

ОСНОВЫ ПРОГРАММИРОВАНИЯ В LABVIEW 8

Методические указания
к проведению лабораторного практикума
по курсу «Проектирование информационно-измерительных систем»
для студентов 5 курса факультета электронной техники
(специальность «Промышленная электроника»)

САРАНСК

2006

Составители: *канд. техн. наук, Н. Н. Беспалов, преподаватель М. В. Ильин*

Лабораторные работы «Основы программирования в LabView 8» по курсу «Проектирование информационно-измерительных систем». / Сост.: Н. Н. Беспалов, М. В. Ильин – Саранск: Ковылк.тип., 2006. – 47 с.

Содержатся методические указания к выполнению лабораторных работ по курсу «Проектирование информационно-измерительных систем» на основе курса National Instruments «LabVIEW. Основы 1». Лабораторные работы выполняются с использованием лабораторного стенда NI ELVIS. Предназначены для студентов V курса специальности «Промышленная электроника», а также для начального обучения студентов инженерных специальностей.

Работа подготовлена кафедрой автоматике. Авторы несут ответственность за точность предоставляемой информации.

Печатается по решению научно-методического совета Мордовского государственного университета им. Н. П. Огарева.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
Занятие 1. Введение в LabVIEW	
Упражнение 1.1 Частотный анализ	5
Занятие 2. Создание ВП.	
Упражнение 2-1. Преобразование °С в °F	8
Занятие 3. Создание подпрограмм ВП	
Упражнение 3.1 Подпрограмма ВП Преобразование °С в °F	11
Занятие 4. Многократные повторения и циклы.	
Упражнение 4-1. ВП Подсчет итераций	13
Упражнение 4-2. Доступ к данным предыдущих итераций	18
Занятие 5. Массивы	
Упражнение 5-1. ВП Работа с массивами	21
Занятие 6. Кластеры	
Упражнение 6-1 ВП Работа с кластерами	23
Занятие 7. Графическое отображение данных	
Упражнение 7-1. Мониторинг температуры	26
Упражнение 7-2. ВП График окружности	30
Занятие 8. Принятие решений в ВП.	
Упражнение 8-1. ВП Извлечение квадратного корня	32
Упражнение 8-2. ВП Узел Формулы	34
Занятие 9. Строки.	
Упражнение 9-1. ВП Компоновка строки	36
Занятие 10. Функции файлового ввода/вывода	
Упражнение 10-1. ВП Запись файла.	39
Упражнение 10-2 ВП Чтение файла	41
Упражнение 10-3. ВП Запись таблицы символов	43
Библиографический список	46

ВВЕДЕНИЕ

Платформа LabVIEW (Laboratory Virtual Instrument Engineering Workbench) фирмы National Instruments предназначена для разработки программного обеспечения для моделирования отдельных реальных физических объектов и создания информационно-измерительных систем и систем автоматизированного и автоматического управления.

Преимуществом LabVIEW над текстовыми языками, таких как C, Pascal и др., где программы составляются в виде строк текста, является то, что в LabVIEW программы создаются в виде графических диаграмм, в легкости восприятия алгоритма программистом.

Широкий спектр оборудования, поставляемого National Instruments, позволяет на основе платформы LabVIEW создавать в короткие сроки различную электронную аппаратуру с широкими функциональными возможностями.

Данное методическое пособие предназначено для студентов инженерных специальностей, начинающих изучать графический язык LabView. Оно содержит десять тематических разделов с упражнениями, предназначенными для лабораторного практикума, выполнение которого позволяет получить начальные сведения для освоения среды программирования.

Занятие 1. Введение в LabVIEW

Упражнение 1.1 Частотный анализ

Цель: открыть и запустить прибор.

1. Запустите LabVIEW (Пуск → Все программы → National Instruments LabVIEW 8.0). Появится диалоговое окно Getting Started.

2. Выберите Find Examples. На экране появится диалоговое окно поиска примеров ВП (рис. 1.1), разбитых по категориям.

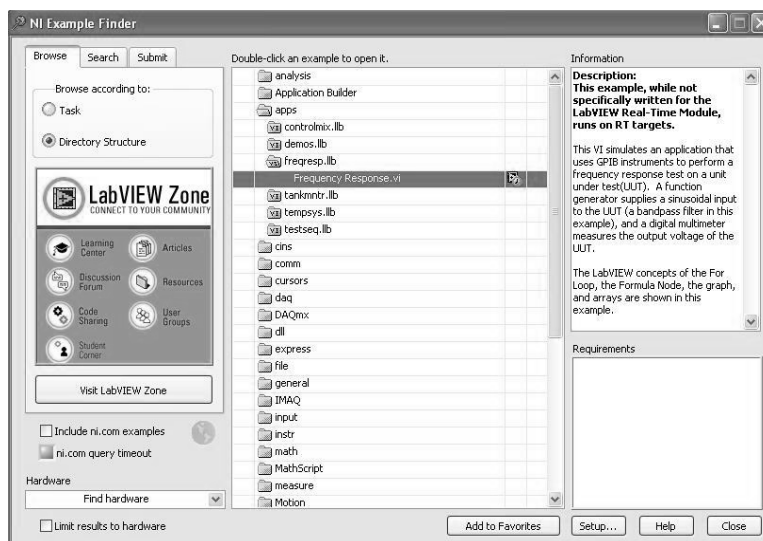


Рис. 1.1 – Окно поиска примеров ВП

3. Перейдите на закладку **Brows** (Обзор). Отметьте пункт **Directory Structure**. В списке примеров выберете папку **Apps**, в которой откройте библиотеку **Fereqresp.lib** и дважды щелкните на **Frequency Response VI**. Появится лицевая панель ВП Частотный анализ (рис. 1.2).

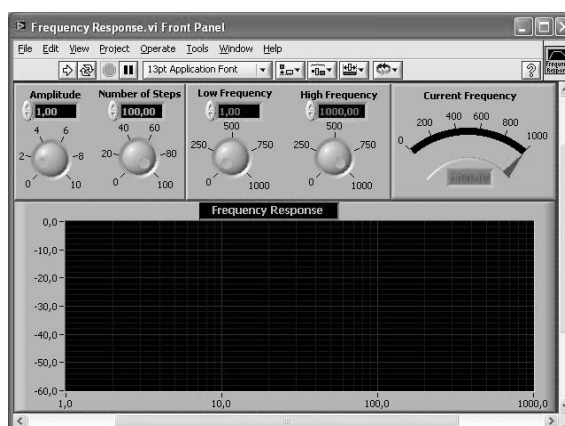





Рис. 1.2 Лицевая панель ВП Frequency Response VI

Лицевая панель

4. На инструментальной панели нажмите кнопку **Run** . Данный ВП моделирует посылку сигнала к измерительному прибору и регистрацию его отклика. Реакцию прибора в частотной области можно будет увидеть на графике лицевой панели.

5. С помощью инструмента УПРАВЛЕНИЕ  измените значение уставки амплитуды **Amplitude**. Изменить значение можно, либо переместив указатель кнопки в нужное положение, либо используя стрелки изменения значений элемента управления, либо введя число непосредственно в дисплей элемента.

Если число введено непосредственно в дисплей элемента, то необходимо нажать кнопку **Enter** , появившуюся на инструментальной панели. Иначе число не будет введено.

6. Нажать кнопку **Run** и запустить ВП. Изменяя значения других средств управления, находящихся на панели, исследовать работу ВП.

Блок-диаграмма

7. Перейдите на блок-диаграмму. Для этого выберите в главном меню **Window** → **Show Diagram** или введите <Ctrl-E> с клавиатуры.

Блок-диаграмма (рис. 1.3) содержит несколько основных объектов, включая подпрограммы ВП, функции и структуры.

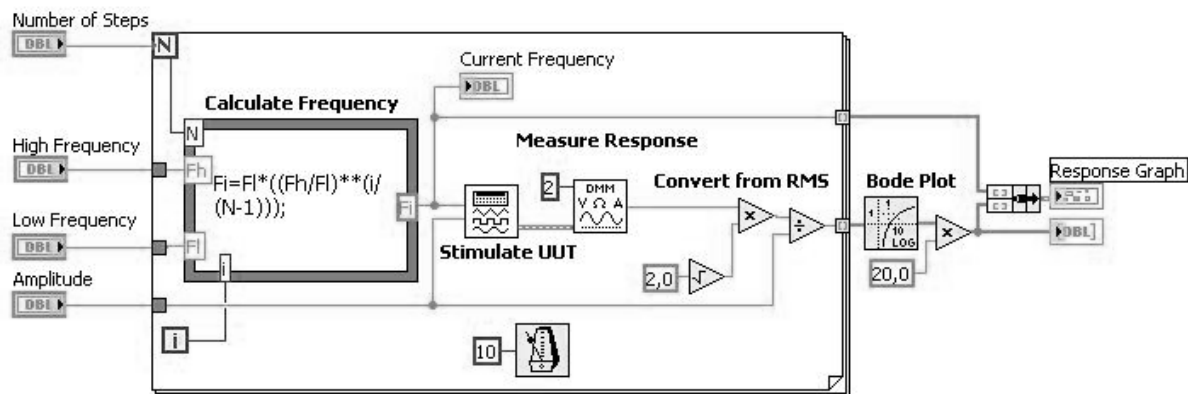


Рис. 1.3 – Блок-диаграмма ВП Frequency Response VI

8. С помощью инструмента УПРАВЛЕНИЕ дважды щелкните по иконке **DMM** (рис. 1.4).

Measure Response

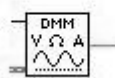


Рис. 1.4 – Иконка ВП DMM

Эта иконка – графическое представление подпрограммы **Demo Fluke 8840A VI**. После двойного щелчка откроется подпрограмма и на экране появится её лицевая панель (рис. 1.5).



Рис. 1.5 – Лицевая панель ВП DMM

Дизайн лицевой панели напоминает мультиметр. Вот почему программы LabVIEW называются виртуальными приборами.

Создавая модульные приложения LabVIEW, можно изменять только части приложения и/или многократно использовать эти части в других приложениях. Например, эта подпрограмма моделирует действие комбинированного прибора **Demo Fluke**, но пользователь может внести в него изменения, чтобы получить новые функции.

9. Выберите в главном меню пункты **File**→**Close** и закройте **Demo Fluke 8840AVI**.

10. Закройте ВП Frequency Response VI.

Конец упражнения 1.1

Занятие 2. Создание ВП

Упражнение 2-1. Преобразование °C в °F

Цель: Создать ВП

Ниже приведена последовательность действий для создания ВП, который будет преобразовывать значение температуры из градусов Цельсия в градусы Фаренгейта.

Лицевая панель

1. Выберите пункт главного меню File»New»VI, чтобы открыть новую лицевую панель.

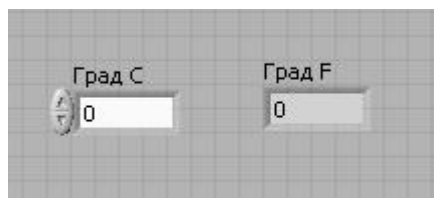
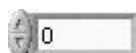


Рис. 2.1– Лицевая панель ВП

2. Создайте цифровой элемент управления. Он будет использован для ввода значений температуры в °C.

a. Выберите цифровой элемент управления в разделе палитры Элементов в подразделе Controls»Numeric (Числовые элементы). Для вывода на экран палитры Controls (Элементы) следует щелкнуть правой кнопкой мыши по рабочему пространству лицевой панели.



b. Поместите цифровой элемент управления на лицевую панель.



c. В поле собственной метки элемента управления напечатайте «Град С» и щелкните мышью в свободном пространстве лицевой панели или нажмите кнопку **Enter**, показанную слева, на инструментальной панели. Если сразу после создания элемента не присвоить имя его собственной метке, то LabVIEW присвоит имя, заданное по умолчанию. Собственная метка в любое время доступна для редактирования, оно производится с помощью инструмента ВВОД ТЕКСТА, показанного слева.



