

Отчёт по проектированию ядра цифрового вычислительного синтезатора синусоидальных сигналов (DDS, Direct Digital Synthesizer)

Определение задачи

Задача: спроектировать цифровой вычислительный синтезатор дискретизированных синусоидальных сигналов (в англоязычных источниках обозначается как DDS или Direct Digital Synthesizer). Результатом проектирования должно быть описание на языке System Verilog, пригодное, в том числе, для использования при разработке интегральных микросхем класса ASIC/ASSP. Основным требованием для проектируемого устройства является обеспечение для спроектированного DDS высокого динамического диапазона, свободного от паразитных составляющих (в англоязычных источниках такой диапазон обозначается как SFDR или Spurious Free Dynamic Range). В качестве технологического ориентира рекомендовано взять зарекомендовавшие себя аналогичные IP-ядра DDS фирмы Xilinx для FPGA (с закрытым исходным описанием) – Xilinx DDS v5.0, Xilinx DDS Compiler 4.0 и Xilinx DDS Compiler 5.0. Дополнительным требованием является наличие в проектируемом DDS выхода как для синуса ($-\sin$), так и для косинуса (\cos).

Схема DDS с усечением значения аккумулятора фазового числа

Основным способом вычислительной генерации синусоидального дискретизированного сигнала является генерация отсчётов с использованием таблицы предрасчитанных отсчётов синусоид, хранящихся в памяти (ROM). На рис. 1 показана наиболее простая схема такого генератора – схема DDS с усечением значения аккумулятора фазового числа.

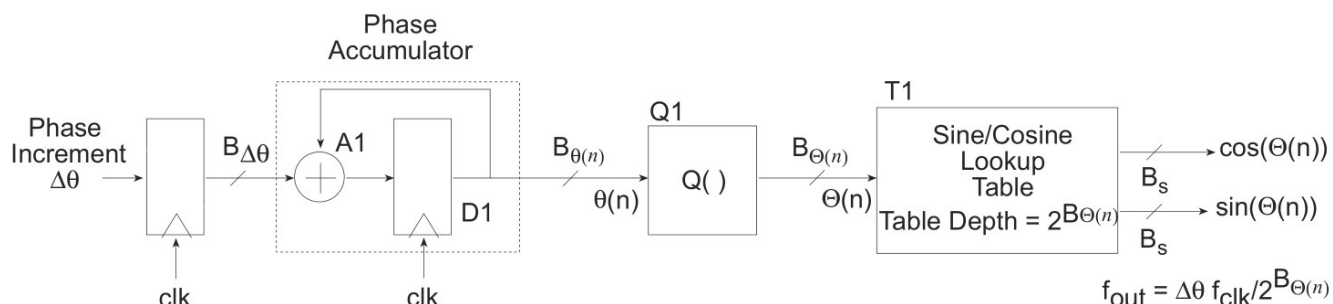


Рис. 1 – Схема DDS с усечением значения фазового аккумулятора

Схема DDS генерирует очередной отсчёт синусоиды в каждом такте синхронизации (после начальной задержки конвейера), таким образом сигнал синхронизации clk задаёт также и частоту дискретизации выходного сигнала. Аккумулятор фазы D1 накапливает текущее число фазы $\theta(n)$ разрядности $B_{\theta(n)}$, а фазовый инкремент $\Delta\theta$ задаёт скорость возрастания фазового числа $\theta(n)$. Важно отметить, что фазовое число $\theta(n)$ не является значением фазы синусоиды, а лишь указывает на долю от полной фазы 2π : текущая фаза $\varphi = 2\pi * \theta(n) / 2^{B_{\theta(n)}}$. Квантователь Q1 берёт от числа $\theta(n)$ только старшие разряды $\Theta(n)$, отбрасывая (усекая, виртуально зануляя) младшие разряды $\theta(n)$. Таким образом $\Theta(n)$ является аппроксимированным значением текущего фазового числа и одновременно является адресом разрядности $B_{\Theta(n)}$ для таблицы T1 значений отсчётов синусоид для полного периода. Другими словами, таблица T1 осуществляет

преобразование усечённого фазового числа во временные отсчёты синусоид. Усечение фазового числа позволяет существенно сократить число записей таблиц (а, значит, и размер ROM), но это действие уменьшает и точность выходного сигнала. Так, для любых значений в некоторой окрестности $\theta(n)$, для которой $Q(\theta(n)) = \Theta(n)$, схема выдаст одно и то же значение отсчёта синусоиды, так как всем значениям такой окрестности $\theta(n)$ будет соответствовать одно и то же число-адрес $\Theta(n)$. Таким образом, точность выходного сигнала для схемы DDS с усечением значения фазового аккумулятора будет ограничиваться как конечной разрядностью B_s записей временных отсчётов таблицы T1, так и дискретностью фазового числа $\Theta(n)$.

