (New Project) або зайти до пункту Project та вибрати New Project. При цьому викликається провідник проєкту, у якому можна виконувати різні зміни у проєкті.

Додайте в проєкт новий віртуальний прилад і розмістіть на його лицьовій панелі контроль та індикатор у вигляді градусника, з'єднайте їхні іконки між собою на діаграмі. Збережіть проєкт на жорсткому диску. У провіднику проєкту виберіть Build Specification, щоб вибрати налаштування виконуваного файлу. Натиснувши праву клавішу миші, виберіть New-Application (.exe). У полі Target Filename вкажіть бажане ім'я файлу, а в полі Destination Directory вкажіть папку, де хочете записати створений файл. На вкладці Source Files вкажіть розроблений вами віртуальний прилад як стартовий (Startup VIs) і натисніть кнопку Build (створити файл, що виконується).

Знайдіть створений файл з розширенням .exe і запустіть його на виконання. Переконайтеся, що програма працює. Створений файл буде працювати без відкриття середовища програмування LabVIEW, але йому недоступні функції редагування блок діаграми. Якщо ви хочете внести зміни в роботу даної програми ви повинні змінити код файлу в проєкті і заново створити файл, що виконується. Створений таким чином файл працюватиме лише на комп'ютерах, де встановлене середовище LabVIEW.

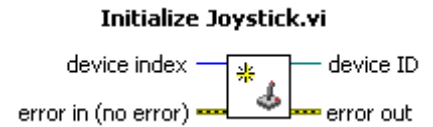
Для того щоб мати можливість запускати таку програму на комп'ютерах без середовища LabVIEW, необхідно створювати спеціальний інсталяційний пакет (installer), який встановить всі необхідні для роботи програми елементи. Це здійснюється шляхом вибору в меню Build Specification пункту New-Installer. При цьому повинні бути вказані елементи, необхідні для встановлення на іншій машині, вимоги до операційної системи, обмеження на розміри файлів та ряд інших відомостей. Результатом роботи є набір файлів, що дозволяє встановити програму на довільну машину і виконати її.

3.2 Функції керування пристроями вводу

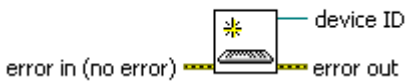


Функції керування пристроями введення розташовані Functions-Connectivity-Input Device Control. Усього на підпалітрі можна побачити шість функцій.

Функція Initialize Joystick ініціалізації джойстика відкриває послання та ініціалізує джойстик за заданим індексом пристрою. За умовчанням на цьому вході встановлено значення 0, що відповідає одному джойстику.



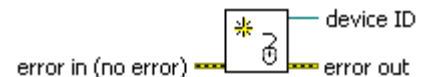
[Initialize Keyboard.vi]



Функція Initialize Keyboard відкриває послання та ініціалізує клавіатуру, підключену до комп'ютера.

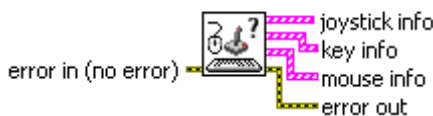
Функція Initialize Mouse відкриває послання та ініціалізує мишу, підключену до комп'ютера.

Initialize Mouse.vi



Функція Query Input Device - запитувати пристрої

Query Input Devices.vi



введення, отримує інформацію про пристрої підключені до комп'ютера. Кластер даних джойстика (joystick info) містить інформацію по джойстику, підключеному до комп'ютера: загальна кількість осей (axes total), загальна кількість кнопок (buttons total),

загальна кількість точок огляду (pov total) та ім'я пристрою (device name). Кластер даних клавіатури (key info) містить інформацію про загальну кількість клавіш на клавіатурі (buttons total) та найменування пристрою (device name). Кластер дані миші (mouse info) містить загальну кількість осей (axes total), загальну кількість кнопок (buttons total) та найменування пристрою (device name).

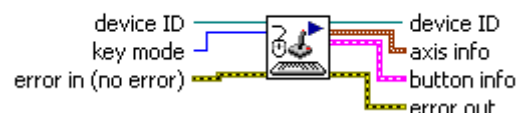
Функція отримання даних введення повертає дані про пристрій, підключений до комп'ютера, тип даних залежить від підключеного до входу ідентифікатора пристрою.

На лівому малюнку показаний вихід функції, якщо на вхід підключено ідентифікатор клавіатури, а на правому малюнку показано вигляд функції, якщо підключений ідентифікатор миші.

Acquire Input Data.vi



Acquire Input Data.vi

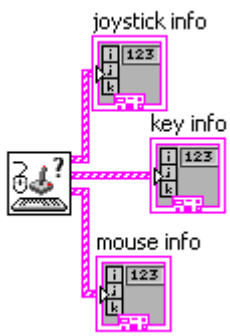


Close Input Device.vi



Функція Close Input Device закрить устройство ввода закрывает устройство ввода, определенное ссылкой на идентификатор устройства

Створіть новий віртуальний прилад і помістіть на блок діаграму функцію Query



Input Device та створіть три індикатори підключені до його трьох виходів. Запустивши прилад на виконання, на лицьовій панелі ви зможете побачити інформацію про пристрої введення підключених до комп'ютера. Додайте на блок діаграму функції ініціалізації миші та клавіатури та підключіть до них функції отримання інформації про пристрій, поміщені в цикл. Створіть відповідні індикатори та перегляньте, що відбувається при натисканні на клавіші клавіатури або кнопки миші, до чого призводить переміщення вказівника миші. Не забудьте закрити пристрої введення після закінчення

виконання програми.

4. Хід роботи

1) Створіть віртуальні прилади згідно з індивідуальними завданнями. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:

- Лицьову панель розроблених приладів, з відображуваною інформацією згідно з індивідуальним завданням;

- Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

1) Як можна створити файл (name.exe) який буде виконуватись в LabVIEW?

2) Що необхідно виконати аби розроблений файл (name.exe) виконувався на інших комп'ютерах?

3) Як засобами LabVIEW визначити факт натиснення будь якої клавіші миші або яка клавіша була натиснута на клавіатурі?

6. Варіанти завдань

1. Розробіть віртуальний прилад, що дозволяє змінювати показання індикатора типу градусник шляхом натискання на клавішу миші (ліва клавіша – збільшення показань на 1, права клавіша зменшення показань на 1), за умови, що курсор миші знаходиться в прямокутнику з координатами 500-550 по горизонталі 150 по вертикалі або натискання клавіш UP і DOWN на клавіатурі. Створіть файл, що виконується, і перевірте працездатність програми.

2. Перевірте працездатність програми та наведіть в звіті лицьову панель та діаграму.

ЛАБОРАТОРНА РОБОТА №14

ВВОД-ВЫВІД АНАЛОГОВОЇ ІНФОРМАЦІЇ З ДОПОМОГОЮ DAQ USB 6008

1 Мета роботи: Отримання практичних навиків використання функцій збору даних та функцій формування аналогових та цифрових сигналів за допомогою модуля USB 6008.

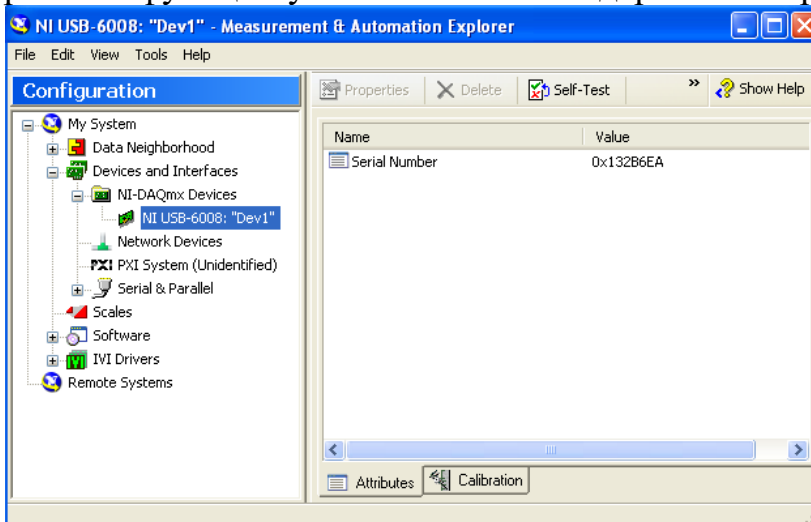
2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6 та модуль вводу - виводу інформації USB 6008.

