

МИНОБНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»

Разработка и исследование цифрового генератора DDS на базе ПЛИС

Электронный лабораторный практикум

САМАРА

2012

Составители: **Корнилин Дмитрий Владимирович,**
Кудрявцев Илья Александрович

Разработка и исследование цифрового генератора DDS на базе ПЛИС [Электронный ресурс] : электрон. лаб. практикум / Минобрнауки России, Самар. гос. аэрокосм. ун-т им. С. П. Королева (нац. исслед. ун-т); сост. Д. В. Корнилин, И. А. Кудрявцев. - Электрон. текстовые и граф. дан. (0,27 Мбайт). - Самара, 2012. - 1 эл. опт. диск (CD-ROM).

В лабораторном практикуме рассматриваются вопросы по разработке цифрового генератора DDS на базе ПЛИС и исследованию его основных характеристик. Лабораторный практикум обеспечивает подготовку по дисциплине «Методы цифровой обработки сигналов в радиотехнических системах» образовательных программ подготовки магистров по специальности 210400.68 "Радиотехника", реализуемых на радиотехническом факультете. Курс 5, 6, семестр А, В.

Лабораторный практикум разработан на кафедре радиотехнических устройств.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 ОСНОВЫ DDS	4
2 СОЗДАНИЕ ПРОЕКТА	6
2.1 Моделирование схемы	7
3 ЗАДАЧИ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ	9
3.1 Задача повышенной сложности	9

ВВЕДЕНИЕ

В настоящее время FPGA нашли широкое распространение благодаря своей гибкости и функциональности, а также и другим функциям, необходимым инженерам. Для эффективного использования FPGA, разработчику требуется владения HDL и знание некоторых специфических приемов.

Одной из самых известных компаний, производящих FPGA, является XILINX, которая предлагает множество различных чипов для различных целей. В нашей лабораторной работе мы будем иметь дело с семейством SPARTAN, хотя те же исследования могут быть выполнены с использованием других FPGA.

FPGA особенно популярны в том случае, когда разработчик должен иметь в своей схеме мощный процессор и несколько сигнальных каналов. В этих случаях может помочь программная реализация CPU / DSP непосредственно в ПЛИС или компании могут предоставлять свои собственные процессоры программного обеспечения и интегрированных процессорных ядер (FPGA типа VIRTEX). Типичным примером является навигационный приемник, который имеет несколько каналов одновременной обработки одного и того же сигнала RF.

Одной из важных частей навигационных приемников и других аналогичных устройств, таких как Software Defined Radio и т.д., является «генератор с цифровым управлением»(NCO), который должен обеспечивать стабильную частоту гармонического сигнала. Один из самых популярных способов решить эту проблему сегодня является использование DDS техники, которые могут быть реализованы в FPGA.

Целью данного тренинга является ознакомление с деталями реализации DDS на основе NCO в FPGA.

1 Основы DDS

Идея очень проста - мы будем синтезировать гармонический сигнал, используя так называемую «Look-Up-Table», в который, как в ROM, сохраним выборки синусоидального сигнала, как показано на рис.1.

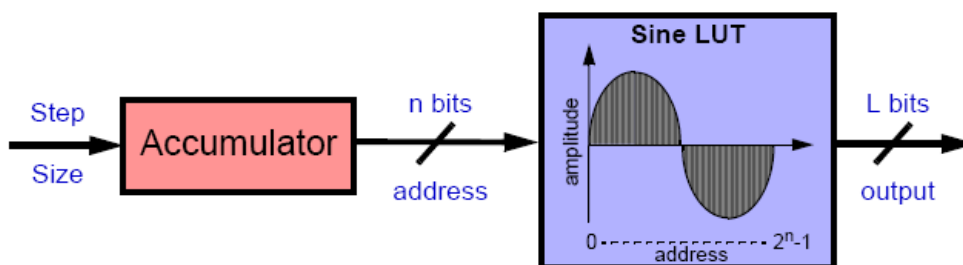


Рисунок 1 - DDS основы

Если необходимо изменить частоту, можно либо изменить тактовую частоту, или размер шага. Второй способ является более практичным и широко используется. Эта идея показана на рис.2.

Размер шага получен из следующего уравнения:

$$\mu = N \frac{f_D}{f_S}, \text{ где}$$

N - Количество записей LUT (2^n);

f_D - требуемая частота;

f_s - Частота дискретизации системы.