4. Создайте цифровой элемент отображения данных. Он будет использован для отображения значений температуры в °F.



a. Выберите цифровой элемент отображения в палитре Элементов в подразделе Controls»Numeric (Числовые элементы).

- b. Поместите элемент отображения данных на лицевую панель.
- c. В поле собственной метки элемента управления напечатайте «Град F» и щелкните мышью в свободном пространстве лицевой панели или нажмите кнопку Enter.



На блок-диаграмме LabVIEW создаст терминалы данных соответствующие элементам управления и отображения. Терминалы данных представляют тип данных соответствующих элементов. Например, терминал данных DBL, показанный слева, представляет тип числовых данных двойной точности с плавающей запятой.

Блок-диаграмма

5. Перейдите на блок-диаграмму, выбрав пункты главного меню Window» Show Diagram.

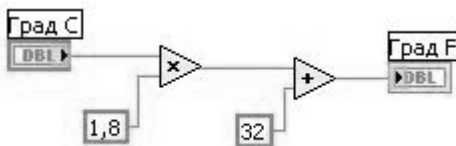


Рис. 2.2 – Блок-диаграмма ВП



6. Выберите функцию **Multiply** (Умножение) из палитры Функций в разделе **Functions»Programming»Numeric** (Арифметические функции). Поместите ее на блок-диаграмму. Для вывода на экран палитры **Functions** (Функций) следует щелкнуть правой кнопкой мыши в рабочем пространстве блок-диаграммы.



7. Выберите функцию **Add** (Сложение) из палитры Функций в разделе **Functions»Programming»Numeric** (Арифметические функции). Поместите ее на блок-диаграмму.



8. Выберите числовую константу из палитры Функций в разделе **Functions»Programming»Numeric** (Арифметические функции). Поместите две числовые константы на блок-диаграмму. После размещения числовой константы на блок-диаграмме поле ввода ее значений подсвечивается и готово для редактирования.

- Одной константе присвойте значение 1,8, другой 32,0.
- Если значение в константу не введено сразу после ее размещения на блок-диаграмме, следует использовать инструмент ВВОД ТЕКСТА.



9. Соедините объекты блок-диаграммы с помощью инструмента СОЕДИНЕНИЕ, показанного слева:

- Для соединения двух объектов на блок-диаграмме инструментом СОЕДИНЕНИЕ следует нажать левой кнопкой мыши на одном из объектов и, не отпуская кнопку, перевести инструмент на другой объект. Последовательность соединения не имеет значения.
- Проводник данных можно изгибать, закрепив его щелчком мыши и переместив курсор в перпендикулярном направлении. Нажав пробел, можно переключить направление движения проводника.
- Для идентификации полей ввода/вывода данных узла или функции следует щелкнуть на выбранном объекте правой кнопкой мыши и в контекстном меню выбрать пункт **Visible Items»Terminals**. Чтобы вернуться к прежнему виду объекта, следует повторить ту же последовательность действий.
- При наведении инструмента СОЕДИНЕНИЕ на поле ввода/вывода данных поле начинает мигать, показывая, что щелчок мыши подключит к нему проводник данных. При этом появляется всплывающая подсказка с указанием имени поля ввода/вывода.
- Для отмены начала соединения следует нажать <Esc>, либо щелкнуть правой кнопкой мыши или щелкнуть инструментом СОЕДИНЕНИЕ на поле источник соединения.

10. Перейдите на лицевую панель, выбрав в главном меню пункт **Window»Show Panel**.

11. Сохраните ВП, он будет использоваться позднее.

a. Выберите пункт главного меню **File»Save**.

b. Укажите каталог d:\student\фамилия\.

c. В диалоговом окне введите *Преобразование C в F (начало)*. vi

d. Нажмите кнопку Save.

Запуск ВП



12. Введите число в элемент управления и запустите ВП.

a. Для ввода числа в элемент управления следует использовать инструмент УПРАВЛЕНИЕ, показанный слева, или инструмент ВВОД ТЕКСТА.



b. Нажмите кнопку **Run**, показанную слева, чтобы запустить ВП.

c. Введите несколько разных значений температуры и запустите ВП.

13. Закройте ВП, выбрав пункт главного меню File»Close.

Конец упражнения 2-1

Занятие 3. Создание подпрограмм ВП

Упражнение 3.1 Подпрограмма ВП Преобразование °C в °F

Цель: Создать иконку и настроить соединительную панель для возможности использования ВП в качестве подпрограммы ВП.

В этом уроке представлена последовательность действий по созданию иконки и настройке соединительной панели для созданного ВП, который переводит значение измеренной температуры из °C в °F.

Лицевая панель

1. Выберите пункт главного меню **File»Open**, укажите папку **d:\student\фамилия** и выберите файл *Преобразование C в F (начало).vi*
Если закрыты все ВП, следует нажать кнопку **Open VI** (Открыть ВП) в диалоговом окне **Lab VIEW**.

Появится следующая лицевая панель:

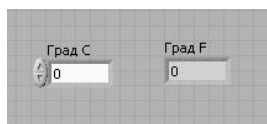


Рис. 3.1 – Лицевая панель ВП Преобразование C в F(начало).vi

Иконка и соединительная панель

2. Щелкните правой кнопкой мыши по иконке ВП и в контекстном меню выберите пункт **Edit Icon** (Редактирование иконки). Появится диалоговое окно редактора иконки **Icon Editor**.
3. Дважды щелкните правой кнопкой мыши по инструменту **ВЫБОР** (показан слева).
4. Нажав кнопку **<Delete>** очистите область редактирования иконки.
5. Дважды щелкните по инструменту **ПРЯМОУГОЛЬНИК** (показан слева), чтобы обвести область редактирования границей выбранного цвета.
6. Создайте следующую иконку:

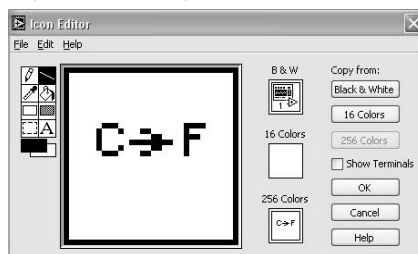


Рис. 3.2 – Иконка ВП преобразования температур

- a. Введите текст инструментом **ВВОД ТЕКСТА**, который показан слева.
- b. Напечатайте «C» и «F».

с. Для выбора размера шрифта дважды щелкните левой кнопкой мыши по инструменту ВВОД ТЕКСТА.



d. Чтобы нарисовать стрелку, воспользуйтесь инструментом КАРАНДАШ.

е. Для передвижения текста и стрелки по полю редактирования иконки используйте инструмент ВЫБОР и стрелки на клавиатуре.

f. В разделе **Copy from** (Копировать) выберите B & W (черно-белую) иконку и **256 Colors** (256-цветный режим) для создания черно-белой иконки, которую LabVIEW использует в случае отсутствия цветного принтера.

g. В разделе **Copy from** (Копировать) выберите **16 Colors** и **256 Colors**.

h. После завершения редактирования иконки нажмите кнопку ОК и закройте **Icon Editor**. Новая иконка появится в правом верхнем углу обеих панелей.



7. Перейдите на лицевую панель, щелкните правой кнопкой мыши на иконке и выберите пункт **Show Connector** (Показать поля ввода/вывода данных) из контекстного меню. Количество отображаемых LabVIEW полей ввода/вывода данных соответствует количеству элементов на лицевой панели. Например, лицевая панель этого ВП имеет два элемента **Град С** и **Град F** и LabVIEW выводит в соединительной панели два поля, показанные слева.

8. Элементам управления и отображения данных назначьте соответственно поля ввода и вывода данных.

a. В пункте главного меню **Help** (Помощь) выберите **Show Context Help** (контекстную подсказку) и выведите на экран окно Context Help (контекстной справки) для просмотра соединений.

b. Щелкните левой кнопкой мышки на левом поле соединительной панели. Инструмент УПРАВЛЕНИЕ автоматически поменяется на инструмент СОЕДИНЕНИЕ, а выбранное поле окрасится в черный цвет.

с. Щелкните левой кнопкой мыши по элементу Град С. Левое поле станет оранжевым и выделится маркером.

d. Щелкните курсором по свободному пространству. Маркер исчезнет, и поле окрасится в цвет данных типа соответствующего элемента управления.

е. Щелкните левой кнопкой мыши по правому полю соединительной панели и элементу Град F. Правое поле станет оранжевым.

f. Щелкните курсором по свободному пространству. Оба поля останутся оранжевыми.

g. Наведите курсор на область полей ввода/вывода данных. Окно **Context Help** (контекстной справки) покажет, что оба поля соответствуют типу данных двойной точности с плавающей запятой.

9. Выберите пункт главного меню File»Save. Сохраните ВП под именем *Преобразование С в F.vi*, он будет использоваться позднее.

10. Выберите пункт главного меню File»Close. Закройте ВП.
Конец упражнения 3-1

Занятие 4. Многократные повторения и циклы

Упражнение 4-1. ВП Подсчет итераций

Цель: использование терминала выходных данных цикла While.

Создайте ВП, который генерирует случайные числа до тех пор, пока одно из них не окажется равным значению, введенному в элемент управления. При этом должно отображаться количество итераций, выполненное циклом.

Лицевая панель

1. Откройте новую лицевую панель. Создайте лицевую панель, разместив на ней элементы управления и отображения, как показано ниже на рис. 4.1.

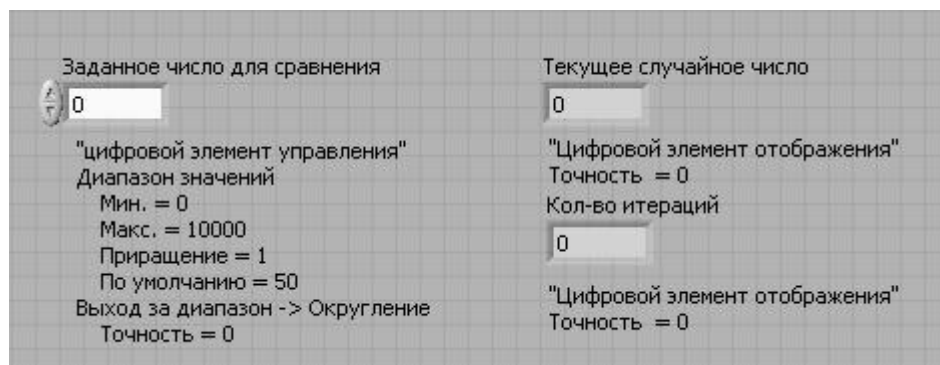


Рис. 4.1 – Лицевая панель ВП Подсчёт итераций

- a. Поместите на лицевую панель числовой элемент управления, находящийся на палитре **Controls»Numeric**. Назовите элемент **Заданное число для сравнения**. Этот элемент задает число, с которым будет проводиться сравнение.
- b. Поместите на лицевую панель числовой элемент отображения, находящийся на палитре **Controls»Numeric**. Назовите элемент **Текущее случайное число**. Этот элемент отображает текущее значение, выданное функцией **Генератор случайных чисел**.
- c. Поместите еще один числовой элемент отображения на лицевую панель. Назовите элемент **Кол-во итераций**. Этот элемент показывает номер текущей итерации.

