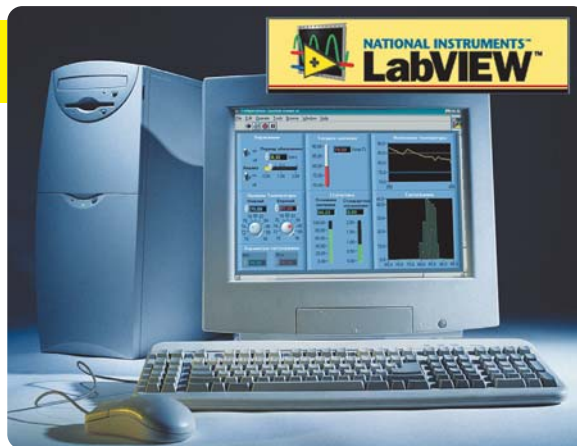


# Учебный практикум по LabVIEW



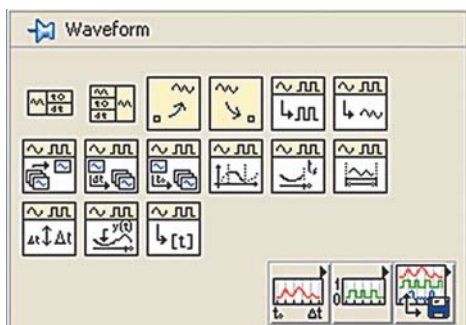
Работая в LabVIEW с различными временными сигналами, очень удобно использовать специальный тип данных Waveform (осциллограмма). Если осциллограмма представлена в виде дискретных отсчетов сигнала, она называется аналоговой (Analog Waveform), если же она является последовательностью цифрового кода, то это – цифровая осциллограмма (Digital Waveform). Принципы графического представления Waveform частично уже рассматривались. Теперь же остановимся на нем детальнее. Итак, тип данных Waveform, основные концепции, создание, функции для работы, генераторы...



В целом, структура **Waveform** подобна структуре кластеров. Она состоит из следующих данных: времени **t0**, соответствующему первому отсчету сигнала (т.е. времени начала осциллограммы), периода дискретизации **dt** сигнала, и непосредственно массива отсчетов **Y** осциллограммы, имеющих тип данных с плавающей запятой двойной точности.

Осциллограмма	
t0	Y
00:00:00 DD.MM.YYYY	-1,402622 -0,067795
dt	-0,160014
1,00m	-0,805194 -0,53765

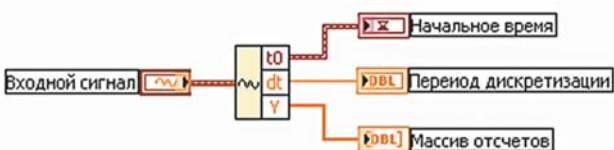
Она основная палитра функций работы с Waveform выглядит так:



Специальные рисунки и , присутствующие на пиктограмме конкретной функции, указывают с каким типом осциллограммы (аналоговой или цифровой)

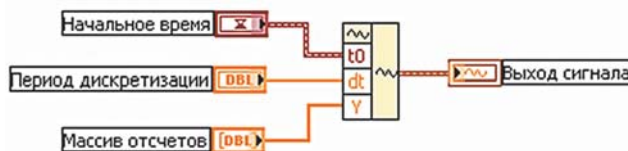
может работать данная функция. Преимущество типа данных Waveform по сравнению с массивами (Array) заключается в задании сигнала в реальном масштабе времени, тогда как Array содержит лишь последовательность отсчетов без какой-либо другой информации о сигнале.