Однако, это уравнение иногда не обеспечивает удовлетворительной точности, потому что нельзя использовать нецелых μ в схеме, показанной на рис.1. Проблема заключается в том, как использовать дробную часть μ ? Эта проблема решается с помощью схемы, показанной на рис. 3.

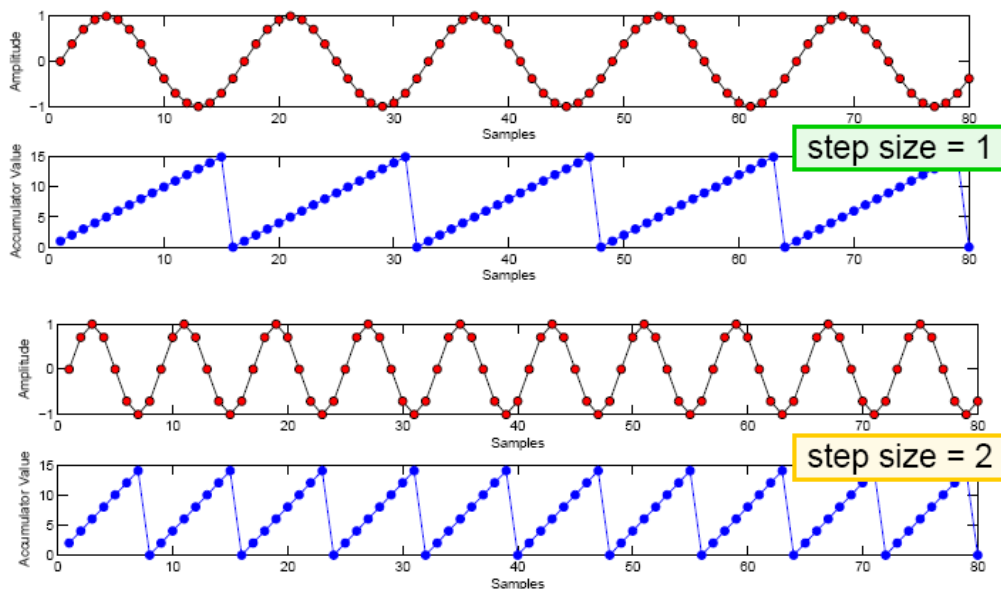


Рисунок 2 - Изменения частоты с шагом

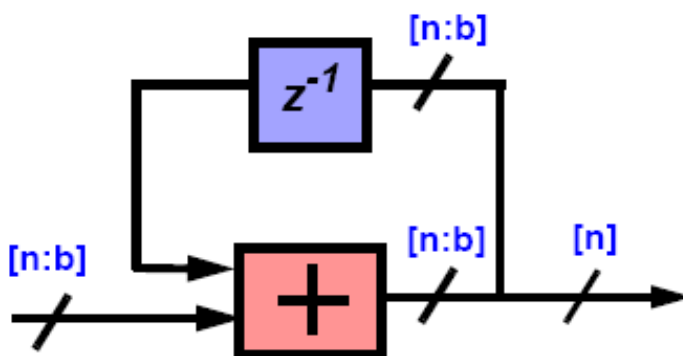


Рисунок 3 - аккумулятор в деталях

Здесь мы будем иметь не целое шаг с b дробных бит. Аккумулятор используется целое значение шага, в то время как только n битов используются для LUT адреса. Теперь точность возрастает с увеличением числа дробных бит. Можно показать, что частотное разрешение может быть описано следующим уравнением:

$$\Delta f = \frac{f_s}{2^b N}$$

Конечно, этот подход имеет недостаток, поскольку в результате происходит усечение фазы и, следовательно, деградация спектральной чистоты сигнала.

Вообще говоря, если необходимо получить высокую спектральную чистоту, нужно больше все биты, которые также требуют большего ТМП. Чтобы решить эту проблему, как правило, разработчики используют различные методы: амплитудный

дизеринг (с добавлением некоторого шума, сглаживание спектральных пиков), фазовый дизеринг (с добавлением некоторого шума на выходе аккумулятора) и фильтрация выходного сигнала, чтобы подавить нежелательные компоненты частоты.

2 Создание проекта

Запустите Xilinx ISE Design Suite, нажав на пиктограмму на рабочем столе. Обычно программа открывает окно наиболее часто используемого проекта. Закройте его (File \ Close Project) и выберите создание нового проекта (File \ New Project).

В следующем окне, вставьте название проекта (путь не должен содержать кириллицу), а также выберите "Schematic" в качестве топ-уровня типа источника и нажмите кнопку "Next".