Установка диапазона данных

Чтобы значения элемента **Заданное число для сравнения** не выходили за рамки диапазона значений, выдаваемых функцией **Генератор случайных чисел**, следует использовать диалоговое окно **Data Range**. Выполните следующие шаги для настройки диапазона выходных значений элемента **Заданное число для сравнения** от 0 до 100000 с шагом изменения 1 и значением по умолчанию равным 50.

2. Щелкните правой кнопкой мыши по элементу **Заданное число для сравнения**. Из контекстного меню выберите пункт **Data Range**. Появится диалоговое окно, показанное ниже.

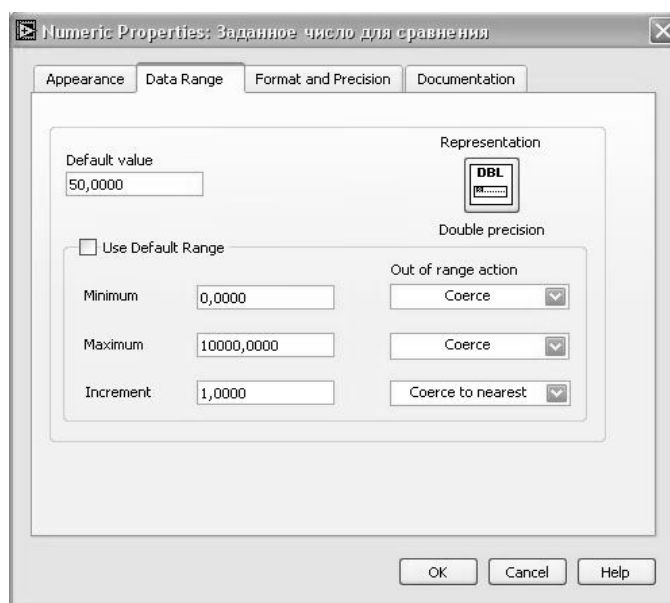


Рис. 4.2 – Диалоговое окно Data Range

3. Снимите выделение с пункта **Use Defaults** (использовать значения по умолчанию).

4. Выберите пункты, показанные в этом примере диалогового окна.
 - a. Установите **Default Value** (значение по умолчанию) равным 50.
 - b. Установите **Minimum Value** (минимальное значение) равным 0 и выберите Coerce.
 - c. Установите **Maximum Value** (максимальное значение) равным 10000 и выберите Coerce.
 - d. Установите **Increment** (значение приращения) равным 1 и выберите Coerce to Nearest.

5. Выберите раздел **Format and Precision** (формат и точность).

Установка количества знаков после запятой

По умолчанию, LabVIEW отображает числовые элементы управления и отображения в виде десятичных чисел с точностью до двух знаков после запятой (3,14). С помощью опции **Format&Precision** можно изменить точность и вид представления значений элементов (**научная нотация, инженерная нотация, формат времени**).

6. Щелкните правой кнопкой мыши по элементу **Текущее случайное число** и выберите в контекстном меню пункт **Format&Precision**. Появится следующее диалоговое окно **Format&Precision** (рис. 4.3).

7. Сделайте настройки, показанные ниже.

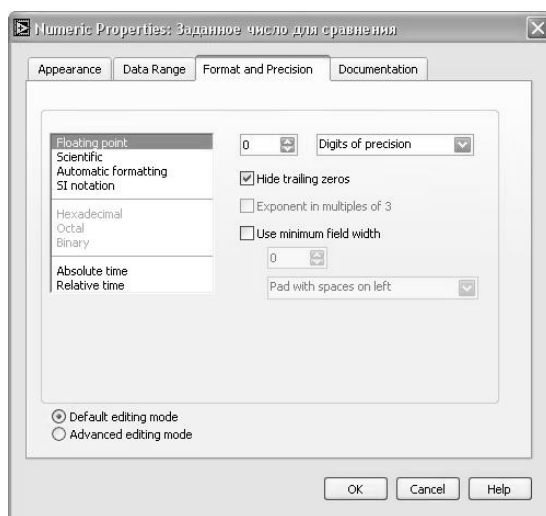


Рис. 4.3 – Диалоговое окно Format&Precision

В поле ввода **Digits of Precision** следует ввести значение 0.

8. Повторите шаги 6 и 7 для элементов отображения **Текущее случайное число** и **Кол-во итераций**.

Блок-диаграмма

9. Создайте блок-диаграмму, как показано на рисунке

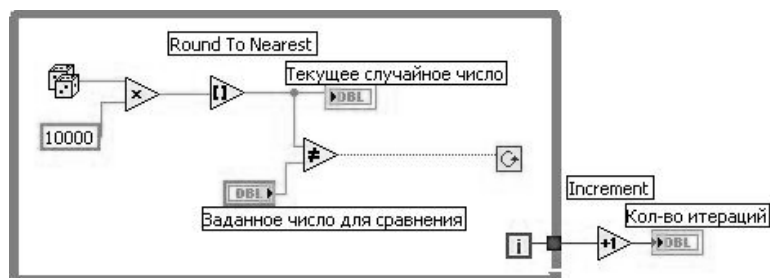







Рис. 4.4 – Блок-диаграмма ВП Подсчёт итераций


 Поместите на блок-диаграмму функцию **Random Number** (Генератор случайных чисел), расположенную на палитре **Функций** в разделе **Function» Programming» Numeric**. Эта функция генерирует случайные числа в пределах от 0 до 1


 Поместите на блок-диаграмму функцию **Multiply** (Умножение), расположенную в палитре **Функций** в разделе **Function» Programming» Numeric**. Эта функция умножает текущее значение с выхода функции **Random Number** (Генератор случайных чисел) на 10000.

 Создайте константу. Для этого следует навести курсор на поле ввода данных функции **Multiply** (Умножение), щелкнуть по нему правой кнопкой мыши и выбрать в контекстном меню пункт **Create» Constant** С помощью инструмента **ВВОД ТЕКСТА** присвойте ей значение 10000.

 Поместите на блок-диаграмму функцию **Round To Nearest** (Округление до ближайшего целого), расположенную в палитре **Функций** в разделе **Function» Numeric**. Эта функция будет округлять полученное в пределах от 0 до 10000 случайное число до ближайшего целого числа.

 Поместите на блок-диаграмму функцию **Not Equal? (≠)**, расположенную в палитре **Функций** в разделе **Function» Programming» Comparison**. Эта функция предназначена для сравнения случайного числа с числом, введенным в элемент управления **Заданное число для сравнения**. Если значения не равны, функция выдает значение **TRUE**.

 Поместите на блок-диаграмму цикл **While**, расположенный в палитре **Функций** в разделе **Function» Programming» Structures**. Наведите курсор на терминал условия выхода, щелкните по нему правой кнопкой мыши и выберите пункт **Continue if True** (Продолжение если Истина).

 Подсоедините терминал счетчика итераций к границе области цикла **While**. На границе цикла появится синий прямоугольник. Терминал выходных данных цикла присоединен к функции приращения. При выполнении цикла счетчик итераций получает приращение равное 1. После завершения цикла значение счетчика итераций передается на выход через терминал выхода цикла. Вне тела цикла значение счетчика итераций увеличивается на единицу для отображения количества выполненных итераций.



Поместите на блок-диаграмму функцию Increment (Приращение на 1), расположенную в палитре Функций в разделе **Function» Programming» Numeric**. Эта функция добавляет 1 к значению счетчика итераций после завершения выполнения цикла. Следует обратить внимание, что на терминале элемента Кол-во итераций имеется серая точка, означающая, что LabVIEW автоматически осуществляет преобразование типа данных счетчика итераций к типу данных терминала элемента Кол-во итераций. Подробнее о приведении типов данных можно прочитать в разделе В "Цикл For (с фиксированным числом итераций)".

10. Сохраните ВП под именем *Подсчет итераций.vi*.

Запуск ВП

11. Перейдите на лицевую панель и измените значение элемента Заданное число для сравнения.

12. Запустите ВП. Измените значение элемента Заданное число для сравнения и запустите ВП снова. При этом элемент **Текущее случайное число** обновляется после каждой итерации цикла, потому что его терминал данных расположен внутри тела цикла. Значение же элемента **Кол-во итераций** обновляется после завершения цикла, потому что терминал данных этого элемента расположен вне тела цикла.



13. Чтобы посмотреть, как ВП обновляет значения элементов отображения информации, необходимо запустить ВП в режиме анимации. Для этого следует нажать на инструментальной панели кнопку **Highlight Execution**, показанную слева. Режим отладки анимирует поток данных, проходящих по блок-диаграмме. Таким образом, имеется возможность наблюдать изменения значений на каждом этапе их генерации.

14. Измените значение элемента Заданное число для сравнения таким образом, чтобы оно с увеличением на 1 выходило за установленный диапазон значений от 0 до 10000.

15. Запустите ВП. Lab VIEW автоматически приведет новое значение к ближайшему значению в указанном диапазоне входных данных элемента.

16. Закройте ВП.

Конец упражнения 4-1

Упражнение 4-2. Доступ к данным предыдущих итераций

Цель: Использование сдвиговых регистров и узлов обратной связи для организации доступа к значениям на предыдущих итерациях цикла For.

Лицевая панель

Создайте лицевую панель, как показано на рис. 4.5, поместив два элемента отображения информации.

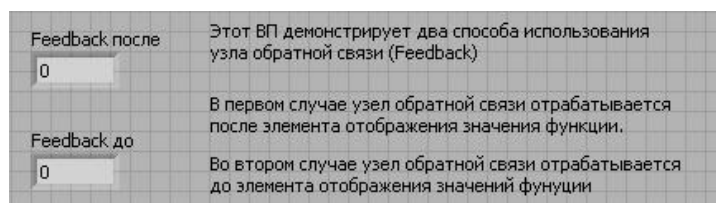


Рис. 4.5 – Лицевая панель ВП

Блок-диаграмма

Создайте блок-диаграмму, показанную на рис. 4.6.

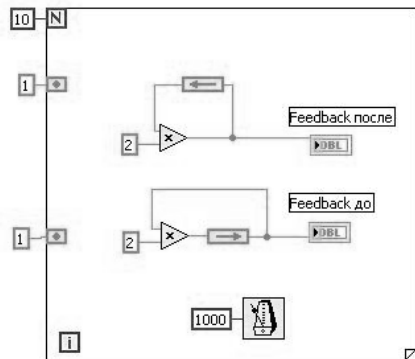


Рис. 4.6 – Блок-диаграмма ВП

1. Поместите на блок-диаграмму цикл **For**.
2. Задайте количество итераций 10.
3. Поместите две функции умножения.
4. Задайте коэффициент умножения равный 2 для каждого умножителя.
5. Соедините вход первого умножителя с его выходом. Связь образуется через узел обратной связи (**Feedback node**).
6. Поместите узел обратной связи (**Feedback node**) из палитры (**Structures**).

7. Соедините выход второго умножителя с входом узла обратной связи.
8. Замкните обратную связь на вход второго умножителя.
9. Появившимся входным терминалом на левой границе цикла, инициализирующие начальное значение регистра обратной связи, задайте значения 1.
10. Для замедления выполнения программы, поместите на блок-диаграмму **функцию ожидания (Wait Until Next ms Multiple)** из панели **Time & Dialog**. Задайте интервал между итерации равный 1000 мс. Можно также использовать режим **Highlight Execution** (анимации выполнения блок-диаграммы).

На данной блок-диаграмме один и тот же процесс выполняется дважды, при этом узел обратной связи помещен в различных местах соединения.

Запуск ВП

Запустите ВП. Программа в верхней части сначала считывает значение узла обратной связи, инициализированного значением 1. Затем это значение передается функции **Multiply** (умножение).