Важно отметить, что для таблицы синуса и косинуса полного периода неотъемлемым свойством является её четвертичная симметрия: имея только подтаблицу первой четверти полного периода синуса или косинуса можно вычислить (отразить без потери точности) остальные три четверти отсчётов полного периода. Соответственно, это свойство позволяет без потери точности сократить объем записей ROM на три четверти за счёт несложной логики отражения значений.

Частоту выходного сигнала синуса f_{out} и косинуса можно выразить как

$$f_{out} = \frac{f_{clk} \Delta\theta}{2^{B_{\theta(n)}}}.$$

Тогда $\Delta\theta$, задающий выходную частоту f_{out} можно вычислить как

$$\Delta\theta = \frac{f_{out} 2^{B_{\theta(n)}}}{f_{clk}}.$$

Частотное разрешение Δf можно выразить как

$$\Delta f = \frac{f_{clk}}{2^{B_{\theta(n)}}}.$$

Зная требуемое частотное разрешение Δf можно найти разрядность $B_{\theta(n)}$:

$$B_{\theta(n)} = \left\lceil \log_2 \left(\frac{f_{clk}}{\Delta f} \right) \right\rceil.$$

В ходе разработки DDS-ядра на языке System Verilog сначала была реализована схема DDS с усечением значения фазового аккумулятора (это обусловлено тем, что более сложные и точные схемы основаны на именно такой схеме).

Интерфейс полученного описания ядра DDS показан ниже (\src\dds_top.sv).

```

1  `include "config.sv"
2
3  module dds_top
4  #(
5      parameter PH_NUM_ACC_WIDTH = P_PH_NUM_ACC_WIDTH,
6      parameter ROM_WIDTH = P_ROM_WIDTH,
7      parameter ROM_ADDR_WIDTH = P_ROM_ADDR_WIDTH
8  )
9  (
10     input clk,
11     input ce,
12     input rst,
13     input [(PH_NUM_ACC_WIDTH-1):0] phase_inc,
14     output signed [(ROM_WIDTH-1):0] minus_sin,
15     output signed [(ROM_WIDTH-1):0] cos
16 );
```

Исходное описание было сделано полностью параметризуемым, но за основу (для обеспечения сравнимости по качеству выходного сигнала с разработками Xilinx) были выбраны значения параметров из документации на Xilinx DDS Compiler 5.0: $B_{\theta(n)} = 32$, $B_{\theta(n)} = 12$, $B_s = 18$ (файл \src\config.sv содержит соответствующие параметры P_PH_NUM_ACC_WIDTH, P_ROM_ADDR_WIDTH, P_ROM_WIDTH), а в качестве выходных сигналов были взяты $-\sin$ и \cos .

Для генерации значений отсчётов $-\sin$ и \cos (которыми инициализируется ROM) была использована модель Simulink, показанная на рис. 2 (DDS\Sinus Generator). Ключ QUARTER_SYMMETRY_OPTIMIZED_ROMS файла \src\config.sv позволяет использовать только $\frac{1}{4}$ часть от памяти ROM для отсчётов полного периода $-\sin$ и \cos . Для выбранных параметров системы $B_{\theta(n)} = 12$ и $B_s = 18$ размер таблицы для полного периода $-\sin$ составляет $2^{12} = 4096$ отсчётов разрядности 18 бит (аналогично для \cos). Размер таблицы для четверти периода составляет 1025 отсчётов как для $-\sin$, так и для \cos . Так, объём памяти одной таблицы для четверти периода составляет $1025 * 18 = 18450$ бит или 2307 байт или 2,26 KB; суммарно для $-\sin$ и \cos : 4,52 KB.

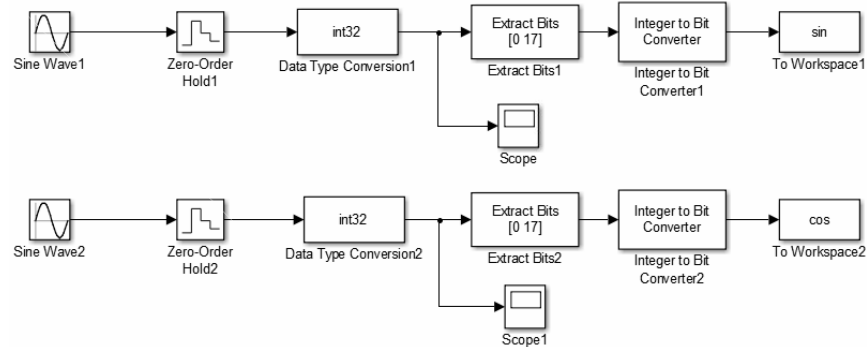


Рис. 2 – модель Simulink для генерации отсчётов $-\sin$ и \cos

На рис. 3 показана схема полученного DDS с усечением фазового числа.