Следует заметить, что если пользователю по каким-то причинам все же удобнее работать с сигналом, заданным непосредственно в виде массива, то осуществить преобразование из Waveform в Array не составляет никакого труда. Для этого необходимо воспользоваться функцией **Get Waveform Components** (получить компоненты осциллограммы):

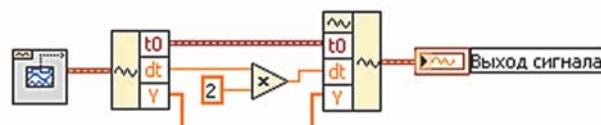


Как видно, данная функция фактически "разбивает" осциллограмму на составляющие ее компоненты. Для реализации обратной процедуры, т.е. "конструирования" осциллограммы, следует использовать функцию **Build Waveform** (создать осциллограмму):

циллограммы, следует использовать функцию **Build Waveform** (создать осциллограмму):



Рассмотрим необходимость использования этих функций на простом примере. Предположим, необходимо вдвое увеличить период дискретизации осциллограммы, т.е. "растянуть" сигнал во времени. Это позволяет сделать следующий фрагмент блок-диаграммы:

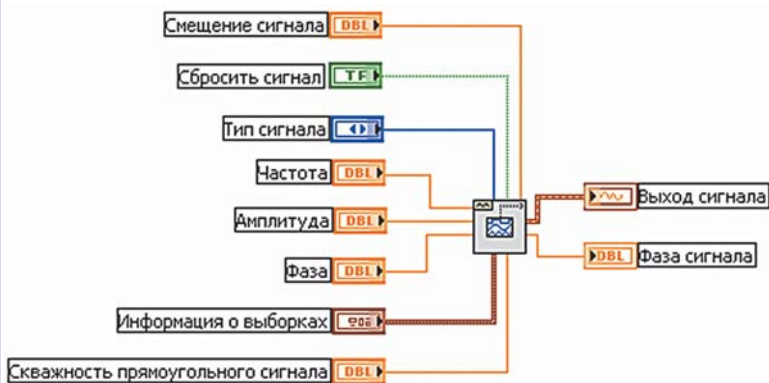


Аналогичный прием можно использовать, если есть необходимость масштабировать сигнал по амплитуде.

**Замечание.** Простое умножение Waveform с константой либо переменной эквивалентно именно масштабированию по амплитуде.

В целом, подпалитры функций LabVIEW для анализа и генерации осциллограмм подобны соответствующим подпалитрам **Signal Processing** и **Signal Generation**.

Давайте-ка рассмотрим возможности генерации виртуальных сигналов в виде осциллограмм в LabVIEW. Для примера возьмем функцию **Basic Function Generator** (Основной генератор функций):



Назначение входов **Смещение сигнала**, **Сброс сигнала**, **Тип сигнала**, **Частота**, **Амплитуда**, **Фаза**, **Сквозность прямоугольного сигнала** следует из их названий, и аналогично входным параметрам функций из палитры Signal Generation. Эти функции были рассмотрены в выпуске ПИКАД №4, 2007г.

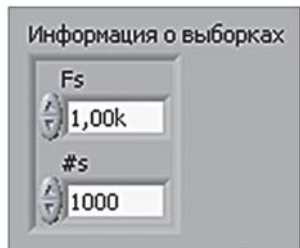
При помощи регулятора **Тип сигнала** можно выбрать форму генерируемого сигнала:

- синусоидальный;
- треугольный;
- прямоугольный;
- пилообразный.

На выходе **Фаза сигнала** возвращается действительное значение его фазы. Значения входов по умолчанию равны:

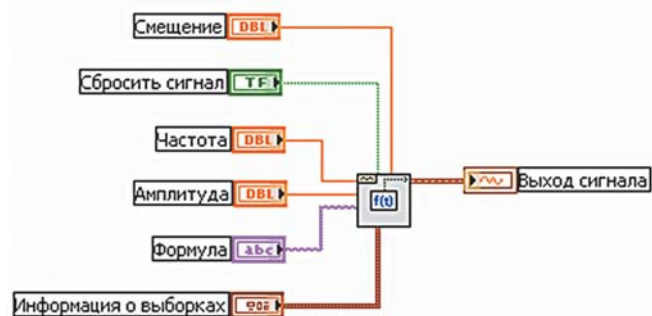
Смещение сигнала - 0;  
 Сброс сигнала - False;  
 Тип сигнала - синусоидальный;  
 Частота - 10 Гц;  
 Амплитуда - 1;  
 Фаза - 0;  
 Сквозность прямоугольного сигнала - 50 %.

**Вход Информация о выборках** задает параметры, которыми, собственно, осциллограмма отличается от других типов данных. Тип данных на этом входе является кластером, состоящим из двух элементов - частоты выборки, выраженной в Гц, и их число. Данные значения по умолчанию равны 1000 Гц и 1000 выборок соответственно.