Программа в нижней части сначала считывает значения узла обратной связи, инициализированного значением 1. Затем это значение передается на цифровой элемент отображения. Функция **Multiply** (умножение) не будет выполняться до следующей итерации цикла.

1. Активируйте режим анимации выполнения блок-диаграммы, нажав на кнопку **Highlight Execution**. Запустите ВП еще раз для наблюдения порядка выполнения программы. Отключите режим анимации для работы ВП в нормальном режиме.

2. Замените узел обратной связи сдвиговым регистром, как показано на следующей блок-диаграмме (рис. 4.7):

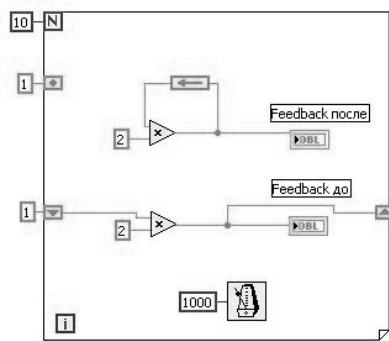


Рис. 4.7 – Блок-диаграмма ВП со сдвиговым регистром

- a. Выделите нижний узел обратной связи и нажмите клавишу **<Delete>**, чтобы удалить его.
- b. Щелкните правой кнопкой мыши по границе цикла и выберите пункт контекстного меню **Add Shift Register**.
- c. Инициализируйте сдвиговый регистр значением 1.
- d. Переименуйте верхний и нижний элементы отображения соответственно **Узел обратной связи** и **Сдвиговый регистр**.

3. Запустите ВП. Обратите внимание, что узел обратной связи и сдвиговый регистр выполняют одинаковые функции.

4. Не закрывайте ВП, перейдите к выполнению дополнительных.

Дополнительно

1. Модифицируйте сдвиговый регистр, чтобы он отображал последние три итерации цикла **For**, как показано на блок-диаграмме (рис.4.8):

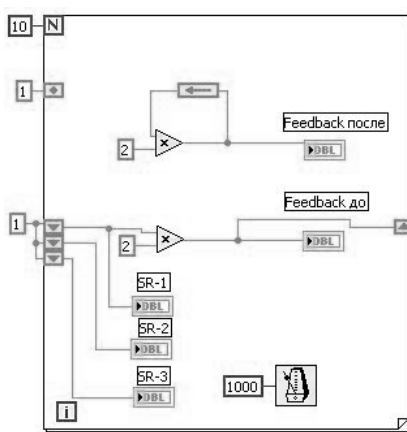


Рис. 4.8 – Блок-диаграмма ВП со сдвиговым регистром, отображающий последних три итерации

- a. Измените размер левого сдвигового регистра до трех элементов.
 - b. Инициализируйте все элементы сдвигового регистра значением 1.
 - c. Щелкните правой кнопкой мыши по каждому элементу и выберите пункт контекстного меню **Create»Indicator**. Назовите каждый элемент отображения.
2. Запустите ВП.
 3. Закройте ВП, не сохраняя его.

Конец упражнения 4-2

Занятие 5. Массивы

Упражнение 5-1. ВП Работа с массивами

Цель: создание массивов и знакомство с функциями обработки массивов

Выполните следующие шаги для создания ВП, который формирует массив случайных чисел, масштабирует полученный массив и выделяет из него подмножество.

Лицевая панель

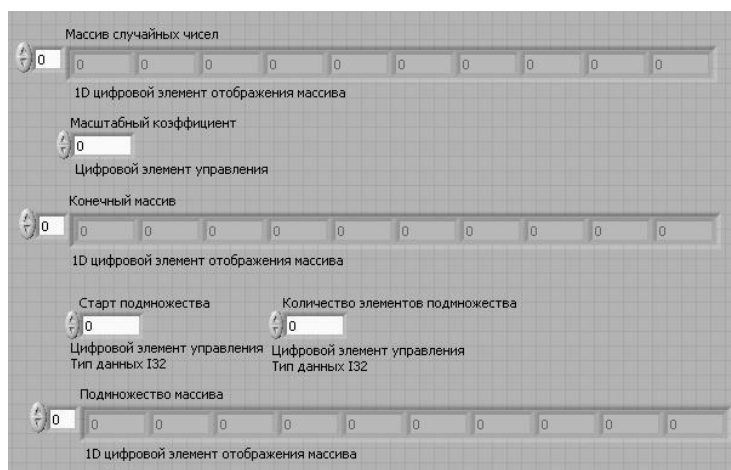


Рис. 5.1 – Лицевая панель ВП

1. Откройте новый ВП и создайте лицевую панель, как показано ниже.
 - a. В палитре **Controls»Modern»Array, Matrix & Cluster** выберите шаблон массива.
 - b. Созданному массиву присвойте имя **Массив случайных чисел**.
 - c. Поместите внутрь шаблона массива цифровой элемент отображения, расположенный в палитре **Controls»Numeric**.
 - d. С помощью инструмента ПЕРЕМЕЩЕНИЕ измените размер массива таким образом, чтобы он содержал 10 элементов.
 - e. Нажмите и удерживайте клавишу **<Ctrl>** и, перемещая элемент **Массив случайных чисел**, создать две его копии.
 - f. Копиям присвойте имена **Конечный Массив** и **Подмножество Массива**.
 - g. Создайте три цифровых элемента управления и присвойте им имена **Масштабный коэффициент**, **Старт подмножества** и **Количество элементов подмножества**.
 - h. Щелкните правой кнопкой мыши по элементам **Старт подмножества** и **Количество элементов подмножества**, в контекстном меню выберите пункт **Representation**, затем пункт **I32**.
 - i. Значения элементов управления данных **пока не** изменяйте.

Блок-диаграмма

2. Постройте блок-диаграмму, как показано ниже.



Рис. 5.2 – Блок-диаграмма ВП



Выберите функцию **Random Number (0-1)**, расположенную в палитре **Functions»Programming»Numeric**. Эта функция будет генерировать случайное число в пределах от 0 до 1.



Выберите цикл **For**, расположенный в палитре **Functions»Programming»Structures**. Этот цикл на терминале выхода накапливает массив из 10 случайных чисел. Терминалу количества итераций присвойте значение 10.



Выберите функцию **Array Subset**, расположенную в палитре **Functions»Programming»Array**. Эта функция выдает подмножество массива, начиная со значения, введенного в элементе **Старт подмножества** и будет содержать количество элементов, указанное в элементе **Количество элементов подмножества**.

3. Сохраните ВП под именем *Работа с массивами.vi*

Запуск ВП

4. Перейдите на лицевую панель, измените значения элементов управления и запустите ВП.

Цикл **For** совершит 10 итераций. **Каждая итерация** создаст случайное число и сохранит его в терминале выхода **из цикла**. В элементе **Массив случайных чисел** отобразится массив из 10 случайных чисел. ВП умножит каждое значение этого массива на число, введенное в элемент управления **Масштабный коэффициент**, для создания массива, отображаемого в индикаторе **Конечный массив**. ВП выделит подмножество из получившегося массива, начиная со значения в элементе **Старт подмножества**, длиной, указанной в элементе **Количество элементов подмножества**, и отобразит это подмножество в индикаторе **Подмножество массива**.

5. Закройте ВП.

Конец упражнения 5-1

Занятие 6. Кластеры

Упражнение 6-1 ВП Работа с кластерами

Цель: Создание кластеров на лицевой панели. Используя функции обработки кластеров, собирать и демонтировать кластеры

Лицевая панель

1. Откройте новый ВП и создайте лицевую панель, как показано ниже.

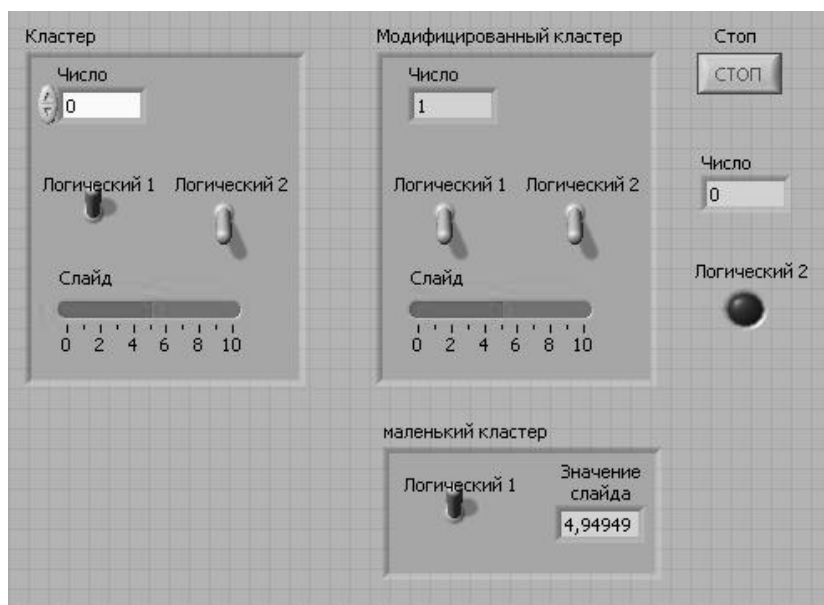


Рис. 6.1 – Лицевая панель ВП

- a. Поместите на лицевую панель кнопку «Стоп» и круглый светодиод.
- b. Из палитры **Controls»Modern»Array, Matrix & Cluster** выберите шаблон кластера.
- c. Объекты лицевой панели, показанные на иллюстрации, поместите в шаблон кластера.
- d. Создайте и переименуйте копию элемента **Кластер** в **Модифицированный Кластер**. После этого щелкните правой кнопкой мыши по границе шаблона кластера **Модифицированный Кластер** и выберите из контекстного меню пункт **Change to Indicator**.
- e. Повторите пункт d для создания элемента **Маленький кластер**. Измените его, как показано на рисунке.

2. Проверьте порядковые номера элементов в кластерах **Кластер** и **Маленький кластер**. Порядковые номера элементов кластера **Модифицированный кластер** и **Кластер** должны совпадать.

- a. Щелкните правой кнопкой мыши по границе шаблона каждого кластера, из контекстного меню выберите пункт **Reorder Controls in Cluster**.
- b. Порядковые номера элементов установите, как показано на рис. 6.2.

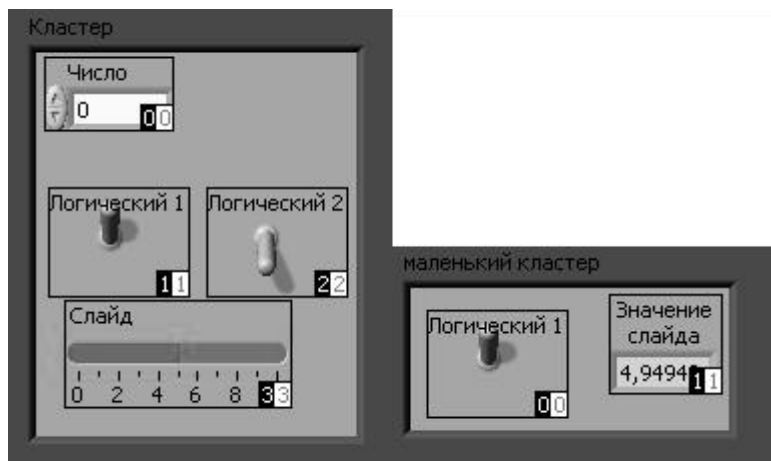


Рис. 6.2 – Порядковые номера элементов в кластере

Блок-диаграмма

3. Создайте блок-диаграмму, как показано на рис. 6.3.