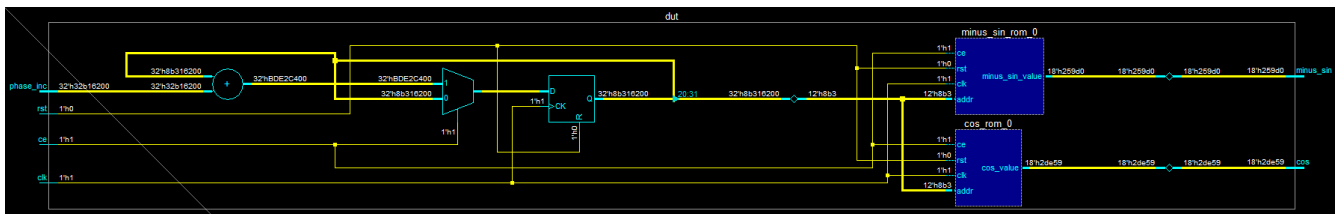


Рис. 3 – схема полученного DDS с усечением фазового числа, полученная в Questasim 10.2c

В ходе работы теста (\src\dds_top_tb.sv) были получены 16384 отсчёта сигналов $-\sin$ и \cos (результаты записываются в файлы), а с помощью встроенной функции MATLAB `sfdcr(x,1000)` (для удобства было принято, что $f_{clk} = 1$ кГц) была получена оценка SFDR для 16384 отсчётов выходного сигнала, взятых из файлов (как для \cos , так и для $-\sin$). Для теста был выбран $\Delta\theta = 850485760$, что соответствует выбранной частоте $f_{out} \approx 198$ Гц. Полученные результаты SFDR показаны на рис. 4. Как указано в документации на “Xilinx DDS Compiler 5.0”, для схемы с усечением фазового аккумулятора $B_{\theta(n)} = \text{ceiling}(\text{SFDR} / 6)$, то есть для $B_{\theta(n)} = 12$ будет обеспечено $\text{SFDR} = 72$ dB. Полученный для результат оценки SFDR для спроектированного DDS (рис. 4) составил ≈ 72 dB, что соответствует расчёту из “Xilinx DDS Compiler 5.0”.