**Важно!** При задании информации о выборках необходимо помнить о теореме Котельникова (или критерии Найквиста, если угодно). При его нарушении, т.е. установке частоты выборки меньше удвоенной частоты самого сигнала запуск программы на выполнение приведет к ошибке. Так что какие либо эксперименты в этом направлении лучше сразу исключить.

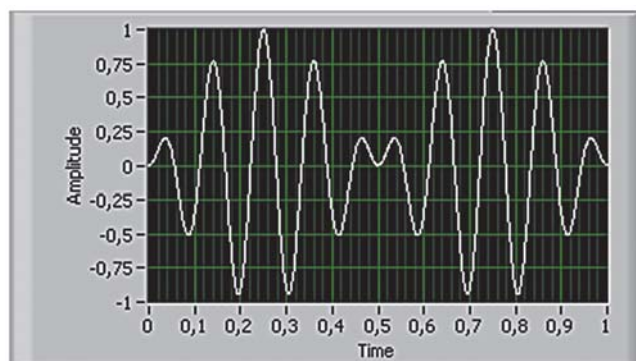
Весьма интересной является функция **Formula Waveform** (расчет осциллограммы по формуле), позволяющая задавать осциллограмму требуемой формы аналитически:



Назначение большинства входов данной функции понятно исходя из описанного выше. Значения по умолчанию у них такие же. Ключевым же является вход **Формула**. На него необходимо подать соответствующую аналитическую формулу, созданную с помощью переменной либо константы строкового типа. При этом следует учитывать, что синтаксис формулы совпадает с синтаксисом узла **Formula Node**, а множество аргументов ограничивается лишь следующими входными параметрами:

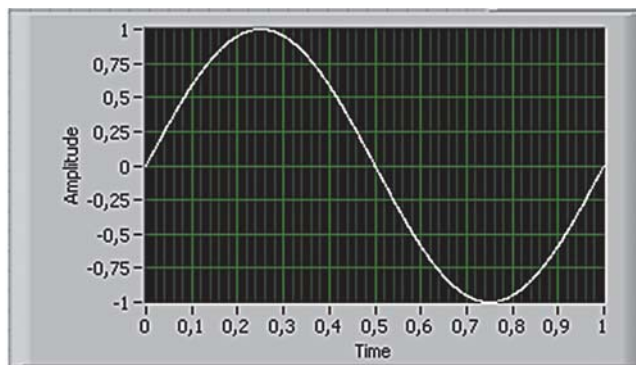
- значение, поданное на вход **Частота**, задается переменной **f**;
- значение, поданное на вход **Амплитуда**, задается переменной **a**;
- значение круговой частоты **w**, вычисляемое с использованием переменной **f**:  $w=2\pi*f$ ;
- имеется также возможность использования в формуле абсциссы создаваемой осциллограммы (времени отсчета **t** номера **n**);
- установленная частота дискретизации сигнала содержится в переменной **fs**.

Значение, устанавливаемое по умолчанию, и подаваемое на вход **Формула**  $\sin(w*t)*\sin(2*\pi(1)*t)$ . Т.о., функция **Formula Waveform** по умолчанию генерирует осциллограмму амплитудно-модулированной синусоиды:



Настоятельно рекомендуем придерживаться синтаксиса при вводе формулы, иначе сообщения об ошибке Вам не избежать!

Также следите за вводимым регистром символов. В частности, для получения осциллограммы синусоиды в поле **Формула** необходимо ввести, например, выражение  $\sin(\pi(2)*t)$ , при этом вы увидите такую осциллограмму:



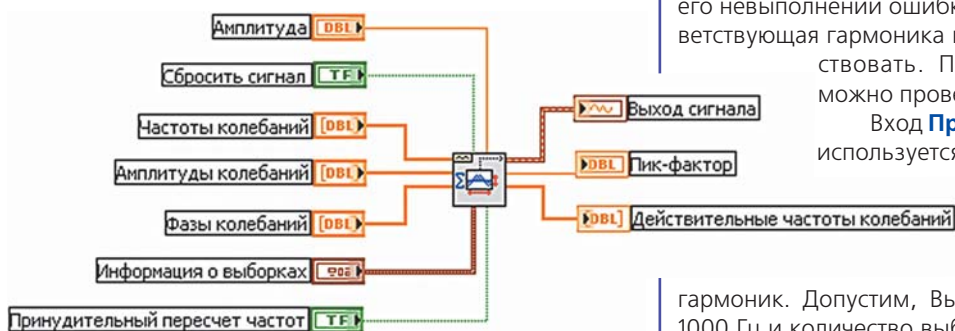
Очевидно, изменение значения аргумента синуса вызовет соответствующие перемены параметров синусоиды, например, частоты. При вводе же выражения прописными буквами (т.е. "SIN(PI(2)\*t)") вы увидите сообщение о некорректности ввода.

Для усвоения изложенного материала создайте и убедитесь в правильности создания следующих осциллограмм:

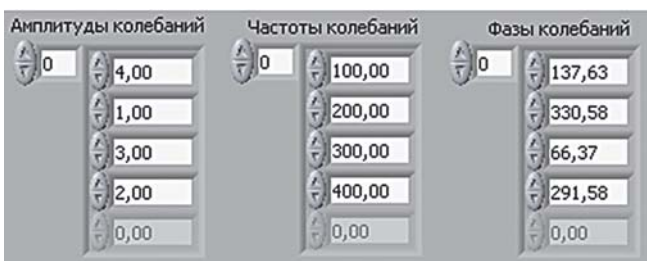
- модели напряжения промышленной частоты. Напомним, что это напряжение является полигармоническим, а частота первой гармоники составляет 50 Гц. Значения начальных фаз можно выбрать произвольными. Количество гармоник, которыми можно ограничиться, выберите на свое усмотрение;
- функции отсчетов ( $\sin x/x$ ).

Если Вы успешно проделали все заданное, можем перейти к рассмотрению других, не менее интересных генераторов осциллограмм. Для моделирования полигармонического сигнала очень удобны **Multitone Generator** (многотональный генератор), **Basic Multitone Generator** (основной многотональный генератор), и **Basic Multitone with Amplitudes** (основной многотональный генератор с заданными амплитудами).

Для использования многотонального генератора необходимо предварительно сформировать входные массивы информативных параметров - амплитуд, частот, фаз:



Задание информативных параметров гармоник на лицевой панели будет выглядеть примерно следующим образом:



Размеры массивов **Частоты колебаний**, **Амплитуды колебаний** и **Фазы колебаний** обязательно должны совпадать! Фактически размер этих массивов - это количество генерируемых гармоник. Задание отрицательных значений в массиве **Частоты колебаний** приводит к ошибке.

Содержимое массива **Фазы** по умолчанию - случайные значения, выраженные в градусах, так что если Вы желаете согласовать гармоники по фазе, устанавливать данные значения необходимо вручную.

Значение, подаваемое на вход **Амплитуда** - это требуемое значение амплитуды созданной осциллограммы, т.е. суммы всех заданных гармоник. Необходимость его задания можно пояснить простым примером. Предположим, виртуальная осциллограмма предназначена для последующего преобразования в реальный сигнал, например, электрическое напряжение. В данном случае необходимо учесть физические ограничения - выходной диапазон напряжений ЦАП, как известно, не бесконечен. Виртуальную же осциллограмму, наоборот, можно создать без всяких ограничений по величине сигнала. Так вот, при подаче на вход **Амплитуда** какого-либо значения, например, 5, созданная осциллограмма нормируется таким образом, чтобы ее амплитуда равнялась этому значению, т.е. равнялась бы 5, независимо от заданных амплитуд отдельных гармоник. Как Вы наверняка догадались, значение 5 предполагается использовать, имея в "арсенале"

ЦАП с выходным диапазоном  $\pm 5$  В. Установка на входе **Амплитуда** отрицательного значения предполагает игнорирование описанной выше нормализации. По умолчанию на входе **Амплитуда** установлено значение -1.