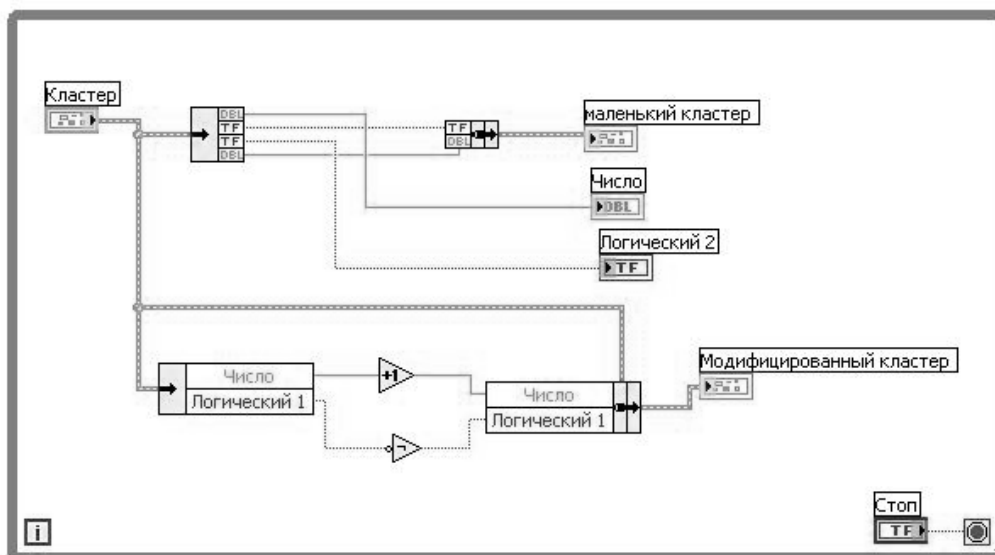


Рис. 6.3 – Блок-диаграмма ВП



Из палитры **Functions»Programming»Cluster** выберите функцию **Unbundle**. Эта функция разъединяет кластер. Измените размер этой функции до **четырёх** полей ввода данных или соедините терминал данных кластера с функцией для автоматического добавления полей ввода данных.



Из палитры **Functions» Programming»Cluster** выберите функцию **Bundle**. Эта функция объединит элементы в кластер **Маленький кластер**.



Из палитры **Functions»Programming»Cluster** выберите функцию **Unbundle by Name**. Эта функция выделит два элемента из кластера **Кластер**. Измените размер функции до двух полей вывода данных. Если имена в полях вывода данных отличаются от показанных на иллюстрации, следует щелкнуть правой кнопкой мыши по имени элемента и в контекстном меню войти в раздел **Select Item**.



Из палитры **Functions»Programming»Numeric** выберите функцию **Increment**. Эта функция добавит 1 к значению элемента **Число**.



Из палитры **Functions»Programming»Boolean** выберите функцию **Not**. Эта функция выдаст логическое отрицание элемента **Логический 1**.



Из палитры **Functions»Programming»Cluster** выберите функцию **Bundle by Name**. Эта функция изменит значения элементов **Число и Логический** в кластере **Кластер** и создаст кластер **Модифицированный кластер**. Измените размер этой функции на два поля ввода данных. Если имена в полях вывода данных отличаются от показанных на иллюстрации, следует щелкнуть правой кнопкой мыши по имени элемента и в контекстном меню войти в раздел **Select Item**.

4. Сохраните ВП под именем *Работа с кластерами, vi*

Запуск ВП

5. Перейдите на лицевую панель и запустите ВП.
6. Поменяйте значения элементов в кластере **Кластер** и запустите ВП.
7. Закройте ВП.

Конец упражнения 6-1

Занятие 7. Графическое отображение данных

Упражнение 7-1. Мониторинг температуры

Цель: приобретение навыков по использованию графика Диаграмм для получения и отображения данных.

Лицевая панель

1. Откройте новый ВП и создайте лицевую панель, как показано ниже на рис. 7.1.

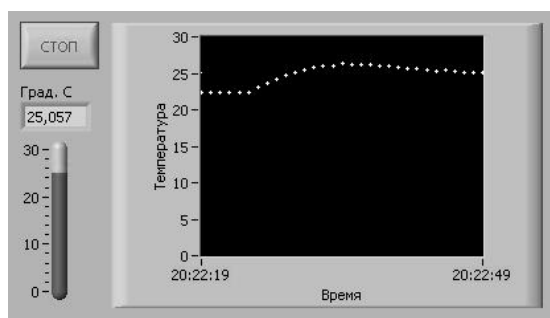


Рис. 7.1 – Лицевая панель ВП

1.1 Выберите график Диаграмм (**Waveform Chart**) из палитры **Controls»Modern»Graph** и поместите его на лицевую панель.

1.2. Щелкните правой кнопкой мыши на графике Диаграмм и настройте параметры в соответствии с рис. 7.2 и 7.3.

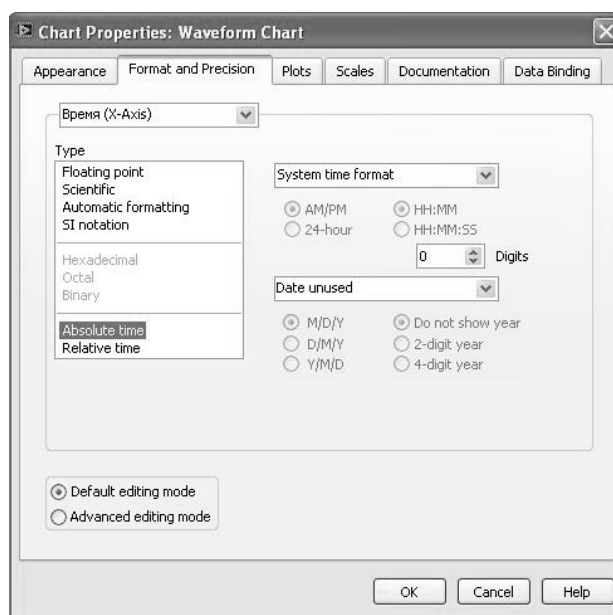


Рис. 7.2 – Вкладка настройки формата и точности единиц осей.

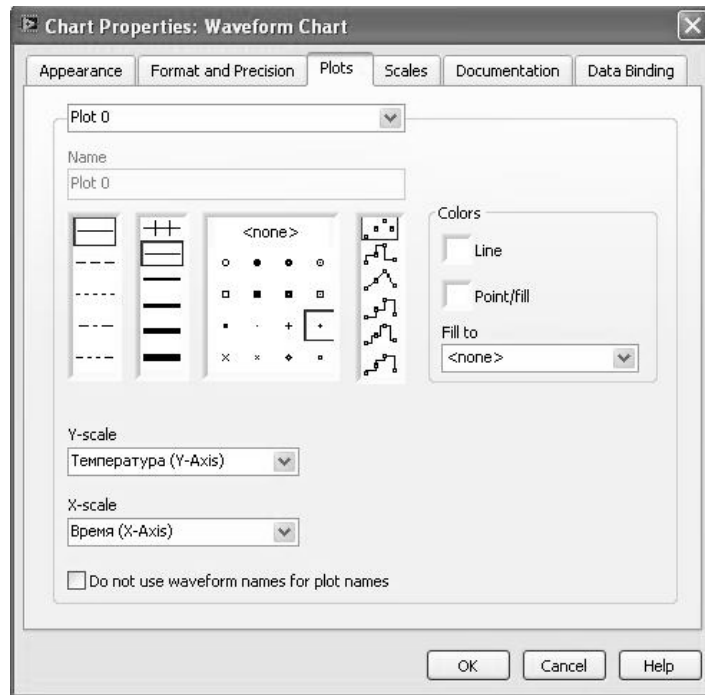


Рис. 7.3 – Вкладка настройки типа отрисовки линии графика.

1.3 Поместите на лицевую панель ВП кнопку «СТОП», останавливающую выполнения измерений; числовой индикатор для отображения температуры в данный момент времени и индикатор в виде градусника.

1.4 Переименуйте оси графика на «Время» (ось X) и «Температура» (Ось Y).

1.5 Измените диапазон отображаемых значений оси температуры на 0–30 °С.

Блок-диаграмма

2 Перейдите на блок-диаграмму, выбрав пункт главного меню **Window»Show Diagram** и создайте блок-схему, изображённую на рис. 7.8.

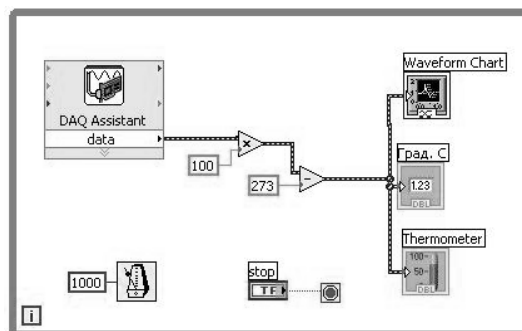


Рис. 7.4 – Блок-диаграмма ВП

2.1 Поместите на бланк блок-диаграммы компонент **DAQ Assistant**. Он находится в палитре **Measurement I/O» DAQmx – Data Acquisition**.

2.2 После помещения компонента появится окно выбора задачи DAQ Assistant (рис. 7.5).

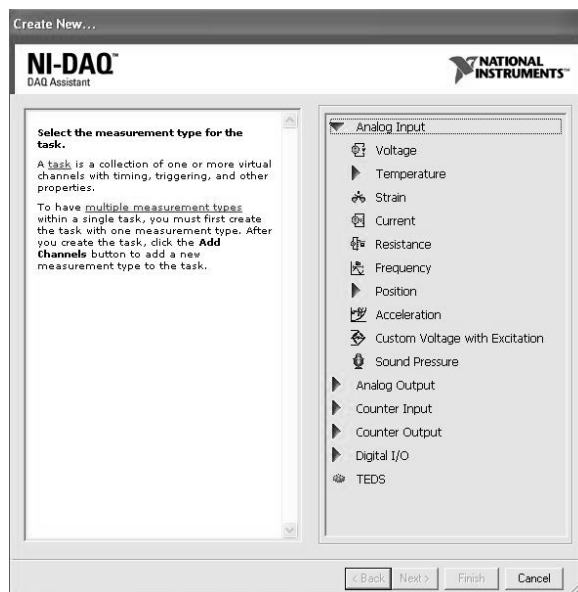


Рис. 7.5 – Окно выбора задачи DAQ Assistant

В появившемся окне раскройте список задач аналогового ввода (**Analog Input**) и выберите задачу измерения напряжения (**Voltage**). Нажмите кнопку **Next**.

2.3 Новое открывшееся окно (рис. 7.6) позволяет выбрать канал измерения. Выберите из списка **ai0** и нажмите кнопку **Finish**.

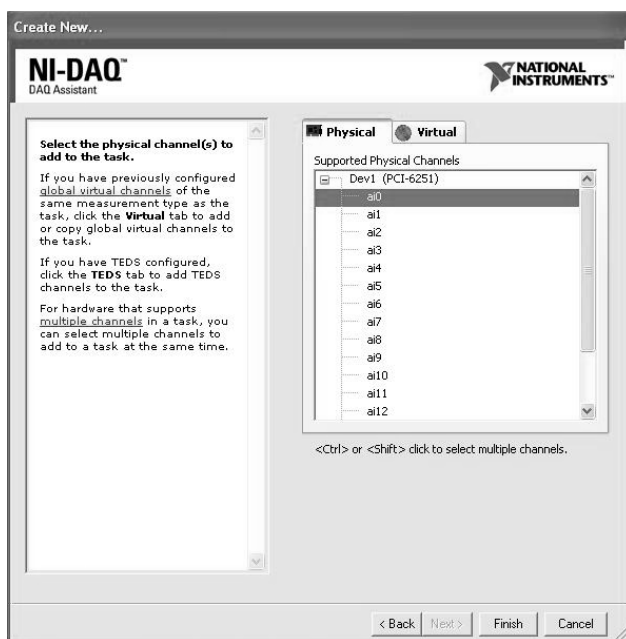


Рис. 7.6 – Окно выбора канала измерения

2.4 Далее открывается окно настройки параметров измерения (рис. 7.7).