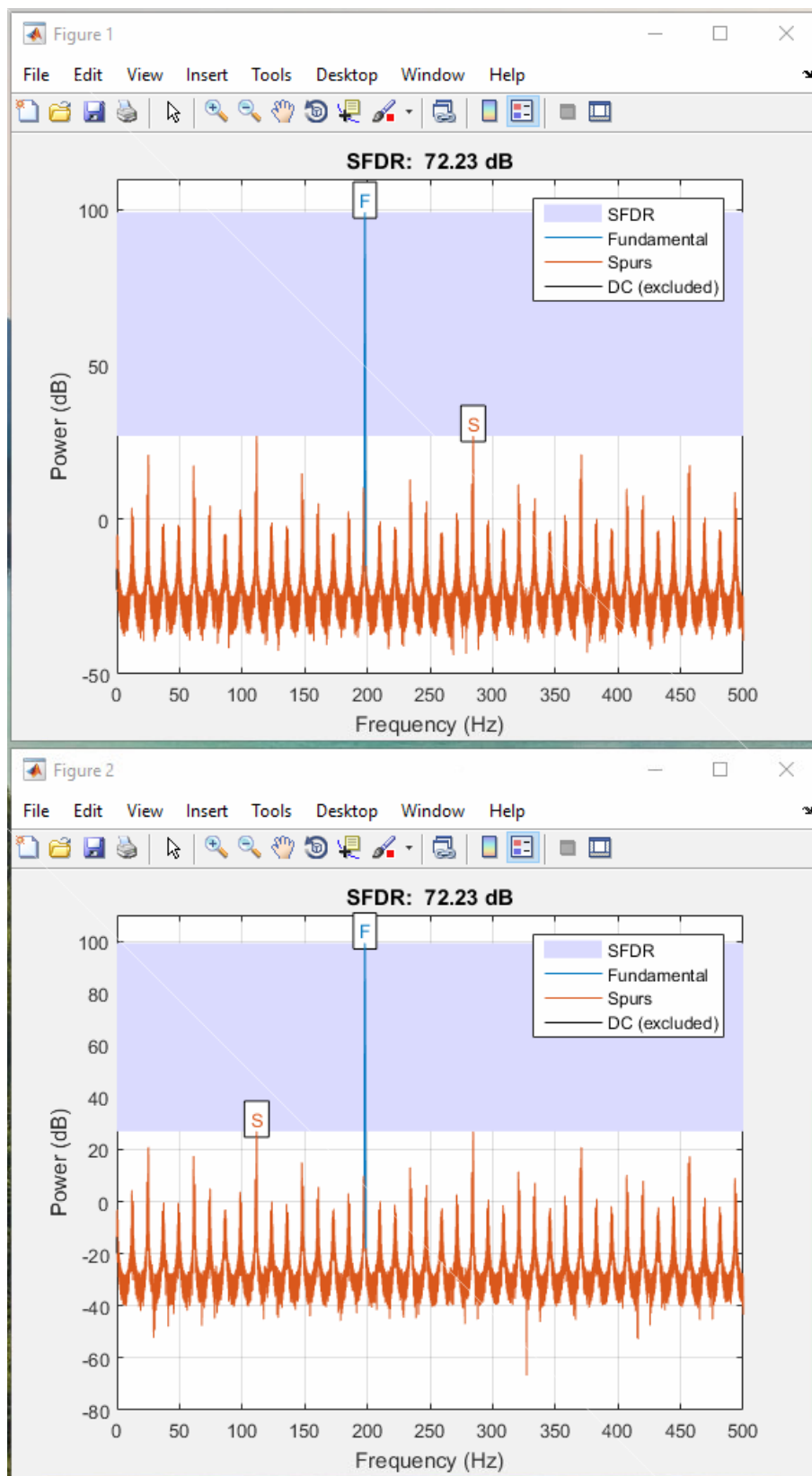


Рис. 4 – результат расчёта SFDR для схемы с усечением фазового аккумулятора для \cos (Figure 1) и для $-\sin$ (Figure 2) для $f_{out} \approx 198$ Гц.

Схема DDS с зашумлением значения аккумулятора фазового числа

Известной модификацией DDS с усечением значения аккумулятора фазового числа является введение в схему генератора случайного (псевдослучайного) аддитивного шума $d(n)$ (рис. 5). Эффективность такого решения объясняется тем, что для схемы без зашумления фазового числа ошибка фазового числа из-за отсечения младших разрядов является периодической и пилообразной, что создаёт пики на графике SFDR (рис. 4). Схема с зашумлением подавляет эти пики, хоть и увеличивает средний уровень шума (“Xilinx DDS v5.0”: “Phase Dithered DDS”). Однако диапазон SFDR в таком случае увеличивается, что и является желаемым качественным улучшением.

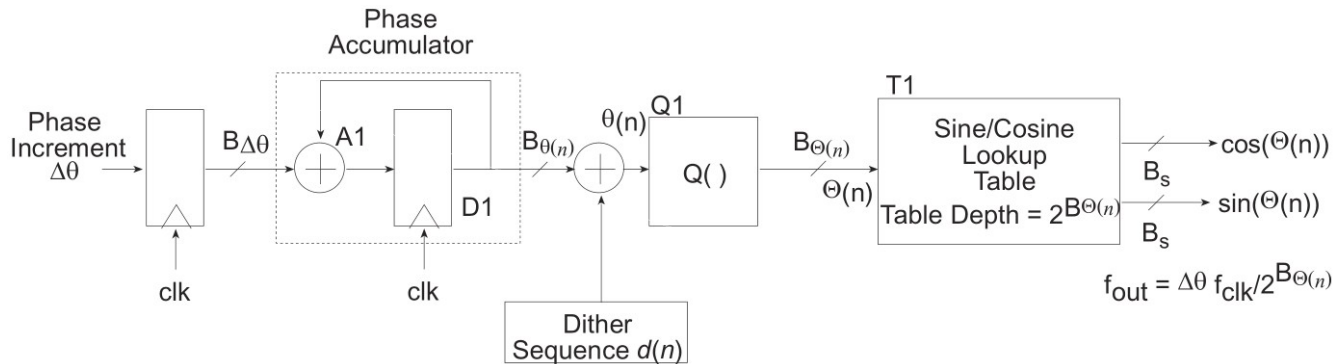


Рис. 5 – Схема DDS с зашумлением значения фазового аккумулятора (Phase Dithering)

Схема DDS с зашумлением значения фазового аккумулятора позволяет получить SFDR выше, чем обычная схема с усечением числа фазового аккумулятора: $B_{\Theta(n)} = \text{ceiling}(\text{SFDR} / 6) - 2$, то есть введение генератора шума позволяет получать такой же SFDR, как и в обычной схеме с усечением фазового числа, но при использовании на два разряда меньшего адреса таблиц T1 (это позволяет экономить память). При использовании одинаковых основных параметров DDS введение зашумления фазового числа добавляет (теоретически) примерно 12 dB к SFDR, возможному для схемы без зашумления. Таким образом, для выбранных параметров DDS можно получить, теоретически, 84 dB SFDR, когда как для предыдущей схемы без зашумления было получено 72 dB SFDR.