На досуге подумайте самостоятельно, каким образом можно нормализовать осциллограмму, если бы вход **Амплитуда** отсутствовал. Несомненно, это поможет Вам в будущем при столкновении с подобными вопросами. Назначение входа **Информация о выборках** аналогично рассмотренному входу для функции **Formula Waveform**. Обязательно запомните, что следить за выполнением критерия Найквиста в данном случае придется Вам! При его невыполнении ошибка выдана не будет, однако соответствующая гармоника в выходном сигнале будет отсутствовать. Подумайте, каким образом это можно проверить.

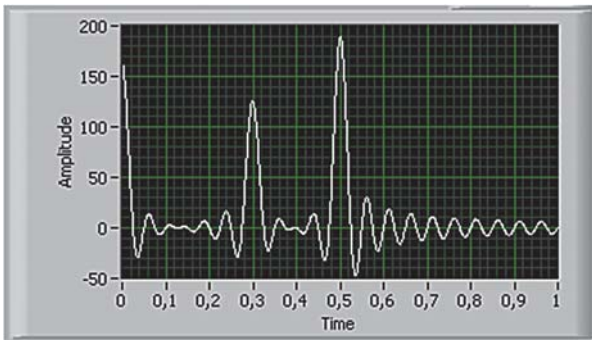
Вход **Принудительный пересчет частот** используется для следующих целей. Дело в том, что в выходной осциллограмме может быть сгенерировано лишь целое число периодов любой из заданных гармоник.

Допустим, Вы задали частоту дискретизации 1000 Гц и количество выборок - 1000 (фактически это означает, что длительность осциллограммы равна 1 с). Это означает, что Вы можете задать частоту гармоник равной 1 Гц, 2 Гц и т.д. Именно гармоники с такими частотами "впишутся" в осциллограмму целое количество раз. Если же Вы зададите одно из значений в массиве **Частоты колебаний** равным, например, 1,2 Гц, то запуск программы на выполнение будет прерван сообщением об ошибке. Если же Вы не можете непосредственно контролировать условие кратности частот (например, если частоты гармоник рассчитываются предварительно по специальному алгоритму), установите на вход **Принудительный пересчет частот** значение **TRUE**, и программа сама округлит установленные частоты гармоник до ближайших "рабочих" значений. Иначе - ошибка. При этом контроль действительных значений частот можно осуществлять с помощью выходного массива **Действительные частоты гармоник**.

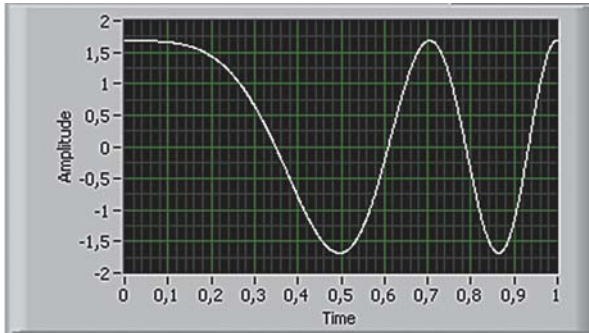
Вход **Сбросить сигнал** предназначен для следующего. Если Вы запускаете свой VI на длительное выполнение либо помещаете "многотональный генератор" в структуру "Цикл", то по умолчанию Вы можете менять параметры на входе генератора, при этом сразу же будет происходить соответствующее изменение осциллограммы. Убедитесь в этом самостоятельно.

А теперь подайте на вход **Сбросить сигнал** значение **TRUE**. Заметили, что осциллограмма перестала меняться при изменении входных параметров? Т.о., в этом случае генератор перестает "реагировать" на изменение входов. По умолчанию значение на входе **Сбросить сигнал** - **FALSE**.

Выход **Пик-фактор** содержит отношение амплитудного значения осциллограммы к среднеквадратическому. Эти данные являются справочными. Их учет необходим, например, если осциллограмму необходимо впоследствии преобразовать в реальный электрический сигнал, а диапазон пик-факторов используемых реальных устройств является ограниченным. Примеры сигналов с различными значениями пик-фактора:



Сигнал с высоким значением пик-фактора

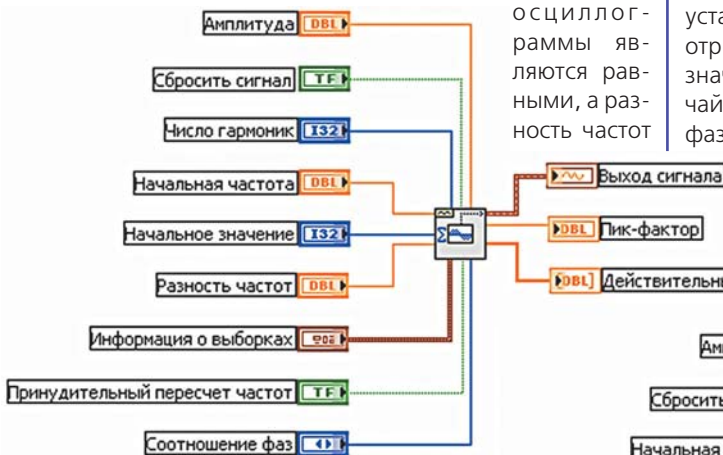


Сигнал с низким значением пик-фактора

**Примечание.** Значение на выходе **Пик-фактор** полностью корректно лишь при отсутствии нормализации осциллограммы (т.е. отрицательного значения на входе **Амплитуда**). Если же присутствует процедура приведения осциллограммы к необходимому диапазону, то выход **Пик-фактор** возвращает отношение амплитуды нормализованного сигнала к с.к.з не нормализованного (именно так!).

Рассмотрим теперь работу **Basic Multitone Generator**. Он является более упрощенным блоком в сравнении с предыдущим. Так, амплитуды гармоник генерируемой им осциллограммы являются равными, а раз-

ность частот



между двумя соседними гармониками - постоянной.

Вход **Начальная частота** предназначен для задания частоты основной гармоники колебания в Гц. Его значение по умолчанию равно 10 Гц. Установка отрицательного значения приводит к ошибке, а о принудительном пересчете частот будет сказано ниже.

Вход **Число гармоник**, как ясно из его названия, содержит целочисленное

значение количества гармоник (включая основную) генерируемой осциллограммы. Его значение по умолчанию равно 3.

На входе **Разность частот** задается "расстояние" по частоте между соседними гармониками. Так, если Вы, предположим, установили **Начальную частоту** равной 37 Гц, а **Разность частот** - 10 Гц, то частоты гармоник (при их количестве 3) будут равны 37, 47, 57 Гц соответственно. Продумайте, с помощью каких функций LabVIEW этот факт можно проверить наглядно.

Следует заметить, что на входе **Разность частот** можно устанавливать и отрицательные значения. Тогда основная гармоника будет находиться справа от дополнительных на сетке частот, а порядок следования гармоник будет обратным (например, 37, 27, 17 Гц при установке разности частот -10 Гц). При этом во избежание ошибки программы необходимо лишь следить, чтобы ни одна из частот гармоник не оказалась отрицательной. Так, при задании пяти гармоник в последнем примере рассчитанная частота пятой гармоники будет равна -7 Гц, что, конечно, бессмысленно.

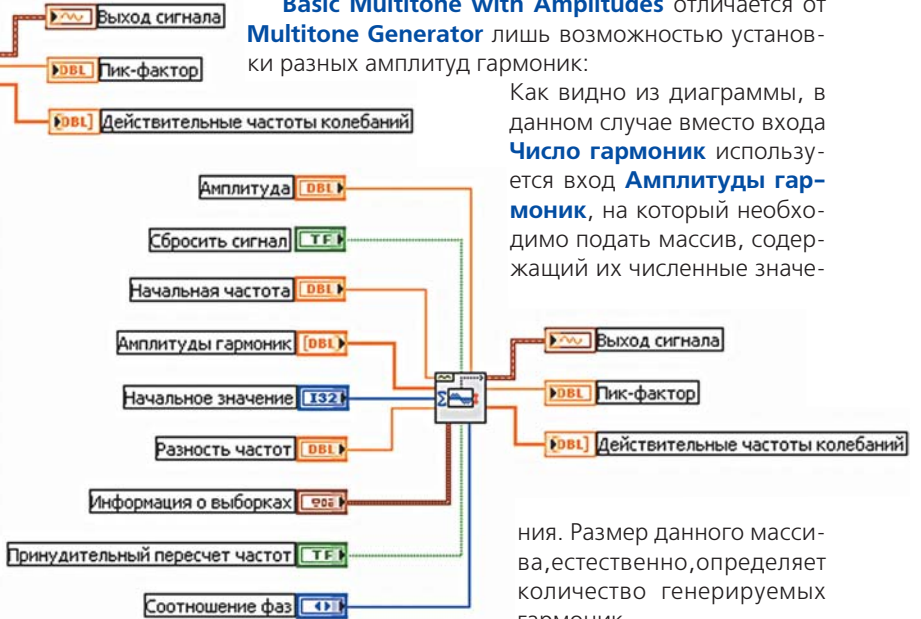
Назначение входов **Амплитуда**, **Информация о выборках**, **Принудительный пересчет частот**, **Сбросить сигнал** и выход **Действительные частоты** гармоник аналогичны одноименным в **Multitone Generator**, рассмотренном выше. Амплитуда гармоник осциллограммы в случае установки отрицательного значения на вход **Амплитуда** равна 1. В противном случае, т.е. при проведении нормализации, происходит пересчет амплитуд по описанному выше принципу.