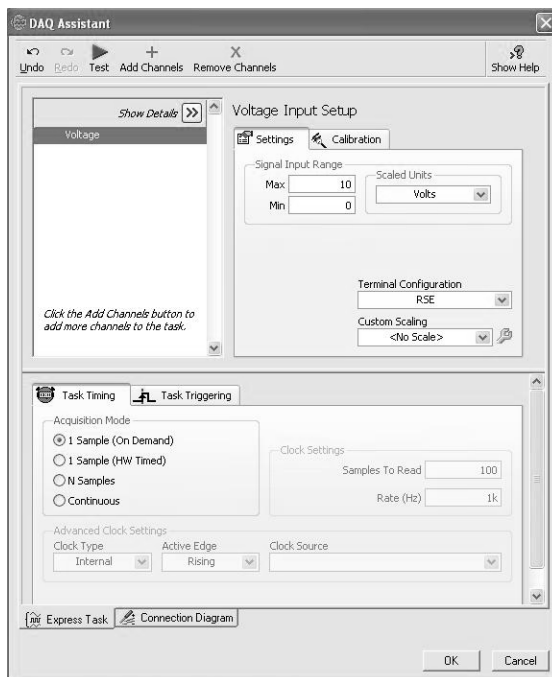


Рис. 7.7 – Окно настройки параметров измерения

Задайте параметры как показано на рис. 7.7.

На вкладке **Settings**:

- диапазон измеряемого напряжения (**Signal Input Range**) от 0 В (**Min**) до 10 В (**Max**);
- конфигурация входных терминалов (**Terminal Configuration**) – измерение с **общим заземлённым проводом (RSE)**.

На вкладке **Task Timing** выберите способ измерения **одна выборка по запросу (1 Sample (On Demand))**.

Нажмите кнопку **Ok**. Настройка параметров измерения закончена.

2.5 Поместите на блок-диаграмму элементы, численно преобразующие измеренное напряжение в температуру по формуле: $T = 100 \cdot U - 273$ (°C).

2.6 Дополнительно разместите **числовой индикатор** и **индикатор в виде термометра**. Соедините параллельно с **Waveform Chart**.

2.7 Для задания интервала измерения поместите ВП **Waite Until Next ms Multiple** из палитры **Time & Dialog**. Создайте константу с значением 1000 мс и соедините со входом ВП.

2.8 Сохраните прибор с именем *Измерение температуры.vi*.

2.9 Запустите прибор и наблюдайте за изменением графика, нагревая датчик температуры пальцами рук.

2.10 Ознакомьтесь с другими настройками **Waveform Chart**.

2.11 Закройте ВП.

Конец упражнения 7.1

Упражнение 7-2. ВП График окружности

Цель: Построить осциллограмму окружности, используя двухкоординатный график осциллограмм.

Выполните следующие шаги для создания ВП, который с помощью двух независимых массивов X и Y построит осциллограмму в форме окружности.

Лицевая панель

1. Откройте новый ВП и создайте лицевую панель, показанную ниже на рис. 7.8.

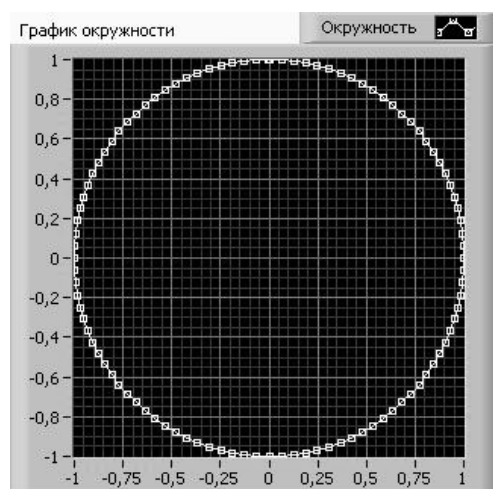


Рис. 7.8 – Лицевая панель ВП

- 1.1 В палитре **Controls»Modern»Graph** выберите **XY Graph**.
- 1.2 Присвойте графику имя **XY график окружности**.
- 1.3 Переименуйте надпись **График 0** на панели **Plot legend** в **Окружность**.
- 1.4 Щелкните правой кнопкой мыши на панели **Plot legend** и выберите из контекстного меню в разделе **Point Style** точки в форме маленьких квадратов.
- 1.5 Переименуйте и измените диапазон шкал, как показано на рисунке.

Блок-диаграмма

2. Постройте блок-диаграмму, как показано ниже.

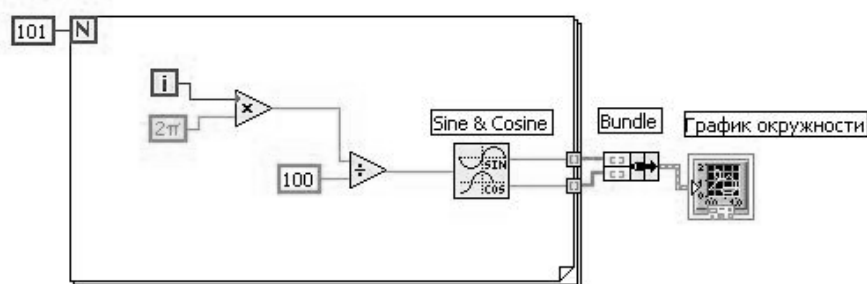


Рис. 7.9 – | Блок-диаграмма ВП

В палитре **Functions»Mathematics»Elementary & Special Function»Trigonometric Functions** выберите функцию **Sine & Cosine**. Эта функция создает массив данных, содержащий один период функций синуса и косинуса.

В палитре **Functions»Programming»Cluster & Variant** выберите функцию **Bundle**. Эта функция объединяет массивы синуса и косинуса в кластер.

В палитре **Functions»Programming»Numeric»Additional Numeric Constants** выберите константу 2π .

3. Сохраните ВП под именем *График окружности.vi*
4. Перейдите на лицевую панель и запустите ВП.
5. Закройте ВП.

Конец упражнения 7-2

Занятие 8. Принятие решений в ВП

Упражнение 8-1. ВП Извлечение квадратного корня

Цель: Изучить структуру Case.

Выполните следующие шаги для построения ВП, который проверяет входное число на знак, вычисляет его квадратный корень или выдает сообщение об ошибке, если число отрицательное.

Лицевая панель

1. Откройте новый ВП и создайте лицевую панель, как показано ниже.



Рис. 8.1 – Лицевая панель ВП

Блок-диаграмма

2. Создайте блок-диаграмму, показанную ниже на рисунке:

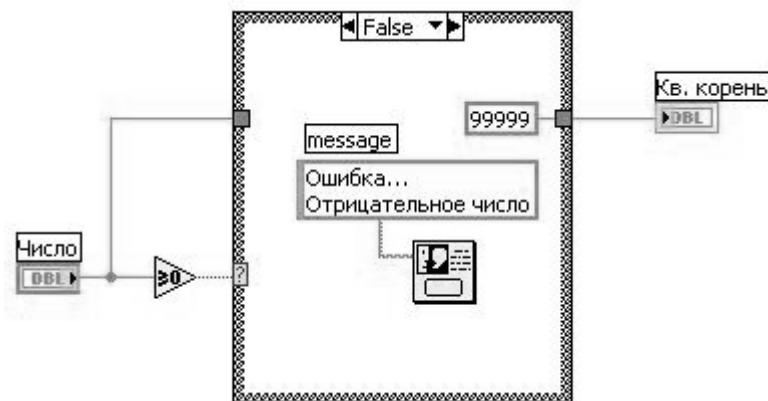


Рис. 8.2 – Блок-диаграмма ВП. Вариант

Поместите на блок-диаграмму структуру **Case**, расположенную в палитре **Functions»Programming»Structures**.

Используйте стрелки уменьшения или увеличения селектора структуры для выбора варианта **FALSE**.

Поместите на блок-диаграмму функцию **Greater or Equal to 0?**, расположенную в палитре **Functions»Programming»Comparison**. Функция возвращает значение **TRUE**, если число больше или равно 0.

Щелкните правой кнопкой мыши по численной константе и в контекстном меню выберите пункт **Format & Precision**. Установите **Digits**

of Precision равным 1, выберите вид представления **Floating Point Notation** и нажмите кнопку ОК.

Поместите на блок-диаграмму **One Button Dialog**, расположенный в палитре **Functions»Time & Dialog**. Это диалоговое окно будет отображать сообщение «**Ошибка... Отрицательное число**».

Щелкните правой кнопкой мыши по полю ввода/вывода message функции **One Button Dialog** и в контекстном меню выберите пункт **Create»Constant**. Введите текст «**Ошибка... Отрицательное число**».

3. Выберите вариант TRUE.

Поместите функцию **Square Root** на блок-диаграмму, как показано ниже. Функция размещена в палитре **Functions»Programming»Numeric**. Она возвращает квадратный корень входного числа.

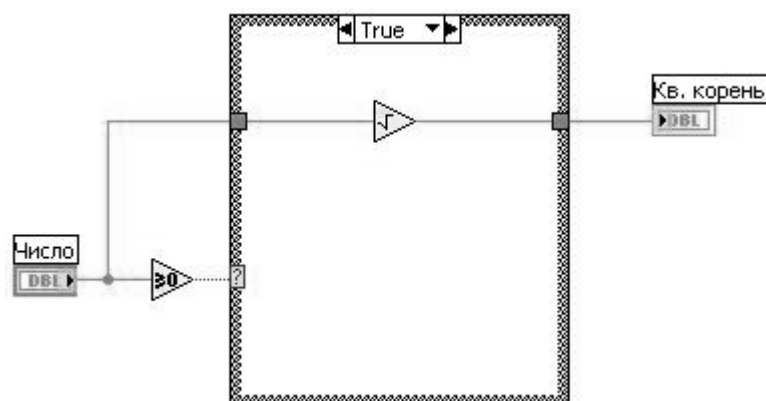


Рис. 8.3 – Блок-диаграмма ВП. Вариант

4. Сохраните созданный ВП под именем *Извлечение квадратного корня, vi*

Запуск ВП

5. Отобразите лицевую панель и запустите ВП.

Внимание Не запускайте ВП кнопкой непрерывного запуска, так как при определенных обстоятельствах запуск этого ВП в непрерывном режиме может привести к бесконечному циклу.

Если входное значение элемента управления Число положительно, то выполнится подпрограмма варианта TRUE и вычисляется значение квадратного корня. Если значение элемента Число является отрицательным, то выполнится подпрограмма варианта FALSE, которая возвращает -99999,0 и отображает диалоговое окно с сообщением «**Ошибка... Отрицательное число**».

6. Закройте ВП.

Конец упражнения 8-1

Упражнение 8-2. ВП Узел Формулы

Цель: Изучить работу структуры узел Формулы.

Выполните следующие пункты для построения ВП, который использует узел Формулы для выполнения нескольких математических операций и выводит результаты в виде графика.

Лицевая панель

1. Откройте новый ВП и постройте лицевую панель, как показано ниже на рисунке.