На следующем этапе проектирования DDS на языке System Verilog была реализована вышеуказанная схема с зашумлением фазового сигнала. Ключ DITHERING_CORRECTION файла \src\config.sv позволяет задействовать участки кода, отвечающие за внедрение генератора шума. Генератор шума был спроектирован на базе двух 10-разрядных LFSR (Linear Feedback Shift Register, файл \src\lfsr.sv), при этом выходные псевдослучайные 10-разрядные слова двух LFSR конкатенируются, давая 20-разрядный шумовой вектор. Так, составной генератор шума псевдослучайным образом вычисляет шумовой вектор, разрядность которого равна разрядности отсекаемой части аккумулятора фазового числа (20 разрядов), затем данный вектор добавляется в младшие разряды текущего полного значения аккумулятора. Важно отметить, что данная секция кода не параметризуется ввиду того, что невозможно полностью параметризовать генератор псевдослучайной последовательности на базе LFSR; поэтому, для каждого нового набора основных параметров DDS может потребоваться перепроектировать генератор псевдослучайного шума.

Для проверки качества выходных сигналов $-\sin$ и \cos был использован аналогичный тест, который был описан в предыдущем разделе. На рис. 6 показан результат расчёта SFDR для схемы с зашумлением фазового числа. Полученный результат $\text{SFDR} \approx 87 \text{ dB}$, что соответствует

расчётному значению и для заданной частоты на 3 dB его превосходит.

В System-Verilog-описании в закомментированном блоке кода можно увидеть строку с “std::randomize(dither_vector)”. Данная строка является несинтезируемой, и её использование в проверочных целях позволяет подменить синтезируемый LFSR-генератор псевдослучайных чисел на несинтезируемый генератор случайных чисел из библиотеки System Verilog. Сопоставление кода с использованием “std::randomize(dither_vector)” и кода с синтезируемым LFSR показало, что при вышеуказанном тесте генератор на основе спроектированного LFSR позволяет получить приблизительно на 1 dB лучший результат.