3 Необхідні довідкові дані по роботі з модулем USB 6008

3.1 Функції підтримки вводу/виводу даних та стандартних інтерфейсів.

Підтримка роботи плат вводу/виводу даних та стандартних інтерфейсів забезпечується функціями що входять категорії Контроль вводу/виводу (Measurement I/O), Зв'язок з приладами (Instrument I/O) та Зір і рух (Vision&Motion).

Початковим етапом роботи з вбудованими платами та стандартними інтерфейсами вводу/виводу даних після встановлення плат та драйверів є їх конфігурування та тестування за допомогою програми Провідник вимірювань та автоматизації (Measurements and Automation Explorer (MAX)). MAX є програмним інтерфейсом Windows, що забезпечує доступ до всіх плат National Instruments. MAX встановлюється за умовчанням під час встановлення LabVIEW. При запуску MAX відкривається діалогове вікно, яке дозволяє вибирати та встановлювати різні режими функціонування плат та стандартних інтерфейсів.



У лівому вікні Конфігурація (Configuration) відображаються розділи конфігурації локальної та віддаленої систем. Перелік розділів включає Оточення даних (Data Neighborhood), Пристрої та інтерфейси (Devices and Interfaces), Шкали (Scales), Програмне забезпечення (Software) та Драйвери IVI (IVI Drivers).

У розділі оточення даних відображаються всі налаштовані завдання NI-DAQmx та глобальні канали NI-DAQmx. Завдання – це найважливіша концепція NI-DAQmx, яка представляє сукупність одного або кількох віртуальних каналів із синхронізацією, запуском та іншими властивостями. Завдання визначає вимірювання або генерацію сигналів, які необхідно виконати. Віртуальні канали, що

входять до складу завдання, включають сукупність налаштувань фізичного каналу DAQ, тип вимірювання та інформацію про нормування значень. У NI-DAQmx можна конфігурувати віртуальні канали як частину завдання або окремо від завдання. Віртуальні канали, створені всередині завдання, є локальними. Віртуальні канали, створені поза завданням, є глобальними та можуть використовуватися окремо. Глобальні канали можуть бути створені в MAX або LabVIEW. Завдання також можуть створюватися як у MAX, так і в LabVIEW.

Розділ Пристрої та інтерфейси вікна Конфігурації дозволяє переглянути параметри встановлених у системі пристроїв та виконати ряд дій щодо їх тестування та калібрування.

Розділ Шкали дозволяє створити, переглянути та за необхідності налаштувати шкалу, що визначає характер перетворення вимірюваної величини.

Розділ Програмне забезпечення дозволяє переглянути опис, розташування та номер версії програм NI.

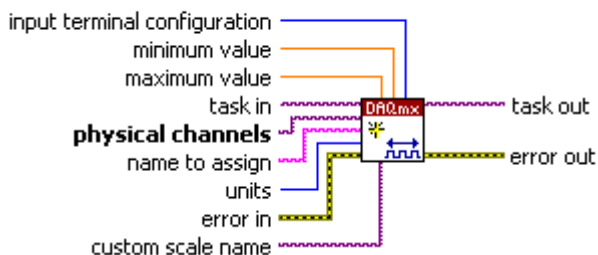
3.2 Функції збору даних DAQmx

Віртуальні прилади і функції збору даних NI-DAQmx відносяться до нового покоління драйверів NI-DAQ. Основним елементом NI-DAQmx є завдання, яке може бути створене як у програмі MAX, так і в LabVIEW. У LabVIEW завдання може бути створене декількома способами:

- Шляхом розміщення елемента управління Ім'я завдання DAQmx (DAQmx Task Name) на лицьовій панелі або константи з таким же ім'ям на блок-діаграмі та вибору пункту Нова задача (New Task) з контекстного меню елемента або константи;
- Шляхом розміщення Експрес ВП Помічник DAQ (DAQ Assistant) на блок-діаграмі;
- Шляхом вибору рядка ВП із шаблону (VI from Template) у вікні початкового запуску та наступного вибору ВП Збір даних за допомогою NI-DAQmx (Data Acquisition with NI-DAQmx).

Створити віртуальний канал DAQmx (DAQmx Create Virtual Channel) –

DAQmx Create Channel (AI-Voltage-Basic).vi

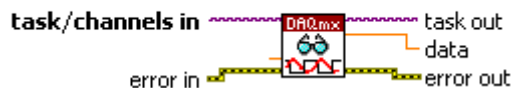


віртуальний прилад створює віртуальний канал (Virtual Channel) або набір віртуальних каналів та додає їх до завдання (Task). При конфігуруванні цього поліморфного ВП за допомогою селектора вибирається тип каналу введення або виведення: аналоговий, цифровий або лічильний. Залежно від виду

вимірювання або генерації сигналу може вибиратися вимірювання температури, генерація напруги або підрахунок подій. У деяких випадках вибирається також тип датчика, такий, наприклад, як термопара або терморезистор при вимірюваннях температури.

Читати DAQmx (DAQmx Read) віртуальний прилад зчитує із завдання чи каналів, визначених користувачем. Реалізація цього поліморфного ВП передбачає вибір формату вибірок, що повертаються, вибір зчитування одиничної вибірки або сукупності вибірок, а також

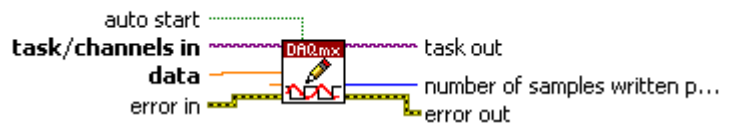
DAQmx Read (Analog DBL 1Chan 15amp).vi



вибір зчитування з одного або декількох каналів.

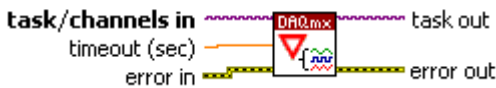
Записати в DAQmx (DAQmx Write) віртуальний прилад записує вибірки задач або каналів, визначених користувачем. Реалізація цього поліморфного ВП передбачає вибір формату вибірок, що записуються, вибір запису одиничної вибірки або сукупності вибірок, а також вибір запису в один або кілька каналів.

DAQmx Write (Analog DBL 1Chan 15amp).vi



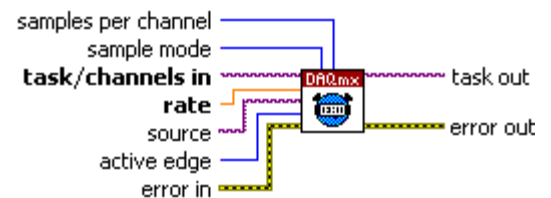
Очікувати на виконання DAQmx (DAQmx Wait Until Done) віртуальний прилад очікує завершення вимірювання або генерації. Цей ВП необхідно використовувати для того, щоб гарантувати завершення заданої операції перед зупинкою завдання.

DAQmx Wait Until Done.vi

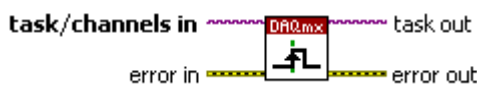


Синхронізація DAQmx (DAQmx Timing) віртуальний прилад конфігурує кількість зібраних або генерованих вибірок і створює буфер. Реалізація цього поліморфного ВП відповідає типу синхронізації, що використовується в задачі.