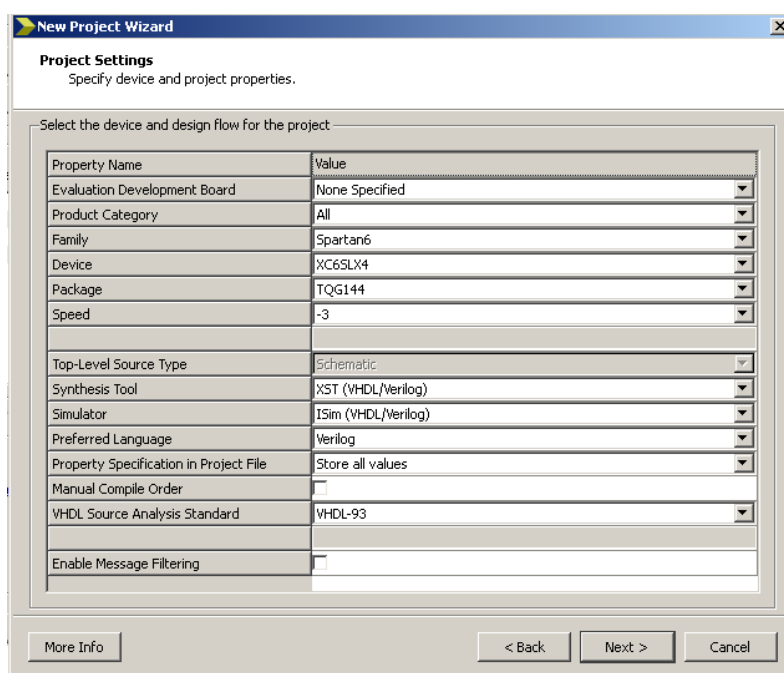


Рисунок 4 – Окно Project Wizard

В следующем окне вы можете выбрать тип FPGA и некоторые другие важные параметры. Для нашего обучения вы можете оставить все возможности «по умолчанию». Следующее окно не требует никаких сложных действий, кроме нажатия кнопки "Finish".

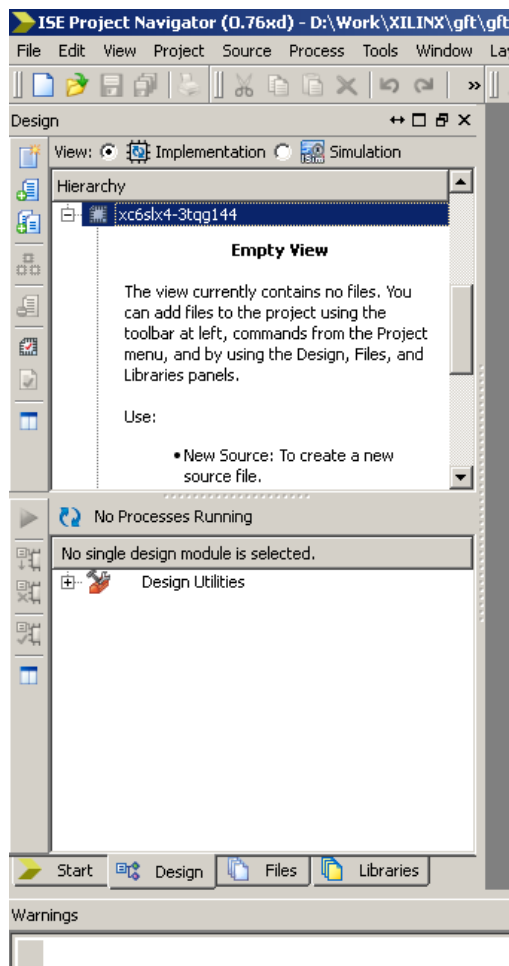


Рисунок 5 – Окно проекта

В левой части рабочего окна вы можете увидеть полезные панели, с помощью которых мы можем управлять компонентами проекта. (Рис. 5)

Здесь мы можем добавить к нашему проекту рабочие файлы. Чтобы сэкономить время, используем готовые исходные файлы: Main.sch, Accumulator.vhd и Sin_lut.vhd. Вы должны скопировать также файлы .sum файлы в каталог проекта. Main.sch это схема NCO, состоящий из двух компонентов: аккумулятора и LUT. Компоненты написаны на языке VHDL и может быть отредактирован как отдельные объекты, если это необходимо. Аккумулятор имеет 32-разрядную структуру и LUT имеет 256 записей.