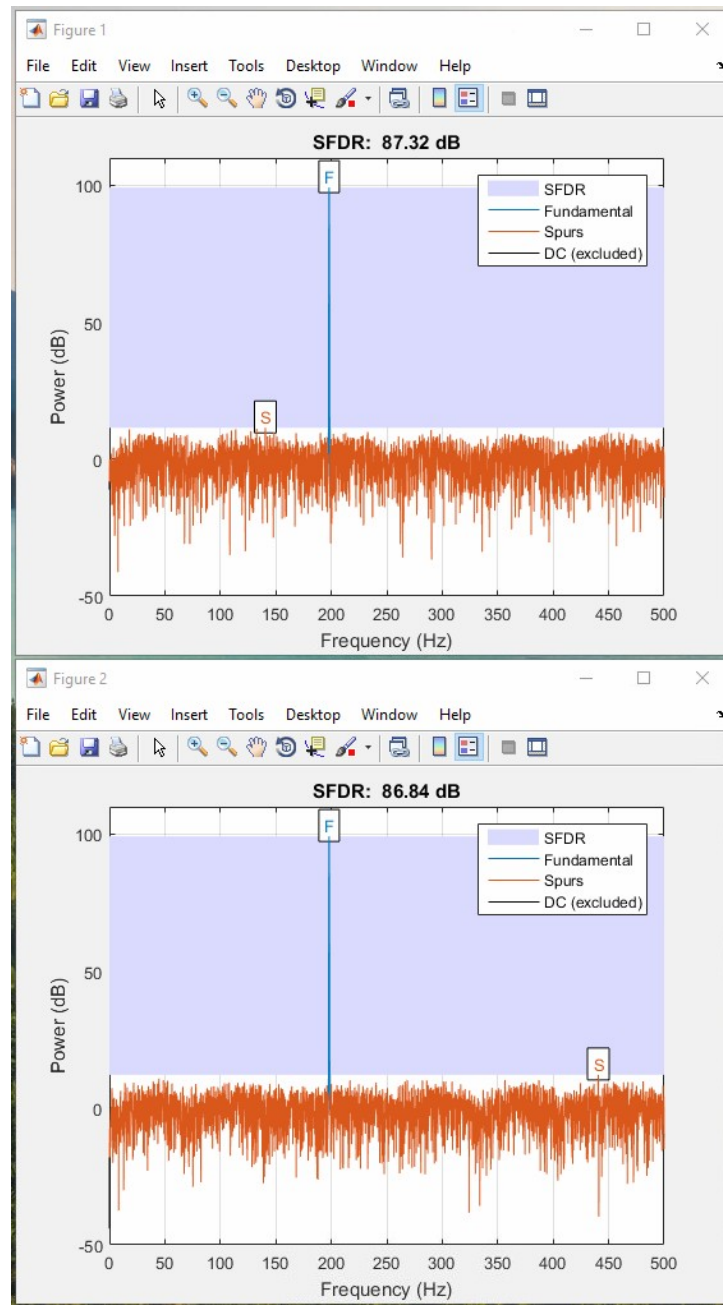


Рис. 6 – результат расчёта SFDR для схемы с зашумлением фазового аккумулятора для cos (Figure 1) и для -sin (Figure 2) для $f_{out} \approx 198$ Гц.

Схема DDS с коррекцией на основе ряда Тейлора

Ещё одной известной и более точной модификацией DDS с усечением значения аккумулятора фазового числа является схема с коррекцией выходных значений таблиц \sin и \cos на основе усечённых данных. Расчёт корректирующих значений ведётся согласно рядам Тейлора для \sin и \cos . В общем виде ряд Тейлора можно записать как:

$$f(x) = f(a) + \frac{(x-a)f'(a)}{1!} + \frac{(x-a)^2 f''(a)}{2!} + \dots + \frac{(x-a)^N f^{(N)}(a)}{N!}$$

Тогда, выбрав только два первых элемента ряда, для $\sin(x)$ и $\cos(x)$ можно приближённо считать:

$$\sin(x) = \sin(a) + (x-a)\cos(a)$$

$$\cos(x) = \cos(a) - (x-a)\sin(a)$$

где x соответствует $\theta(n)$, a соответствует $\Theta(n)$, $\sin(a)$ – значение, полученное из таблицы (ROM), $\sin(x)$ – уточнённое значение $\sin(a)$, $(x-a)$ – ошибка фазы. Важно отметить, что $(x-a)$ является ошибкой фазы, а не фазового числа: для вычисления ошибки фазы выражение $(\theta(n) - 2^{(B_{\theta(n)} - B_{\Theta(n)})} * \Theta(n))$, соответствующее $(x-a)$, необходимо домножить на $(2\pi / 2^{B_{\theta(n)}})$. Для получения $(\theta(n) - 2^{(B_{\theta(n)} - B_{\Theta(n)})} * \Theta(n))$ достаточно просто считать усечаемую часть фазового числа и дополнить её слева нулями до разрядности $B_{\theta(n)}$: значит, в данном случае, фактически, не нужно производить вычисления. Используя вышеуказанные формулы и пояснения, можно составить наглядную схему DDS с коррекцией на основе ряда Тейлора (рис. 7).

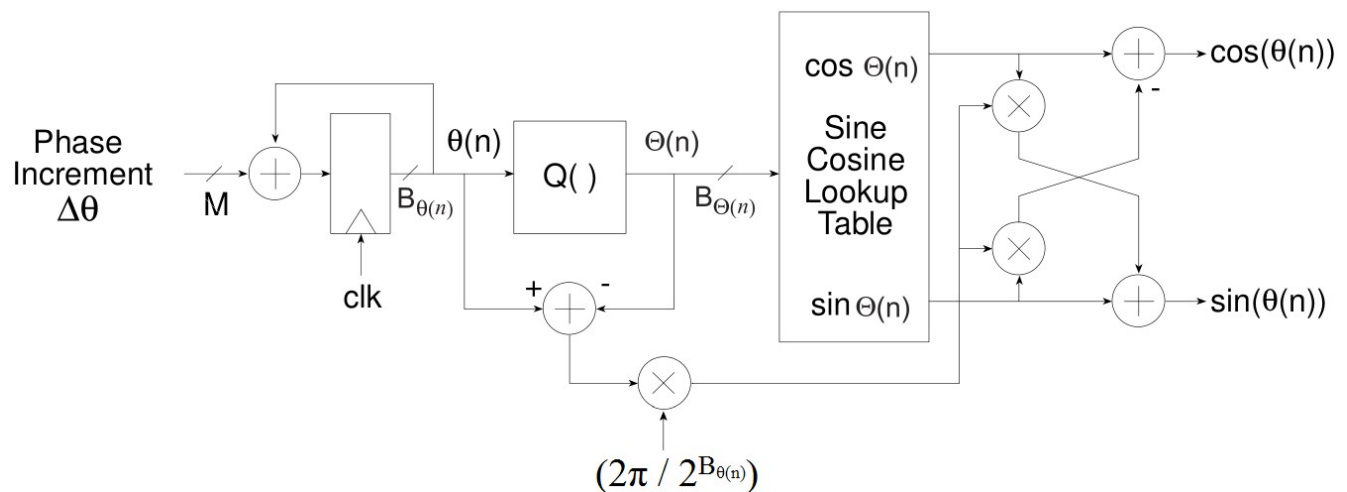


Рис. 7 – Упрощённая схема DDS с коррекцией на основе ряда Тейлора

Таким образом, схема DDS с коррекцией на основе ряда Тейлора относительно обычной схемы с усечением фазового числа требует три дополнительных умножителя и два дополнительных оконечных сумматора.