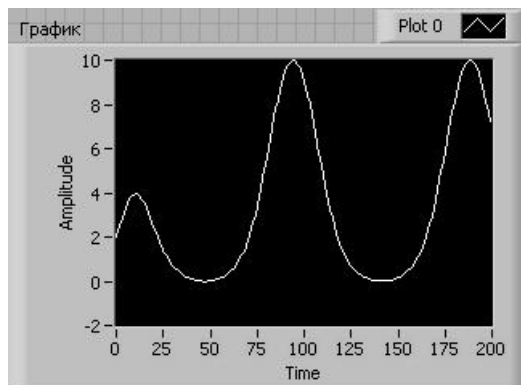


Рис. 8.4 – Лицевая панель ВП.

Блок-диаграмма

2. Постройте блок-диаграмму, показанную на рисунке:

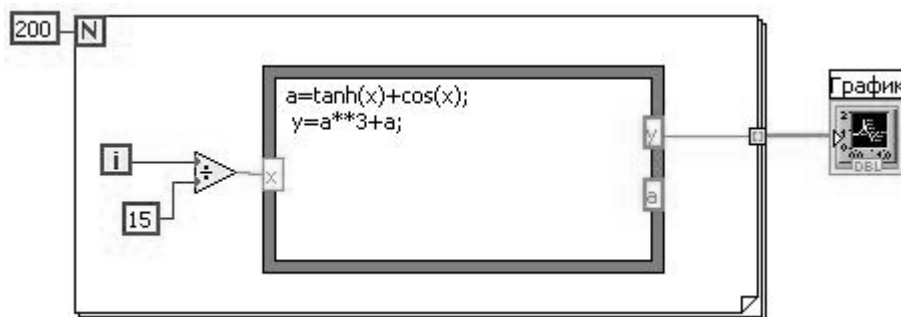


Рис. 8.5 – Блок-диаграмма ВП

Поместите на блок-диаграмму узел Формулы (**Formula Node**), расположенный в палитре **Functions»Programming»Structures**.

3. Создайте входной терминал x , щелкнув правой кнопкой мыши по левой границе структуры и выбрав пункт **Add Input** из контекстного меню. Введите значение x в появившееся окно.

4. Создайте терминалы выходных данных y и a , щелкнув правой кнопкой мыши по левой границе структуры и выбрав пункт **Add Output** из контекстного меню. Допустимо открытие выходного терминала для промежуточных переменных, например a .

Замечание. При открытии входных или выходных терминалов имя переменной должно точно соответствовать имени переменной, используемой в арифметическом выражении. Также необходимо учитывать то, что прописные и заглавные буквы различаются.

5. Введите следующие уравнения в узел Формулы. (** – это оператор степени). Для получения информации о синтаксисе узла Формулы используйте справку Lab VIEW Help.

$$a = \tanh(x) + \cos(x);$$

$$y = a^{**3} + a;$$

6. Сохраните ВП под именем *Узел Формулы.vi*

Запуск ВП

7. Отобразите лицевую панель и запустите ВП. На графике отобразится кривая, соответствующая функции $y = f(x)^3 + f(x)$, где $f(x) = \tanh(x) + \cos(x)$.

В каждой итерации ВП делит значения терминала текущей итерации на 15,0. Частное является аргументом x узла Формулы, по которому рассчитывается значение функции y . Далее ВП строит график массива y .

8. Закройте ВП.

Конец упражнения 8-2

Занятие 9. Строки

Упражнение 9-1. ВП Компоновка строки

Цель: Приобретение навыков по использованию функций обработки строк.

Ниже приведена последовательность действий для создания ВП, который преобразует числовые данные в строку и объединяет строку с другими строками в одну. Затем после поиска по шаблону полученная часть строки переводится в числовой формат.

Лицевая панель

1. Откройте новый ВП и оформите лицевую панель, как показано ниже на рисунке. Воспроизводить комментарии и подписи к элементам не обязательно.

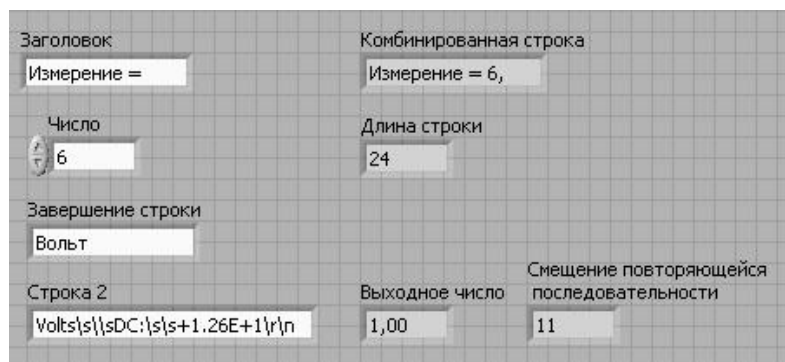


Рис. 9.1 – Лицевая панель ВП

- a. Щелкните правой кнопкой мыши по элементу **Строка 2** и выберите из контекстного меню режим отображения **'\ Codes Display**.
- b. Для элементов **Длина строки** и **Смещение повторяющейся последовательности** установите тип представления данных **I32**

Блок-диаграмма

2. Постройте блок-диаграмму, как показано ниже:

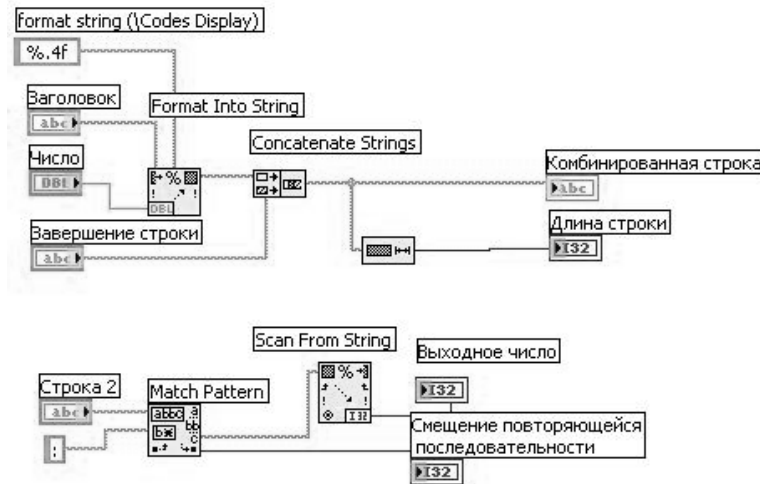


Рис. 9.2 – Блок-диаграмма ВП

Выберите функцию **Format Into String**, расположенную в палитре **Functions»Programming»String**. Эта функция преобразует число в строку.

a. Щелкните правой кнопкой мыши по функции **Format Into String** и выберите пункт **Edit Format String** для вызова соответствующего диалогового окна.

b. Выделите опцию **Use specified precision** и в поле ввода текста введите значение 4 для преобразования элемента **Число** в строку с четырьмя знаками после запятой.

c. Нажмите на кнопку **OK**. Lab VIEW создаст формат строки `%.4f`, используя выбранную опцию.

d. С помощью инструмента **ВВОД ТЕКСТА**, введите пробел с обеих сторон `%.4f` и нажмите клавиши `<Shift+Enter>`. Таким образом, на элементе **Комбинированная строка** числовые данные появятся с пробелами с обеих сторон.

e. Щелкните правой кнопкой мыши по константе и выберите режим отображения **'V Codes Display** из контекстного меню. Введенные пробелы заменятся на `\`.

Выберите функцию **Concatenate Strings**, расположенную в палитре **Functions»String**. Эта функция объединит входящие в нее строки в одну.

Выберите функцию **String Length**, расположенную в палитре **Functions»String**. Эта функция выдаст значение количества символов в объединенной строке **Комбинированная строка**.

Выберите функцию **Match Pattern**, расположенную в палитре **Functions»Programming»String**. Эта функция осуществляет поиск в элементе **Строка 2** по шаблону `:` (двоеточие).

f. Щелкните правой кнопкой мыши по полю **regular expression** и

выберите пункт контекстного меню **Create»Constant**, введите двоеточие и нажмите на клавиши **<Shift+Enter>**.

Выберите функцию **Scan from String**, расположенную в палитре **Functions»Programming»String**. Эта функция преобразует строку после двоеточия в числовые данные.

Иконка ВП и соединительная панель

3. Перейдите на лицевую панель и создайте иконку и соединительную панель для использования созданного ВП в качестве подпрограммы в других ВП. Подробная информация о методах оформления иконки и соединительной панели находится в Уроке 3, *Подпрограммы ВП*.

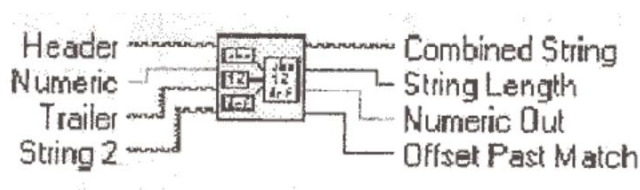


Рис. 9.3 – Иконка ВП

4. Сохраните ВП под именем *Компоновка строки, vi*. Этот ВП будет использоваться позднее.

Запуск ВП

5. Измените значение элементов на лицевой панели и запустите ВП. ВП объединит элементы: **Заголовок**, **Число** и **Завершение строки** в строку **Комбинированная строка** и выдаст значение длины строки.

ВП также найдет месторасположение подстроки: в элементе **Строка 2**. При выполнении ВП преобразует строку после двоеточия в число Выходное число и выводит на экран индекс первого элемента после двоеточия в элемент **Смещение повторяющейся последовательности**.

6. Сохраните и закройте ВП.

Конец упражнения 9-1

Занятие 10. Функции файлового ввода/вывода

Упражнение 10-1. ВП Запись файла.

Цель: Записать данные в файл.

Ниже приведена последовательность действий для создания ВП, который объединяет строку, числовые данные и модуль строки в файл. В упражнении 10-2 будет создан ВП, который считывает и отображает его содержимое.

Лицевая панель

1. Откройте новый ВП и оформите лицевую панель, как показано ниже:

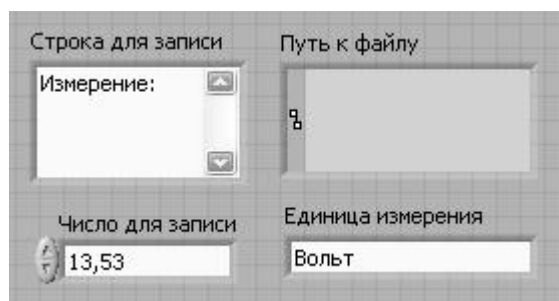


Рис. 10.1 – Лицевая панель ВП

- В палитре **Controls»Modern»String & Path** выберите элемент отображения пути. Этот элемент отобразит путь к созданному файлу данных.
- Щелкните правой кнопкой мыши по элементу **Строка для записи** и в контекстном меню выберите пункт **Visible Items»Scrollbar**.

Блок-диаграмма

2. Постройте следующую блок-диаграмму:

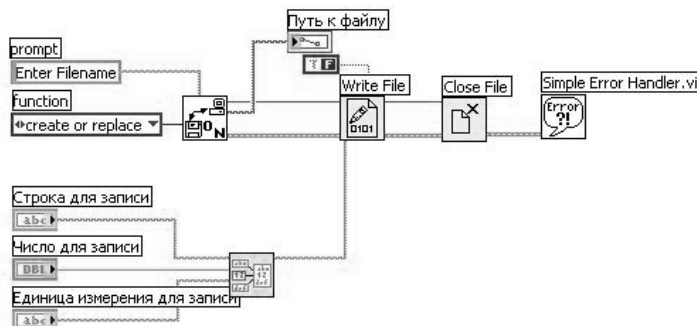


Рис. 10.2 –

Блок-диаграмма ВП

Выберите в разделе **Functions»Select a VI** ВП *Компоновка строки.vi*, созданный в упражнении 9-1, и поместите его на блок-диаграмму. Этот ВП объединяет три строки в одну.