DAQmx Timing (Sample Clock).vi



DAQmx Start Trigger (None).vi



Запуск DAQmx (DAQmx Trigger) віртуальний пристрій конфігурує запуск завдання. Реалізація цього поліморфного ВП відповідає наявності та

типу запуску задачі.

Почати завдання DAQmx (DAQmx Start Task) віртуальний прилад переводить завдання на стан виконання для початку вимірювання або генерації.

DAQmx Start Task.vi



Використання цього ВП необхідне деяких програм і є додатковим іншим. Якщо цей ОП не використовується, то вимірювальна задача починається автоматично при виконанні ОП Читати DAQmx Read.

DAQmx Stop Task.vi



Зупинити завдання DAQmx (DAQmx Stop Task) ВП зупиняє завдання та повертає її до стану, в якому завдання перебувало перед використанням ВП Почати завдання DAQmx або перед використанням

ВП Записати в DAQmx із встановленим у стан Істина входом автоматичний початок.

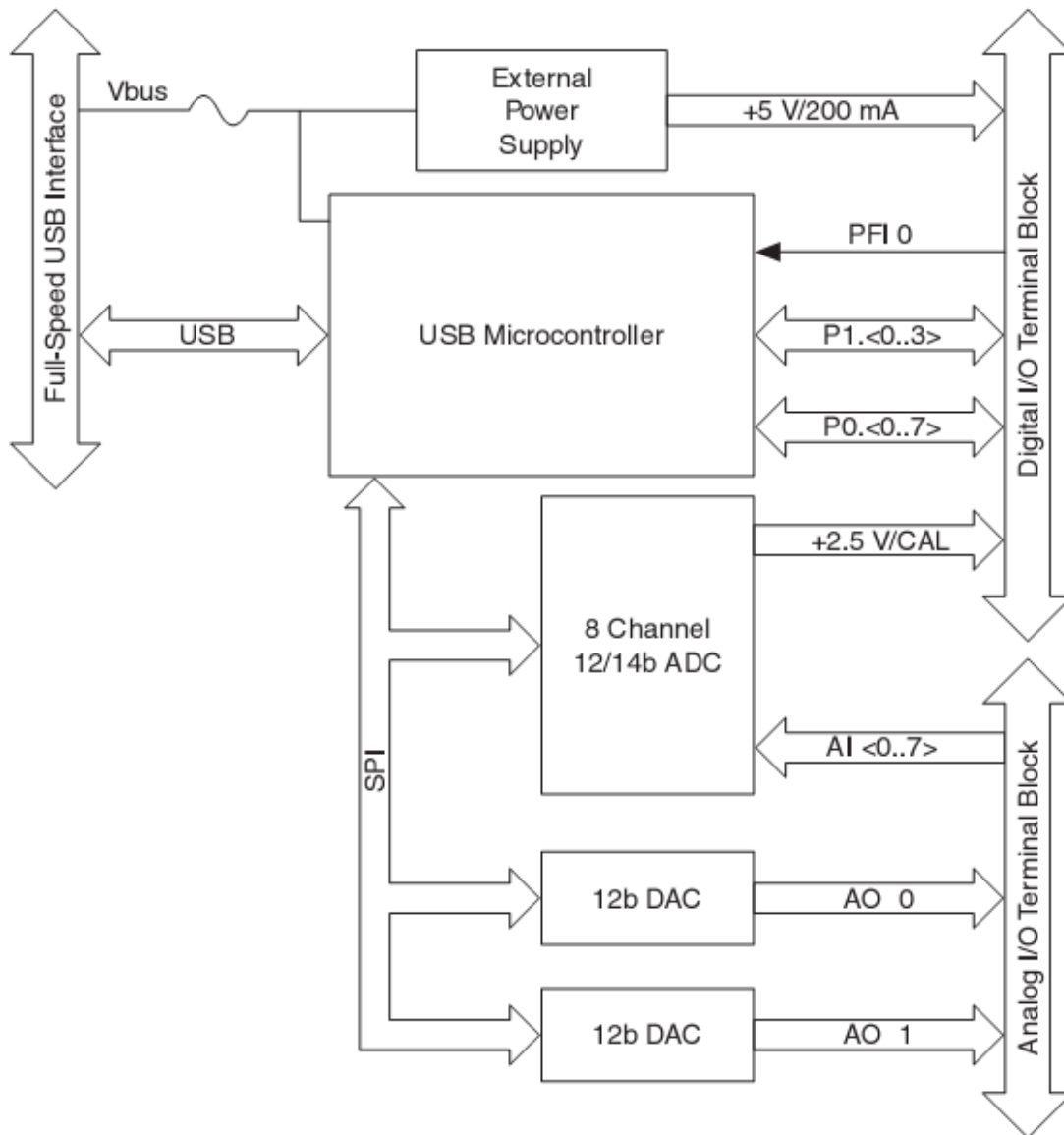
Очистити завдання DAQmx (DAQmx Clear Task) ВП очищує завдання. Перед очищенням цей ЗП зупиняє завдання та звільняє всі ресурси, відведені задачі. Після очищення завдання не може використовуватися, незважаючи на його створення.

DAQmx Clear Task.vi



3.3 Характеристики модуля введення/виводу USB 6008

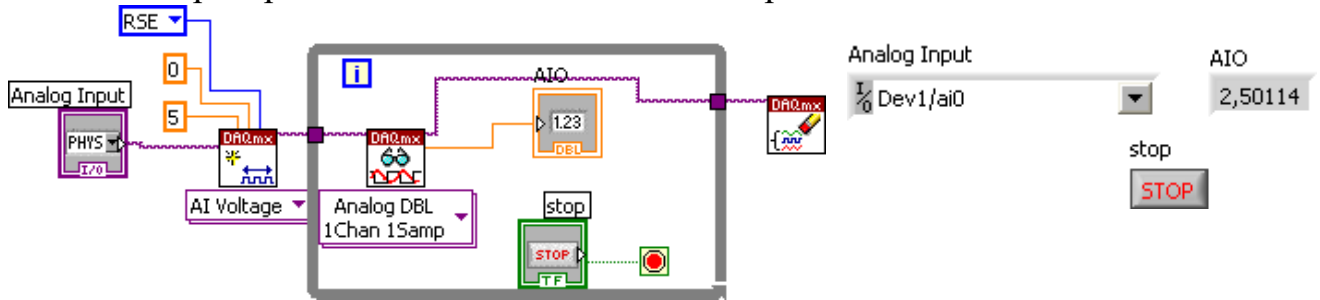
Модуль введення/виведення USB 6008 один з найпростіших з пропонованих фірмою National Instruments. Він пропонує 12 каналів введення/виведення цифрової інформації, 8 каналів введення аналогової інформації (12 бітний АЦП), два канали виведення аналогової інформації (два 12 бітних ЦАП) та один канал для підрахунку зовнішніх подій (32 бітний лічильник). На малюнку представлено спрощену схему модуля.