На следующем этапе проектирование ядра DDS на языке System Verilog была реализована логика коррекции на основе ряда Тейлора (ключ TAYLOR_SERIES_CORRECTION файла \src\config.sv). Перед проектированием была изучена документация на ядра DDS фирмы Xilinx. На рис. 8 показана Simulink-схема DDS с коррекцией на основе ряда Тейлора (схема взята из “FPGA Signal Processing - Direct Digital Synthesis”). На схеме используются те же самые (взятые

за основу ранее) параметры DDS. Результат расчёта SFDR для такой схемы показан на рис. 9.

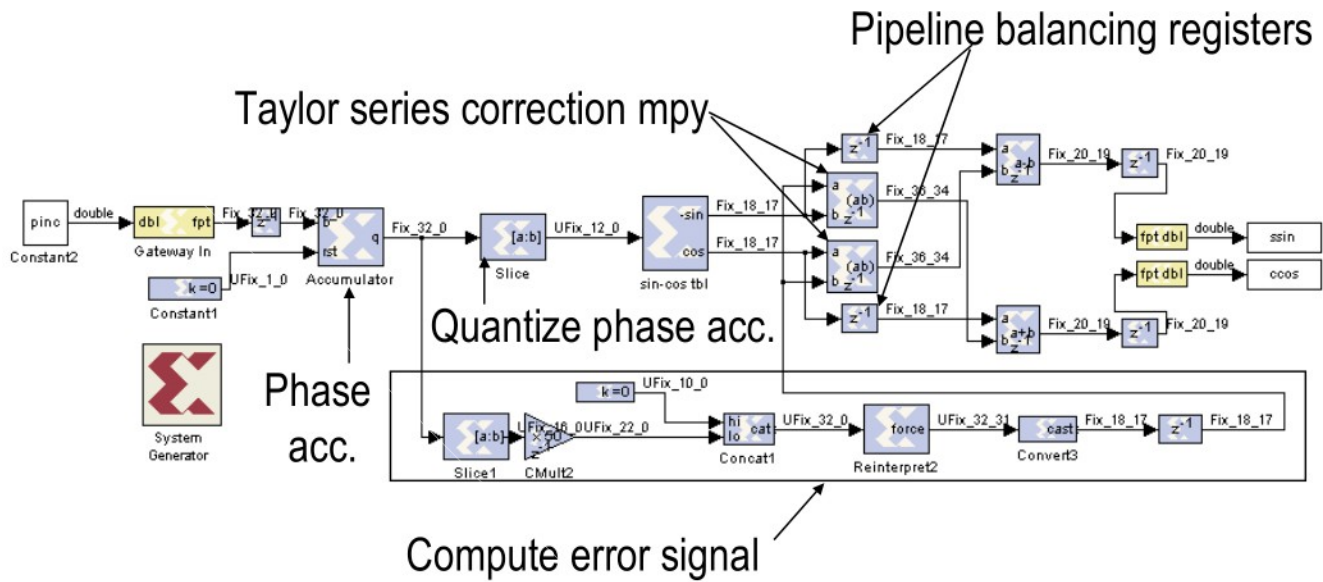


Рис. 8 – Simulink-схема DDS с коррекцией на основе ряда Тейлора

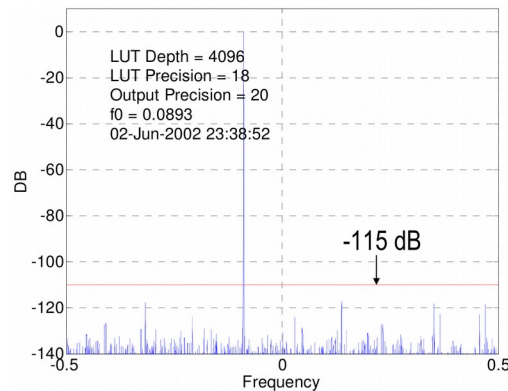


Рис. 9 – Расчёт SFDR для Simulink-схемы DDS с коррекцией на основе ряда Тейлора

На рис. 8 особый интерес представляют этапы расчёты блока “Compute error signal”. На данной схеме берутся только старшие 16 разрядов отсекаемых разрядов (число N), а младшие 4 разряда просто отбрасываются, после чего они умножаются на 50, где $50 \approx (\pi * 16)$, где $\pi \approx (11.0010)_2$. Расчёт ошибки фазы в таком случае выглядит как $(2^{-32} * 2 * (\pi * 16) * N) = (2^{-31} * 50 * N)$. В дальнейшем на схеме к представлению числа ошибки добавляется знаковый разряд (0 так как здесь может быть только положительное число), причём число ошибки представляется как UFix_32_31, то есть сразу после знакового разряда в представлении числа идут разряды после запятой. Далее число фазовой ошибки округляется до формата UFix_18_17, так как именно такой формат используют отчёты синусоид в таблицах -sin и cos.

Преимуществом схемы на рис. 8 является то, что для умножения на константу 50 в данном случае используется умножитель разрядности только 16×6 . Но из-за взятия неполного отбрасываемого числа и взятия числа π с небольшой точностью (всего 6 разрядов), точность подсчёта ошибки фазы оказывается ниже желаемой. При проектировании ядра на System Verilog сначала была выбрана именно такая структура. Однако затем структура была изменена: для 32-

разрядного аккумулятора были взяты все 20 отсекаемых разрядов, а число π было взято с большей разрядностью ($\pi \approx (11.001001)_2 = 201 * 2^{-6}$). Полученная схема использует умножитель разрядности 20×8 , но она позволяет получить до +1 dB к SFDR на практике.

Также можно обратить внимание, что на рис. 8 выходные отсчёты имеют разрядность 20 бит, хотя разрядность значений таблиц составляет 18 бит. В ходе тестов обнаружилось, что 20-битный вывод данных после схемы коррекции на основе ряда Тейлора в пределах погрешности соответствует 18-битному выводу по параметру SFDR, в то время как 20-разрядный вывод не сохраняет полный размах для 20-разрядных отсчётов $-\sin$ и \cos . Ввиду этого было принято решение использовать 18-разрядный вывод.