Вы должны добавить в файл схемы проекта. Чтобы сделать это, нажмите на название проекта правой кнопкой мыши и выберите «Add Source» из контекстного меню. Добавить Test.v в проект в том же порядке.

2.1 Моделирование схемы

Перед моделированием необходимо проверить схему.

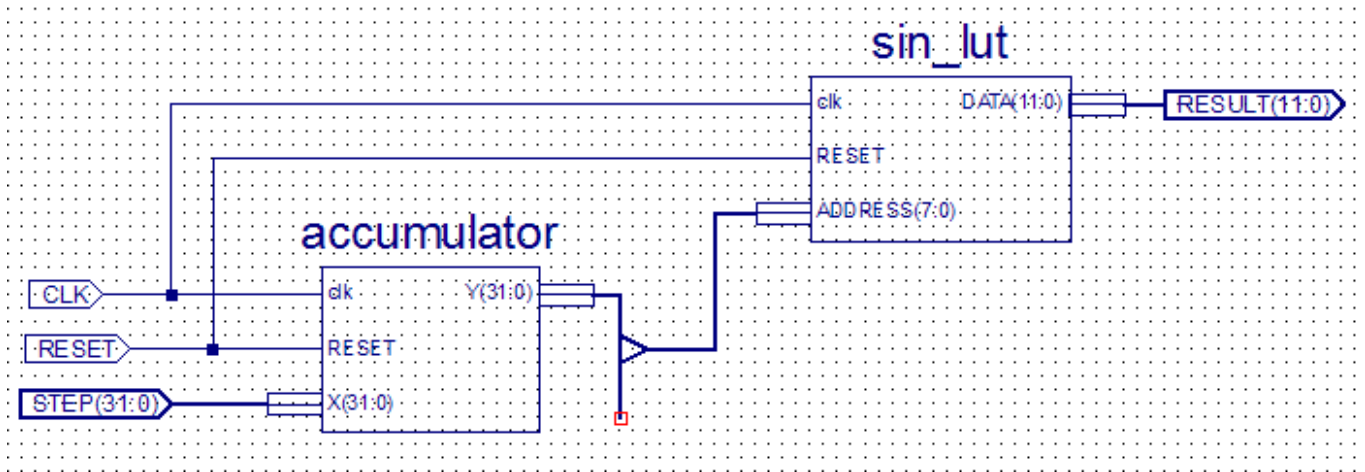


Рисунок 6 - схема NCO

Для имитации NCO необходимо сбросить внутренние схемы компонентов, установите значение шага, а затем примените тактового сигнала. Все эти операции задаются в файле Test.v, созданный в качестве описания Verilog. Для создания этого файла, выберите "New source" в том же меню, а затем "Verilog Test fixture". Введите имя файла, выберите ассоциации с основным и нажмите "Next". Вы увидите сценарий, где мы можем добавить необходимые тестовые сигналы.

Добавьте следующие строки после "// Bidirs":

```
integer hFile;
integer X;
```

А также добавить следующую инструкцию после "//Initialize inputs", заменив все по умолчанию строки ниже, установленного с помощью мастера.

```
initial begin
    hFile=$fopen("res.txt");
    #1 RESET = 1;
    STEP = 150000000;
    #1 CLK = 0;
    #1 CLK = 1;
    #1 CLK = 0;
    #1 RESET = 0;
    repeat(5000) #1
        begin
            CLK=~CLK;
            X=RESULT;
            if (CLK==0)
                if (X<2048) $fwrite(hFile,"%0t\t%d\n",$time,X);
                else $fwrite(hFile,"%0t\t%d\n",$time,X-4096);
            end
        $fclose(hFile);
    end
endmodule
```


Чтобы начать моделирование выбрать “Simulation” В дизайне запись, выберите моделирования файл в окне проекта и в процессе окне ниже начать “Simulate Behavioral Model” процесс. Вы увидите новое окно ISIM с сигналом формы. Кроме этих сигналов, res.txt текстовый файл будет создан. Вы можете загрузить этот файл в программе Excel, чтобы увидеть форму синусоидальной формы.

3 Задачи для самостоятельной работы

1. Используя Excel или любой другой инструмент, вычислить частоту полученного сигнала;
2. Сравните результаты ваших расчетов со значением, определяемым входным шагом;
3. Измените шаг входного сигнала, исследуйте, как оно влияет на частоту и форму сигнала.

3.1 Задача повышенной сложности

Измените значение *width* в accumulator.vhd, создайте новый символ схемы и исследуйте его роль в функционировании NCO.