3.4 Використання функцій збору даних DAQmx для доступу до USB 6008

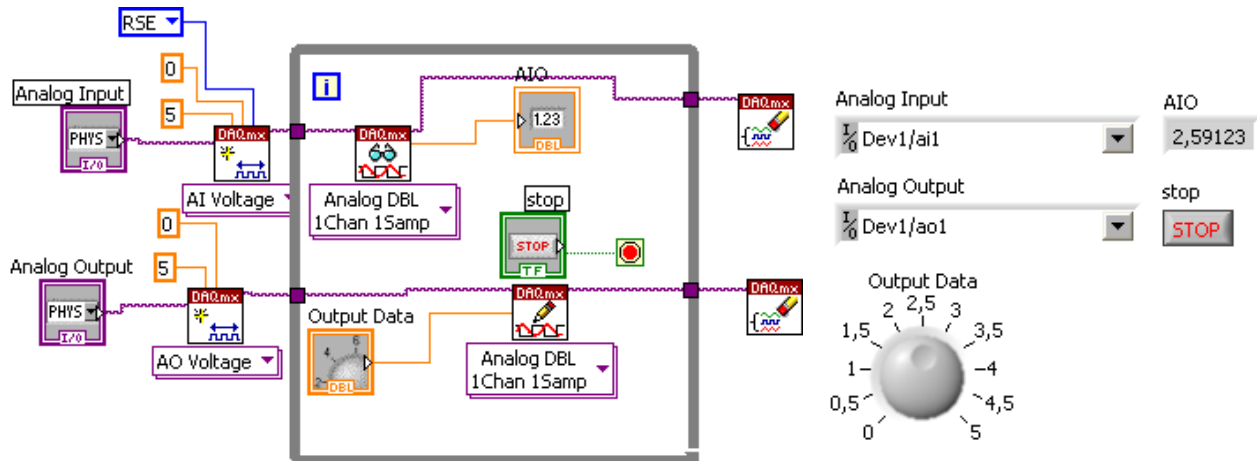
На першому етапі нашим завданням буде виконання вимірювань нульовим каналом АЦП (AI0, контакт №2 модуля USB 6008) підключеного до джерела постійної напруги 2,5В (контакт №30 модуля USB 6008). Для цього створимо віртуальний прилад, який створює і конфігурує віртуальний канал введення аналогової інформації для вимірювань в діапазоні від 0 до 5 В, в циклі розмістимо команду читання і відобразимо результат вимірювання на індикаторі. Після закінчення роботи завдання очищується. В якості робочого каналу вибирається

канал АІО пристрою підключеного до комп'ютера.

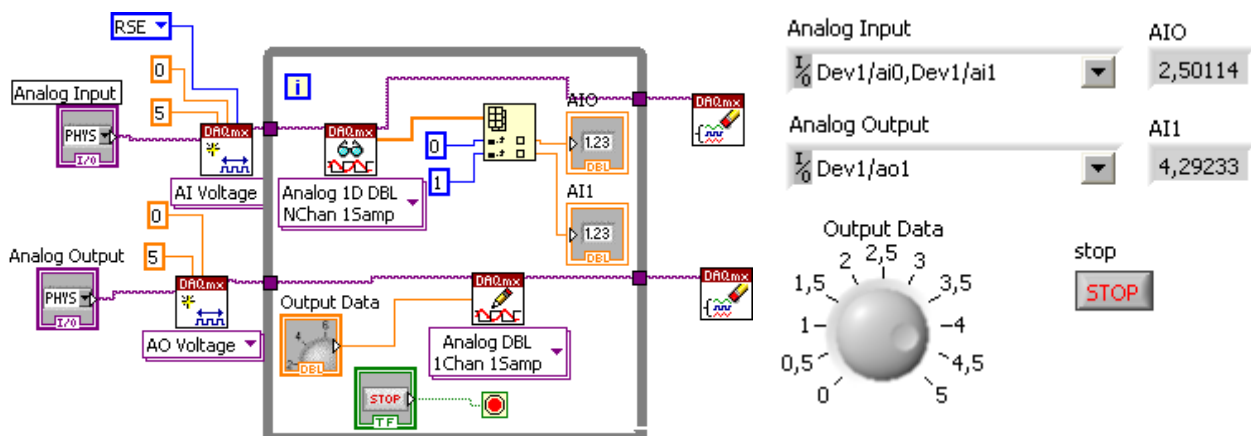


Запустіть програму та переконайтеся у відповідності показань вимірюваному значенню – 5В.

Далі додамо до приладу елементи управління ЦАП і перевіримо працездатність схеми шляхом одночасного вимірювання вихідного сигналу першого каналу ЦАП (АО1), підключеного до першого каналу АЦП (АІ1), для цього змініть нульовий канал АЦП на перший. Переконайтеся, що сигнал, який подається на вхід ЦАП, і який знімається з виходу першого каналу АЦП Далі додамо до приладу елементи управління ЦАП і перевіримо працездатність схеми шляхом одночасного вимірювання вихідного сигналу першого каналу ЦАП (АО1), підключеного до першого каналу АЦП (АІ1), для цього змініть нульовий канал АЦП на перший. Переконайтеся, що сигнал, який подається на вхід ЦАП, і який знімається з виходу першого каналу АЦП однакові.

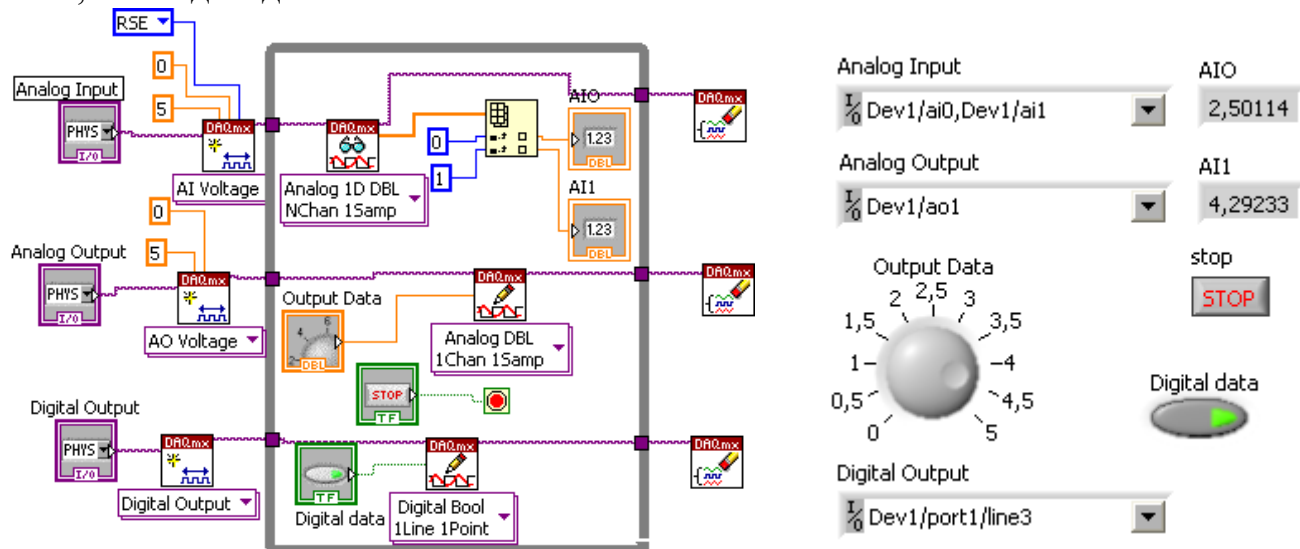


Для одночасного вимірювання двома каналами АЦП модифікуємо програму в такий спосіб.



Зверніть увагу, що для забезпечення одночасного вимірювання по кількох каналах необхідно змінити режим ВП DAQmx Read, а в найменуванні фізичного каналу вказати через кому або через двокрапку якими каналами будуть проводитися вимірювання.

Для перевірки роботи з цифровими каналами додамо засоби керування світлодіодом, підключеним через токообмежуючі резистори, своїм анодом до порту P1.3, а катодом до землі.



Натискаючи на контроль стану лінії, переконайтеся, що червоний світлодіод, підключений до модуля, також змінює стан.

4. Хід роботи

1) Створіть віртуальні прилади згідно з індивідуальними завданнями. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

- 2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:
- Лицьову панель розроблених приладів, з відображуваною інформацією згідно з індивідуальним завданням;
 - Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

- 1) Як може бути створено завдання вимірювання або генерації сигналів NI-DAQmx?
- 2) Як створити віртуальний канал і визначити його тип?
- 3) Яким чином визначають канали, які використовуються в режимі багатоканальної роботи віртуального каналу і як отримати доступ до результатів вимірювань?