С помощью входа **Соотношение фаз** можно выбрать, как должны распределиться фазы гармоник генерируемой осциллограммы. Очевидно, что при выборе случайного распределения фаза каждой гармоники выбирается случайным образом из диапазона 0..360°. Использовать же линейное соотношение рекомендуется для уменьшения пик-фактора.

Вход **Начальное значение** опрашивается лишь при установке случайного соотношения фаз. При установке отрицательного либо нулевого значения на этом входе значения фаз каждый раз генерируются по-новому и случайным образом. В противном случае генерация значений фаз происходит лишь один раз.

**Basic Multitone with Amplitudes** отличается от **Multitone Generator** лишь возможностью установки разных амплитуд гармоник:

Как видно из диаграммы, в данном случае вместо входа **Число гармоник** используется вход **Амплитуды гармоник**, на который необходимо подать массив, содержащий их численные значения.



Выход сигнала

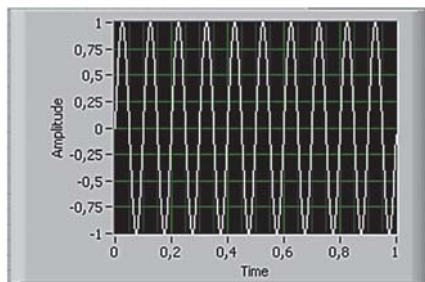
Пик-фактор

Действительные частоты колебаний

Размер данного массива, естественно, определяет количество генерируемых гармоник.

**Опробовав все три вида многотональных генераторов, подумайте, с помощью какого из них моделирование напряжения промышленной частоты является наиболее простым и удобным.**

После изучения вопросов генерации осциллограмм остановимся на некоторых приемах и особенностях графического представления осциллограмм. Соедините выход "основного генератора функций" со входом графического индикатора Waveform Graph на блок-диаграмме. Запустите программу на выполнение, не изменяя никаких дополнительных настроек. В результате вы получите нечто похожее на следующую картинку:



А теперь, предположим, что Вам необходимо выводить осциллограмму не на весь экран индикатора, а оставить свободные поля по бокам, сверху и снизу. Для этого выполните следующие действия:

- щелкните правой кнопкой "мыши" на области графического экрана, из выпадающего меню выберите пункт **X Scale**. В появившемся подменю уберите флажок напротив **Autoscale X**. Тем самым Вы отменяете режим автоматического масштабирования горизонтальной оси графика, т.е. самоподстройки ее границ под входной сигнал;

- проделайте аналогичную процедуру для вертикальной оси. В первом выпадающем меню выберите пункт **Y Scale**. В появившемся подменю уберите флажок напротив **Autoscale Y**;

- теперь необходимо вручную задать границы осей. Для этого выполните следующее. Перейдите на блок-диаграмму. Нажмите правой кнопкой "мыши" на иконку, соответствующую графическому индикатору. Далее последовательно выберите следующие пункты выпадающих подменю: **Create** » **Property Node** » **X Scale Range** » **Maximum**.