Поместите на блок-диаграмму подпрограмму ВП **Open/Create/Replace File VI**, расположенную в палитре **Functions»Programming»File I/O**. Этот ВП выводит на экран диалоговое окно для создания файла.

a. Щелкните правой кнопкой мыши по полю **prompt** и в контекстном меню выберите пункт **Create»Constant** для создания константы. Введите имя файла. При запуске ВП появится окно выбора файла, которое будет называться **Введите имя файла**.

b. Щелкните правой кнопкой мыши по входному полю **function** и в контекстном меню выберите пункт **Create»Constant**. Для выбора пункта выпадающего меню **create or replace** следует использовать инструмент УПРАВЛЕНИЕ,



Выберите функцию **Write File**, расположенную в палитре **Functions»Programming»File I/O**. Эта функция записывает объединенную строку в файл.



Выберите функцию **Close File**, расположенную в палитре **Functions»Programming»File I/O**. Эта функция закрывает файл.



Выберите подпрограмму ВП **Simple Error Handler VI**, расположенную в палитре **Functions»Programming»Time & Dialog**. Этот ВП проверяет кластер ошибок и выводит диалоговое окно при возникновении ошибки.

3. Сохраните ВП под именем *Запись файла, vi*.

Запуск ВП

4. Поменяйте значения элементов управления на лицевой панели и запустите ВП. Появится диалоговое окно Введите имя файла.
5. Введите в диалоговое окно название файла *демофайл.txt* и нажмите на кнопку Save или ОК.
ВП запишет в файл данные из элементов **Строка для записи**, **Число для записи** и **Единица измерения для записи**.
6. Закройте ВП.

Конец упражнения 10-1

Упражнение 10-2 ВП Чтение файла

Цель: Считать данные из файла.

Ниже приведена последовательность действий для создания ВП, который читает файл, созданный в упражнении 9-2, и выводит данные в строковом элементе отображения.

Лицевая панель

7. Откройте новый ВП и создайте лицевую панель, используя элемент управления путем к файлу и строковый элемент отображения в палитре **Controls»String & Path**.

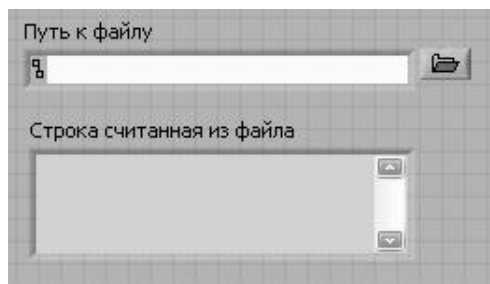


Рис. 10.3 –

Лицевая панель ВП

Блок-диаграмма

8. Постройте следующую блок-диаграмму:

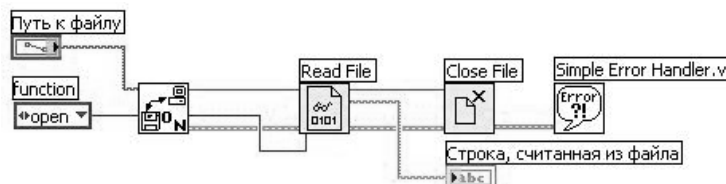


Рис. 10.4 –

Блок-диаграмма ВП

В палитре **Functions»Programming»File I/O** выберите подпрограмму **Open/Create/Replace File VI**. Этот ВП выведет на экран диалоговое окно, которое используется для создания и открытия файла.



- Щелкните правой кнопкой мыши по входному полю **prompt** и из контекстного меню выберите **Create»Constant** для создания константы **Выбрать имя файла**.
- Щелкните правой кнопкой мыши по полю **function** и выберите в контекстном меню пункт **Create»Constant** для создания константы. С помощью инструмента УПРАВЛЕНИЕ выберите пункт выпадающего меню **open**.



3 палитре **Functions»Programming»File I/O** выберите функцию **Read File**. Эта функция читает количество байт данных с начала файла, определяемое значением поля **count**.



В палитре **Functions»Programming»File I/O** выберите функцию **Close File**. Эта функция закрывает файл.



В палитре **Functions»Programming»Time& Dialog** выберите подпрограмму **Simple Error Handler VI**. Этот ВП проверяет кластер ошибок и, в случае появления ошибки, выводит на экран диалоговое окно.

9. Сохраните ВП под именем *Чтение файла.vi*,

Запуск ВП

10.Перейдите на лицевую панель и с помощью инструмента УПРАВЛЕНИЕ выберите кнопку **Browse** (обзор) в элементе управления Путь к файлу.

11.Выберите файл *демофайл.vi* и нажмите на кнопку Open или ОК.

12.Запустите ВП. Строка, считанная из файла, отобразится на лицевой панели ВП.

13. Закройте ВП.

Конец упражнения 10-2

Упражнение 10-3. ВП Запись таблицы символов

Цель: Сохранить 2D массив в текстовый файл в виде таблицы так, чтобы он был доступен редактору электронной таблицы.

В упражнении 9-4 строка была отформатирована таким образом, что позиция табулятора разделял столбцы, а символ конца строки разделял строки. Выполните следующие шаги для изучения ВП, который сохраняет числовой массив в файл в формате, доступном для редактора электронных таблиц.

Лицевая панель

1. Создайте лицевую панель ВП.

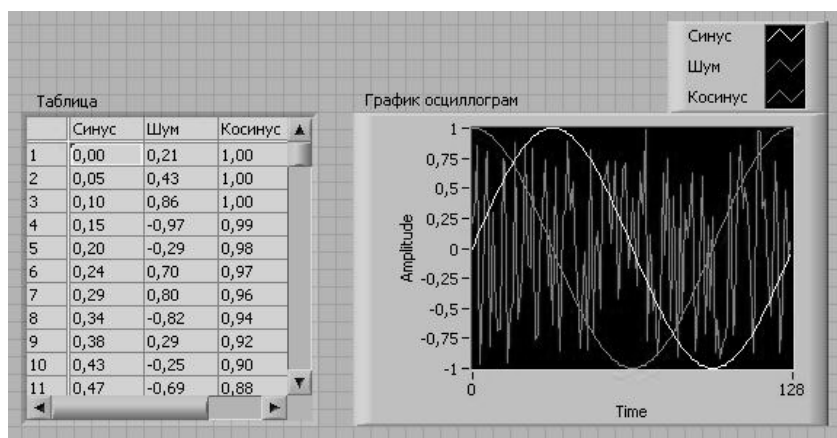
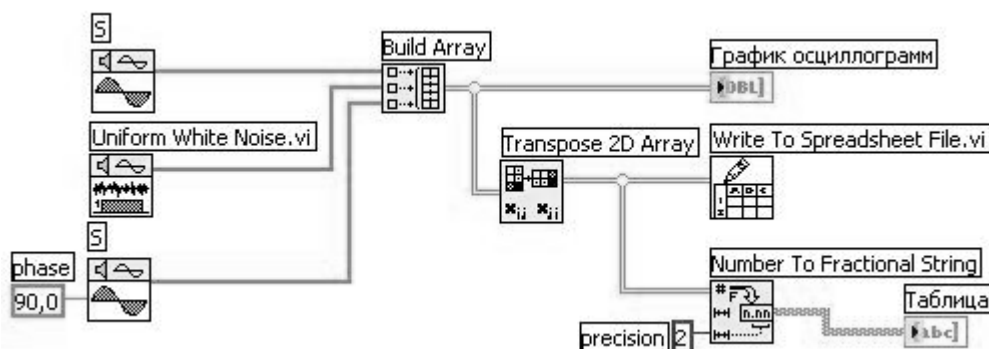


Рис. 10.5 –

Лицевая панель ВП

Блок-диаграмма

2. Создайте блок-диаграмму:



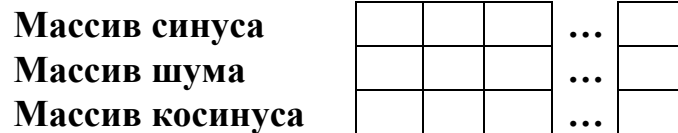
- ВП **Sine Wave Pattern VI**, расположенный в палитре **Functions»Signal Processing»Signal Generation** выдает числовой массив из 128 элементов, содержащий значения синусоидального сигнала. Константа 90,0 определяет фазу гармонического сигнала.



- ВП **Uniform White Noise VI**, расположенный в палитре **Functions»Signal Processing»Signal Generation** выдает числовой массив из 128 элементов, содержащий **значения** шумового сигнала.



- Функция **Build Array**, расположенная в палитре **Functions»Programming»Array**, компонует 2D массив из массивов синуса, шума и косинуса.



- Функция **Transpose 2D Array**, расположенная в палитре **Functions»Programming»Array**, перестраивает элементы 2D массива так, что элемент $[i,j]$ становится элементом $[j,i]$, как показано ниже:

S	N	C



- ВП **Write To Spreadsheet File VI**, расположенный в палитре **Functions» Programming»File I/O**, форматирует **2D массив в строку** таблицы символов и записывает строку в файл. Формат строки показан на рисунке, где стрелка \rightarrow показывает позицию табулятора, а символ параграфа $\¶$ показывает окончание **строки** символов.



- Функция **Number To Fractional String**, расположенная в палитре **Functions»Programming»String»String/Number Conversion**, преобразует массив чисел в массив строк, которые отображаются в таблице.

Запуск ВП

3. Запустите ВП. ВП генерирует 2D массив из 128 строк и 3 столбцов. Первый столбец содержит данные осциллограммы синуса, второй – осциллограммы шума, третий столбец – данные осциллограммы косинуса. ВП выводит осциллограммы каждого столбца и отображает

данные в виде таблицы. После вывода данных на экран появляется диалоговое окно.

4. Введите имя файла *осциллограмма.vi* и нажмите на кнопку Save или ОК. Позднее этот файл будет использоваться.

5. Закройте ВП.

Примечание В этом примере сохраняются в файл только три массива. Для добавления большего количества массивов следует увеличить количество полей ввода данных функции **Build Array**.

Дополнительная часть

Откройте файл текстовым редактором или редактором электронных таблиц и просмотрите его содержимое.

6. Откройте текстовый редактор.

7. Откройте файл *осциллограммам*. Данные синуса представлены в первом столбце, шум – во втором, косинус – в третьем.

8. Выйдите из текстового редактора или редактора электронных таблиц и вернитесь в LabVIEW.

Конец упражнения 10-3

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Виноградова, Н. А. Разработка прикладного программного обеспечения в среде LabVIEW: Учебное пособие / Н. А. Виноградова Я. И. Листратов, Е. В. Свиридов. – М. : Издательство МЭИ, 2005. – 49 с.
2. Тревис, Дж. LabView для всех / Джеффри Тревис : Пер. с англ. Н. А. Клушин – М. : ДМК Пресс; ПриборКомплект, 2004. – 544 с.

ОСНОВЫ ПРОГРАММИРОВАНИЯ В LABVIEW 8

Учебное издание

Методические указания
к проведению лабораторного практикума
по курсу «Проектирование информационно-измерительных систем»

Составители: **Н. Н. БЕСПАЛОВ, М. В. ИЛЬИН**

*Печатается в соответствии с предоставленным
оригинал-макетом*

Сдано в набор __.12.2006. Подписано в печать __.12.2006.

Гарнитура Таймс. Печать офсетная. Формат 60x84 1/16.

Уч.-изд. л. 0,00 Усл. печ. л. _____. Тираж 200 экз.

Заказ

Мордовский госуниверситет
430000 г. Саранск, ул. Большевистская, 68