6. Варіанти завдань

1. Змініть програму 11 лабораторної роботи, для генерації різних видів сигналів, використовуючи нульовий канал ЦАП (AO0) модулю USB 6008 та перевірте результат за допомогою осцилографу.

2. Налаштуйте програму і наведіть в звіті лицьову панель і діаграму.

ЛАБОРАТОРНА РОБОТА №15

РОЗРОБКА ЕЛЕМЕНТІВ КОРИСТУВАЧА ПРИЗНАЧЕНИХ ДЛЯ КЕРУВАННЯ ТА ІНДИКАЦІЇ В LABVIEW

1 Мета роботи: Отримання практичних навиків з розробки зовнішнього вигляду елементів користувача та ініціювання змінення поведінки елементів призначених для індикації та контролю в середовищі LabVIEW що допоможе зробити більш інформативними зовнішній вигляд панелей керування віртуальних приладів які знаходяться в розробці.

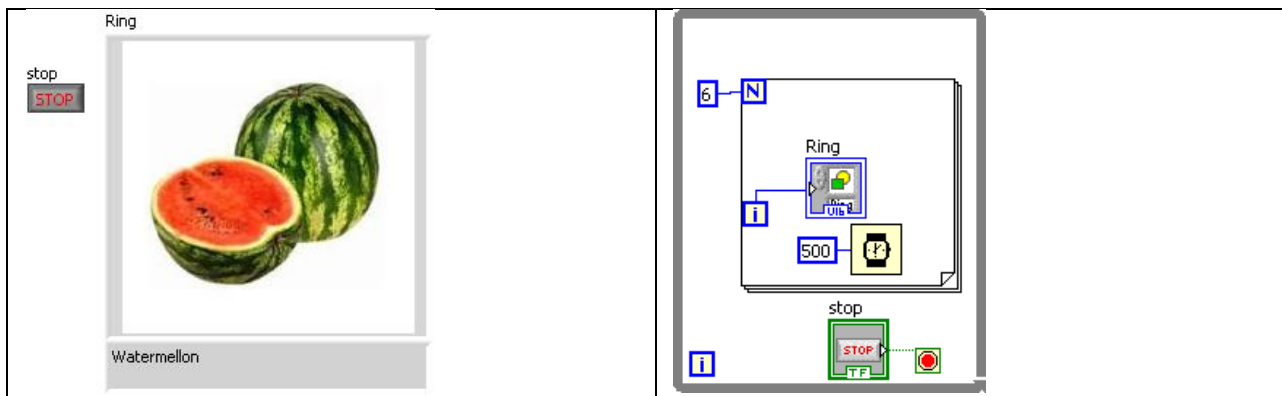
2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6 та та найпростіші графічні редактори.

3 Необхідні довідкові дані

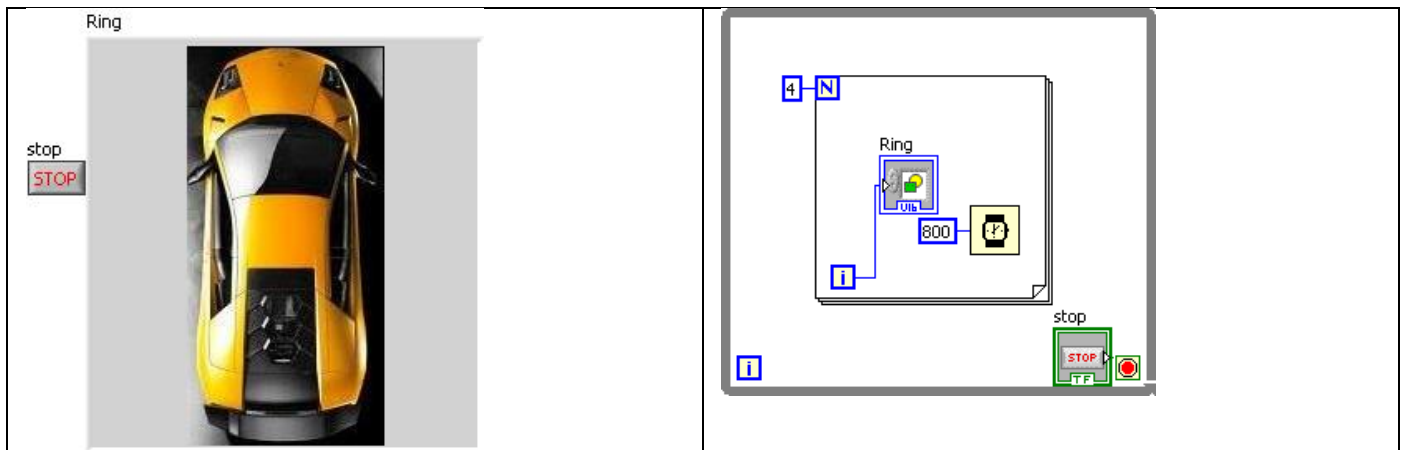
3.1 Використання засобів LabView для розробки більш «живої» лицьової панелі віртуальних приладів, що розробляються.

При необхідності відображення графічної інформації, що супроводжує або пояснює виконання програми, можна скористатися вбудованими в LabView елементами Pict Ring і Text&PictRing, аналогом цих елементів Text Ring ми користувалися в 11 лабораторній роботі. Основною відмінністю елементів Pict Ring та Text&PictRing є те, що вони можуть відображати графічний матеріал (картинки) заздалегідь завантажені користувачем в ці елементи. Text&PictRing може одночасно відображати картинку та текстову інформацію в різних віконцях. Для розміщення картинки на одній із закладок елементів Pict Ring та Text&PictRing необхідно попередньо завантажити її в буфер пам'яті (розділ меню Edit→Import picture to clipboard), а потім використати контекстне меню елементів "Import picture from clipboard" для вставки картинки. В елементі Text&PictRing для кожної картини можна зробити свій підпис, який можна додати у розділі властивостей елемента "Edit items". Спочатку елементи Pict Ring і Text&PictRing є елементами контролю та вибір відповідної картини (кожній з яких відповідає деяке ціле число від 0 до n-1, де n кількість картинок поміщених в елемент) здійснює оператор за допомогою покажчика мишки. Змінивши властивості елементів з контроль на індикатор, можна програмними засобами керувати зміною картинок відтворюваних в елементах Pict Ring і Text&PictRing шляхом зміни значення числа поданого на вхід даного елемента.

Як приклад скористаємось елементом Text&PictRing для відображення шести різних картинок із назвами об'єктів зображених на цих картинках. Для цього скористаємось заздалегідь підготовленими картинками. Послідовно додайте до елемента Text&PictRing усі шість картинок і підпишіть назви об'єктів зображених на них. Помістивши розроблений елемент у структуру циклу та задавши часову затримку, можна спостерігати послідовну зміну картинок на лицьовій панелі. Розроблений прилад може мати вигляд подібний до представленого на малюнках.



Для створення ефекту зміни положення або стану можна використовувати ту саму картинку, представлену в різних ракурсах. Як приклад можна використовувати картинку жовтого автомобіля з різною орієнтацією (напрямок ліворуч, праворуч, вгору та вниз). Додавши в елемент Pict Ring всі чотири запропоновані зображення автомобіля і зробивши програму подібну до попередньої можна створити ілюзію обертання автомобіля за або проти годинникової стрілки.



3.2 Створення власних елементів керування в LabView

Ви можете переробляти елементи керування та відображення за своїм бажанням з метою кращої відповідності програмі та більш вражаючого графічного інтерфейсу. Можна зберігати створений елемент керування або відображення в директорії або бібліотеці ВП, як це робилося з віртуальними приладами. Загальноприйняте використовувати розширення .ctl при найменуванні файлів елементів керування. Ви навіть можете створити стандарт елемента керування, якщо хочете задіяти його в багатьох місцях одного і того ж ВП, зберігши цей елемент як визначник типу. При внесенні змін до визначника типу LabVIEW автоматично оновлює всі ВП, в яких він присутній.