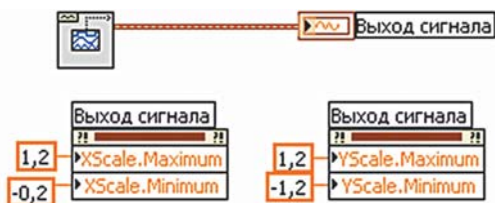


После этого на блок-диаграмме появится вот такая иконка ▲  
Потяните за нижний ее край, при этом появится снизу строка **XScale.Maximum**. Как Вы наверняка заметили, значения минимума и максимума оси **X** можно лишь считать, поскольку сигнальные выходы находятся справа иконки;

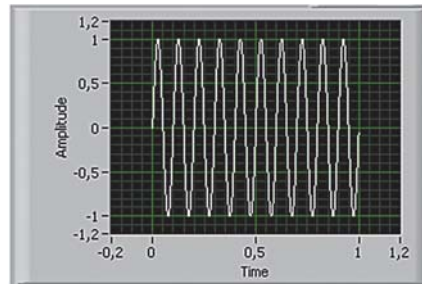
- нажмите правой кнопки мыши на иконку, выберите меню пункт **Change All To Write**. Теперь значения границ горизонтальной оси можно задавать "извне";

- аналогичную процедуру проделайте для задания границ вертикальной оси. Для этого необходимо выбрать следующие подменю **Create** » **Property Node** » **Y Scale Range** » **Maximum**. Дальнейшие действия являются аналогичными.

- наконец, для полей на экране задайте следующие границы осей:



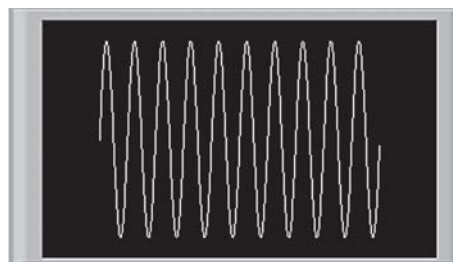
Полученная в результате выполнения программы осциллограмма, таким образом, будет иметь следующий вид:



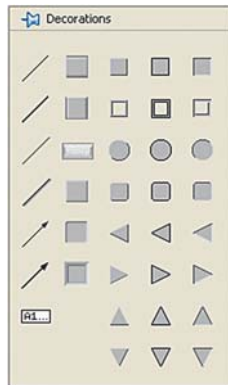
Как видим, она выглядит намного привлекательнее исходной - сигнал не "упирается" в края экрана, и поэтому лучше воспринимается.

Как Вы наверняка успели догадаться, значения границ осей можно задавать не постоянными, а при необходимости регулировать в процессе выполнения программы. Их также можно просчитывать путем специальных алгоритмов. С помощью таких приемов можно моделировать режимы работы реальных осциллографов - визуальный сдвиг графика сигнала вправо-влево и вверх-вниз, его "растяжение" и "сжатие" на экране. Возможности, которые открываются перед Вами в этом направлении - безграничны. Для еще большего сходства с реальными графическими экранами приборов сделайте следующее. Для начала "уберите" название осей и шкалы. Это делается для того, чтобы исключить возможность их динамического изменения в процессе работы программы. Более того, в аналоговых осциллографах они просто отсутствуют. Для этого нажмите правой кнопкой мыши на область экрана осциллограммы и в появившемся меню выберите пункт **Visible items**. В подменю уберите флажки напротив **X Scale**, **YScale**. Результатом этих действий должно стать "исчезновение" осей.

Теперь необходим последний "штрих" - убрать линии сетки экрана. Для этого в упомянутом только что меню выберите **Properties**. При этом перед Вами появится дополнительное окно свойств **Waveform Graph**. Выберите в нем закладку **Scales**, и в ее области **Grid Style and Color** выберите черный цвет для **Major grid** и **Minor grid**. Итак, при дальнейшем запуске программы экран осциллограммы еще больше изменился.



Требуемую сетку Вы можете "нарисовать" вручную, используя компоненты палитры **Decorations**.



Потренируйтесь менять цвета выбранных линий (для их большего подобия на линии сетки, например, реального аналогового осциллографа). Выбор цвета осуществляется на панели инструментов **Tools**. Самостоятельно разобраться в этом не составит труда.

*Материал подготовлен студентами старших курсов Национального технического Университета Украины "Киевский политехнический институт" факультета авиакосмических систем.*