Як створити свій власний елемент керування?

Зазвичай при здійсненні цієї операції потрібно вставити малюнки в створювані елементи керування. Тому спочатку слід знайти (створити) файли малюнків і відповідну графічну програму. У LabVIEW відсутній редактор малюнків. При створенні елемента керування використовується форма існуючого елемента, такого, наприклад, як логічний світлодіод або повзунковий числовий елемент керування.


Для прикладу розглянемо створення елемента керування, який змінює колір автомобіля на картинці, скориставшись двома картинками автомобіля з різним кольором.

Помістіть логічний елемент керування у вигляді світлодіода на передню панель і виділіть його.

Запустіть редактор, вибравши Customize Control із меню Edit.

Вікно редактора елемента керування покаже логічний елемент.

У режимі редагування вікно редактора працює як передня панель. Ви можете викликати контекстне меню елемента керування для зміни його характеристик (масштаб, точність тощо)

У режимі налаштування (Customize mode), в який ви перейдете клацанням миші по кнопці із зображенням інструмента , можна змінити розмір, колір та змінити різні компоненти малюнка елемента управління. Замінити стандартні малюнки світлодіода своїми можна кількома способами: імпортувавши малюнок з буфера - "Import picture from clipboard", імпортувавши малюнок з буфера зі збереженням поточного розміру картини - "Import from clipboard at same size" чи імпортувавши картинку безпосередньо з файлу - "from file" або імпортувавши картинку безпосередньо з файлу зі збереженням поточного розміру картини – "Import from file at same size".

У режимі налаштування викличте контекстне меню логічного елемента керування та зручним способом виберіть функцію імпортування малюнка.

Поверніться в режим редагування елемента контролю (Change to edit mode), змініть стан світлодіода з "Брехня" на "Істина" і знову перейшовши в режим налаштування (Change to customize mode) повторіть попередні операції для варіанта істина логічного елемента, використовуючи інший малюнок машинки.

Збережіть створений елемент керування, вибравши функцію "Зберегти як..." з меню Файл. Ви повинні отримати файл з розширенням .ctl. У разі запиту на заміну оригінального елемента знову розробленим, виберіть варіант не замінювати.

Створіть на базі світлодіодного елемента керування два різні елементи контролю: у першому елементі при надходженні сигналу керування, повинен змінитися колір автомобіля, а у другого елемента повинна змінюватися орієнтація автомобіля.

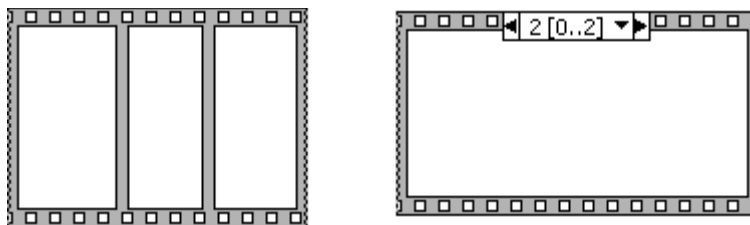
Перевірте працездатність створених елементів контролю, використовуючи їх у програмі як контроль та як індикатор.

При створенні власного елемента керування з використанням числового повзункового елемента керування алгоритм дій аналогічний, з тією різницею, що замінюється тільки один елемент - покажчик числового повзункового елемента керування зображенням, що зберігається у файлі, або заздалегідь скопійованим у буфер пам'яті.

Створіть чотири числових повзункових елементи керування з покажчиками у вигляді автомобілів різного кольору, спрямованих в одну сторону. Перевірте працездатність створених елементів контролю, використовуючи їх у програмі як контроль і як індикатор.

3.3 Забезпечення заданої послідовності виконання операцій у LabVIEW

Визначення порядку виконання програми шляхом організації її елементів певну послідовність називається управлінням потоком даних. У звичайних мовах програмування, наприклад, Basic або C, завжди є управління потоком даних, оскільки оператори виконуються в тій послідовності, в якій вони записані в програмі. Для управління потоком при обробці даних у LabVIEW використовуються структури послідовності (Flat Sequence Structure або Stacked Sequence Structure).



При необхідності до структури додається потрібна кількість кадрів, а також є можливість вставляти/видаляти кадри в довільному місці послідовності. У структурі Flat Sequence відразу видно код програми у всіх вікнах, що поміщаються на екрані монітора комп'ютера, а в структурі Stacked Sequence, у кожний момент часу, доступний для перегляду лише один кадр, що має свій номер, починаючи з цифри "0", для перегляду наступних вікон потрібно прогортати вікна до необхідного номера, така структура більш компактна і займає менше місця на екрані монітора.

При використанні структур послідовності, що складаються з декількох послідовно розташованих вікон, програма виконується кадр за кадром, що забезпечує виконання тієї чи іншої частини програми до або після виконання іншої частини програми. Код розташований у вікнах послідовної структури виконується зліва направо, код починає виконання лише за умови, що всі дані підключені до вікна готові (доступні), а виходять дані з вікна тільки за умови, що закінчилося виконання програми в даному вікні.

4. Хід роботи

1) Створіть елементи керування описані у третьому розділі. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:

- Лицьову панель розроблених приладів, з відображуваною інформацією згідно з індивідуальним завданням;
- Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

1) Поясніть призначення елементів контролю Pict Ring та Text&PictRing та особливості їх застосування?

2) Яким чином можна створити власні елементи керування в LabVIEW?

3) Як забезпечується керування потоком даних у LabVIEW?

6. Варіанти завдань

Завдання

1. Використовуючи елементи контролю, створені в пункті 3.2, розробіть програму:

А) непарний номер за списком у журналі

Перегони на автомобілях: У гонках беруть участь чотири автомобілі, що мають різні кольори кузова. Гонка починається натисканням елемента управління у вигляді авто, що змінює свій колір (з червоного на зелений). Всі автомобілі повинні пройти трасу довжиною 100 км. Для кожного автомобіля випадково задається швидкість руху по трасі в діапазоні від 80 до 100 км/год, гонка закінчується в момент досягнення фінішу першим автомобілем. Час, показаний переможцем, необхідно відобразити на цифровому індикаторі.

Б) парний номер за списком у журналі

Естафетна гонка на автомобілях. У гонках беруть участь чотири автомобілі, що мають різні кольори кузова. Гонка починається натисканням елемента управління який виглядає як авто яке змінює свою орієнтацію в просторі. Для кожного автомобіля випадково задається швидкість руху по трасі в діапазоні від 95 до 120 км/годину. Всі автомобілі повинні проїхати ділянку довжиною 25 км, причому перший і третій автомобілі рухаються по екрану праворуч наліво, а другий і четвертий рухаються зліва направо. Кожен наступний автомобіль починає рух, коли попередній автомобіль проїжджає свою ділянку траси. Гонка закінчується, коли четвертий автомобіль приходить до фінішу свого етапу. За результатами гонки на індикаторах повинні бути відображені час проходження частки траси кожним із учасників естафети та сумарний час, показаний командою.

2. Налаштувати програму та навести у звіті лицьову панель та діаграму.

ЛАБОРАТОРНА РОБОТА №16

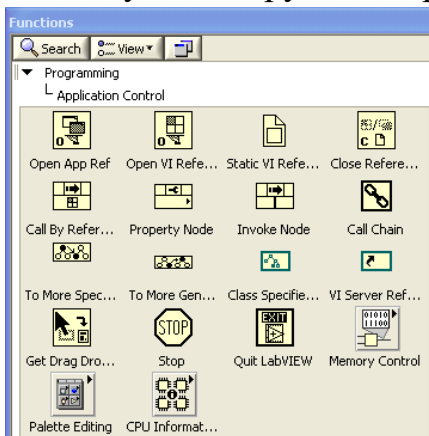
КЕРУВАННЯ ПРОГРАМОЮ В LabVIEW

1 Мета роботи: Отримання практичних навиків з керування програмою в середовищі LabVIEW.

2 Опис лабораторного макету: В ході лабораторної роботи використовується інструментарій пакету LabView 8.6

3 Необхідні довідкові дані

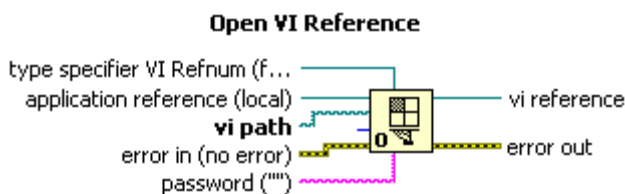
3.1 Функції керування програмою.



LabVIEW забезпечує широкий набір функцій взаємодії ВП між собою та іншими додатками. Функції, розміщені на панелі керування програмою (Application Control), дозволяють одним локальним або віддаленим ВП викликати функції та властивості інших ВП за їх посиланням. Ці функції дозволяють програмно за допомогою сервера ВП керувати властивостями програми (LabVIEW), віртуального приладу, елементів керування та індикації. Під керуванням розуміється динамічна зміна властивостей (атрибутів) ВП або об'єктів лицьової панелі, динамічний виклик та

завантаження ВП у пам'ять з наступним запуском та закриттям. Звернення до сервера може здійснюватися за допомогою технології ActiveX або за допомогою виклику за посиланням. Розглянемо деякі з цих функцій, які розташовані Functions-Programming-Application Control.

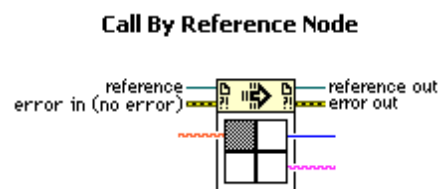
Функція Відкрити посилання на ВП (Open VI Reference) повертає посилання на ВП, спеціальний елемент керування або



глобальну змінну, визначені рядком імені або шляхом до місця розташування ВП на диску. На вхід описувач типу посилання на ВП (type specifier VI Refnum) необхідно

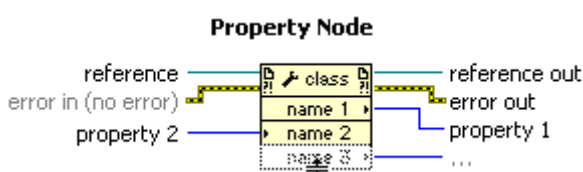
підключити елемент управління або константу посилання на ВП для створення посилання на ВП суворого типу, яке має бути передане функції

Функція виклику за посиланням (Call By Reference Node). Вхід шлях до ВП (vi path) рядок з ім'ям ВП, до якого створюється посилання, або шлях, що містить повний шлях цього ВП. Функція завантажує визначений чином ВП в пам'ять, якщо він ще не був відкритий, і повертає посилання на цей ВП.

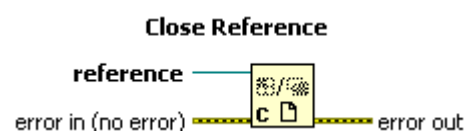


містить до таким

Функція Вузол виклику (Call By Reference Node) викликає ВП, заданий на вході посилання (reference). Посилання має бути суворого типу. У нижній частині функції знаходиться область, де відображається сполучна панель викликаного ВП. До терміналів цієї панелі, як і до терміналів підприладу, можуть бути підключені елементи керування та індикатори.

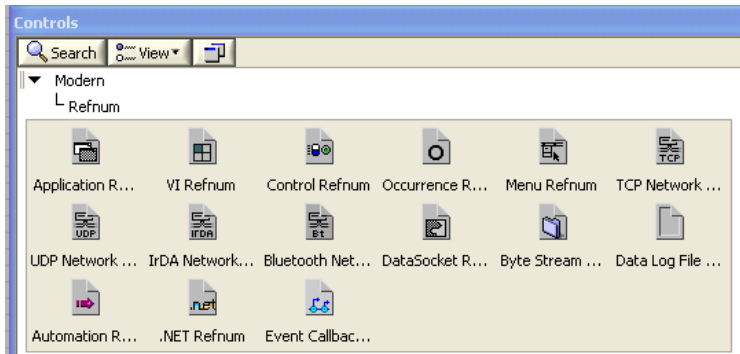


Вузол властивості (Property Node) отримує



(читає) та/або встановлює (записує) властивості за посиланням (reference). Вузол властивості автоматично адаптується до класу об'єкта, що задається за посиланням. Вузол виконує кожен термінал у порядку зверху донизу. Якщо в терміналі сталася помилка, то вузол зупиняється на цьому терміналі, повертає помилку та не виконує наступні термінали. Щоб ігнорувати помилку та продовжити виконання програми, можна встановити опцію Ігнорувати помилки всередині вузла (Ignore Errors Inside Node) контекстного меню.

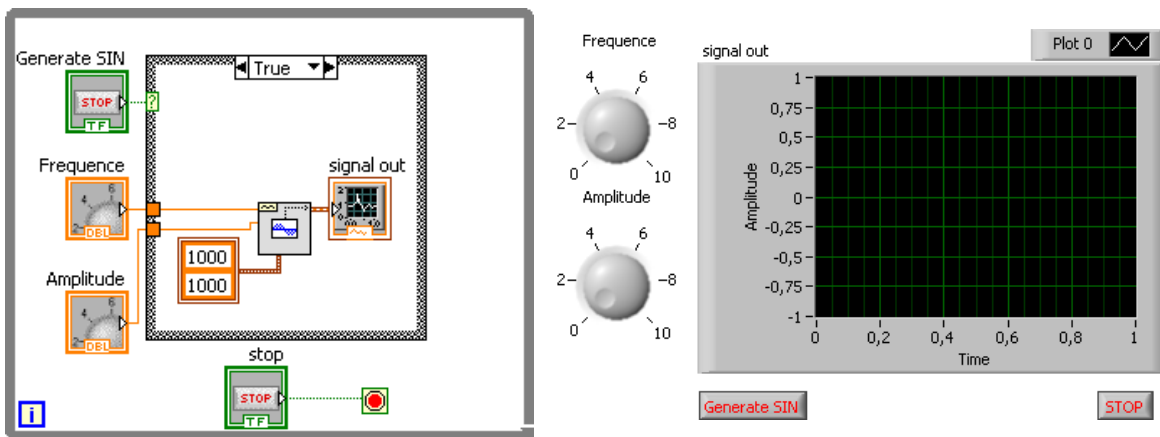
Функція Закрити посилання (Close Reference) закриває посилання, пов'язане з відкритим ВП, об'єктом ВП або відкритою програмою.




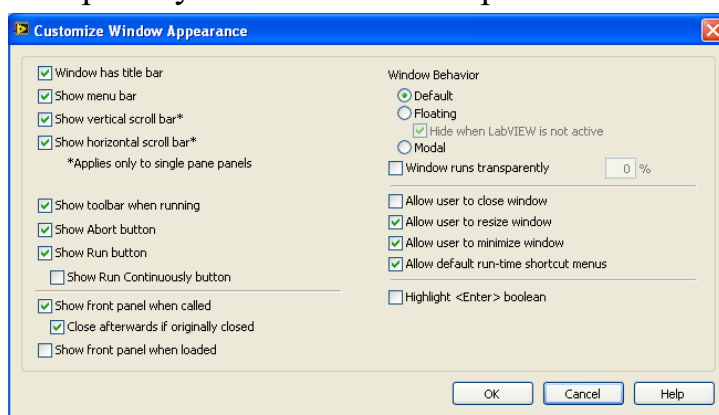
У LabVIEW є цілий ряд елементів керування посиланнями (Refnum Controls) розташований у підменю Control-Modern-Refnum. Вони призначені для роботи з файлами, директоріями, пристроями та мережевими з'єднаннями.

3.2 Динамічний виклик одного ВП з іншого

Для ознайомлення з процедурою динамічного виклику приладів заготовимо кілька допоміжних віртуальних приладів, які генерують різні види сигналів (синусоїдальний, косинусоїдальний і трикутний).



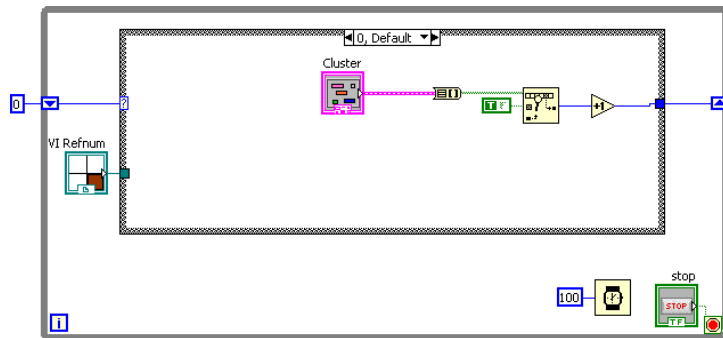
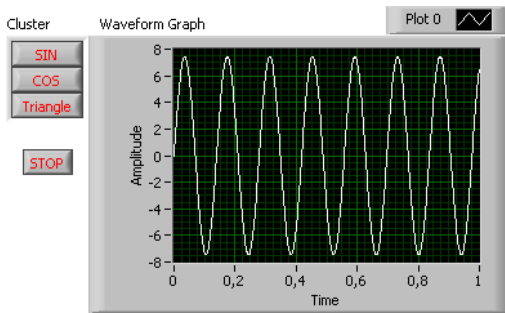
Для кожного ВП необхідно створити сполучну панель виду , для цього на лицьовій панелі приладу станьте на іконку у верхньому лівому кутку, натисніть праву клавішу миші та виберіть Show Connector. Потім ставши на сполучну панель, виберіть пункт Patterns і виберіть шаблон на 4 контакти. Перемістіть вказівник миші на нижній правий контакт сполучної панелі, натисніть ліву клавішу миші, а потім вкажіть Waveform Graph і знову натисніть на ліву клавішу миші.



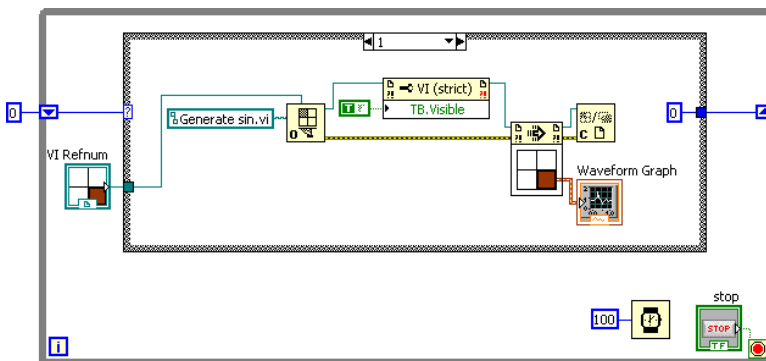
Для розробленого приладу необхідно встановити ряд

властивостей, виберіть із меню File підменю VI Properties. У вікні, яке відчинилось, виберіть категорію Window Appearance, виберіть пункт Custom, натисніть кнопку Customize і встановіть властивості як на малюнку. Натисніть кнопку OK і збережіть прилад з ім'ям, що асоціюється з синусом. Збережіть цей прилад з іменами пов'язаними з косинус і трикутним сигналом. Для перейменованих приладів на блок діаграмі замініть генератор синусоїдального сигналу на генератор трикутного та косинусоїдального сигналів відповідно.

Основний прилад викликатиме на вибір оператора один із допоміжних приладів, і відображатиме у себе на індикаторі результат роботи допоміжного приладу. Кнопки

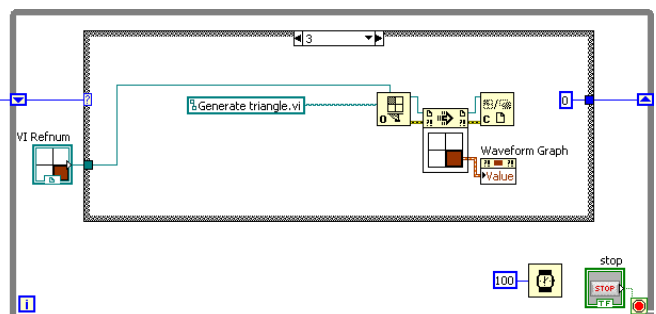
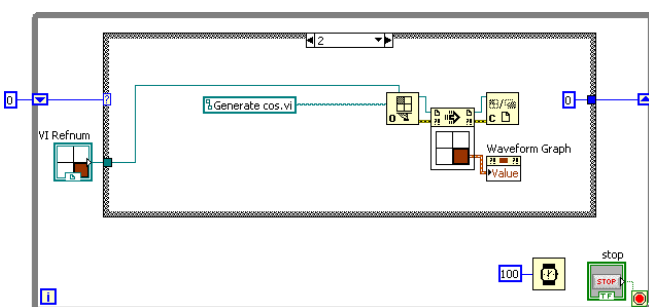


виклику одного з допоміжних приладів згруповані у кластер, стан якого аналізується у нульовій вкладці структури Case. Тут вихідний сигнал кластера перетворюється на одновимірний масив і за допомогою функції Search 1D Array визначається номер елемента масиву, який має стан True, певний номер збільшується на одиницю і використовується для вибору подальших дій. Якщо була



натиснута перша кнопка в кластері (її індекс 0), то виконуватиметься перша закладка структури Case. На малюнку наведено схему з'єднань для першої вкладки. Тут задіяні функції Open VI Reference, Call By Reference Node, Close Reference та вузол Property Node із властивістю Tool Bar-Visible.

Друга та третя закладки відрізняються лише ім'ям файлу, що викликається. Їх можна продублювати вибором якості Duplicate Case у структурі Case і замінити імена файлів, що викликаються.



VI Refnum розмістіть на лицьовій панелі основного приладу меню Control-Modern-Refnum. Станьте на нього вказівником миші, натисніть праву клавішу миші

та виберіть пункт Select VI Server Class-Browse і вкажіть один із розроблених раніше допоміжних приладів.

Запустіть основний прилад і, натиснувши послідовно кнопки вибору виду сигналу, перевірте працездатність методу динамічного виклику приладів. Поміняйте на першій вкладці константу True на False та зверніть увагу, як зміниться лицьова панель одного з допоміжних приладів.

4. Хід роботи

1) Створіть віртуальні прилади описані у третьому розділі. Під час виконання лабораторної роботи використовуйте можливості системи Help середовища LabView.

2) Складіть індивідуальний звіт про виконану роботу, який повинен включати:

- Лицьову панель розроблених приладів, з відображеною інформацією згідно з індивідуальним завданням;

- Панель діаграм розроблених приладів

3) Захист звіту відбувається індивідуально для кожного студента.

5. Контрольні питання:

1) Для чого використовується функція Відкрити посилання на ВП?

2) Для чого використовується функція Вузол виклику за посиланням?

3) Для чого використовується функція Закрити посилання?

6. Варіанти завдань

Завдання

Помістіть розроблені прилади в проєкт і створіть файл, що виконується. При налаштуванні програми яка розробляється стартовим приладом виступає основний прилад, допоміжні прилади мають бути занесені до категорії Always Included.

2. Налагодити програму та навести у звіті лицьову панель та діаграму.