При проектировании умножителей и сумматоров были учтены вопросы округления и переполнения разрядной сетки. При конечных умножениях формата $U_{Fix_18_17} \times U_{Fix_18_17}$ полученные результаты формата $U_{Fix_36_34}$ округляются до формата $U_{Fix_18_17}$ по схеме уточнённого округления к ближайшему дробному числу (такая схема требует сумматора с единицей, ключ `ENABLE_ROUNDING_TO_NEAREST_FRACTION`). Для предварительных System-Verilog-описаний без применения округления (но с применением отсечения) возникали ситуации переполнения разрядной сетки при суммировании элементов ряда Тейлора (с последующим инвертированием знака числа), что вызывало значительный шум и катастрофическое уменьшение SFDR до 50 dB. Кроме того, введение округления позволило незначительно увеличить SFDR (увеличение составило менее 1 dB). Для избежания возможных редких ситуаций переполнения разрядной сетки даже в случае использования округления в описание были введены блоки коррекции переполнения разрядной сетки (при коррекции переполнения используются максимальное положительное и минимальное отрицательное значение синусоиды для заданной разрядности, ключ `ENABLE_OVERFLOW_CORRECTION`).

Для коррекции ситуации, в которой $-\sin$ или \cos после суммирования элементов ряда Тейлора получают значение формата $U_{Fix_18_17}$ равное 100000000000000000, для которого нет парного положительного числа у $-\sin$ или \cos , в схему опционально вводится блок ограничения значений для $-\sin$ или \cos (возможное 1000000000000000000 заменяется на 100000000000000001, для которого есть положительная пара 0111111111111111). За задействование этой подсхемы отвечает ключ `FORCE_CONSTRAINED_OUTPUT` файла `\src\config.sv`.

Согласно другим источникам (“Xilinx DDS v5.0”, “Xilinx DDS Compiler 5.0”), использование схемы DDS с коррекцией на основе ряда Тейлора с двумя элементами ряда позволяет получить для указанных параметров ядра DDS ($B_{\theta(n)} = 32$, $B_{\Theta(n)} = 12$, $B_s = 18$) приблизительно 118 dB SFDR, что на 46 dB больше того SFDR, который можно получить для простейшего DDS с усечением фазового числа.

Для проверки качества выходных сигналов $-\sin$ и \cos для полученного System-Verilog-описания был использован тот же самый тест, который был указан в предыдущих разделах. На рис. 10 показан результат расчёта SFDR для полученной схемы DDS с коррекцией на основе ряда Тейлора. Полученный результат SFDR ≈ 118 dB, что соответствует расчётному значению из “Xilinx DDS v5.0” и “Xilinx DDS Compiler 5.0”.

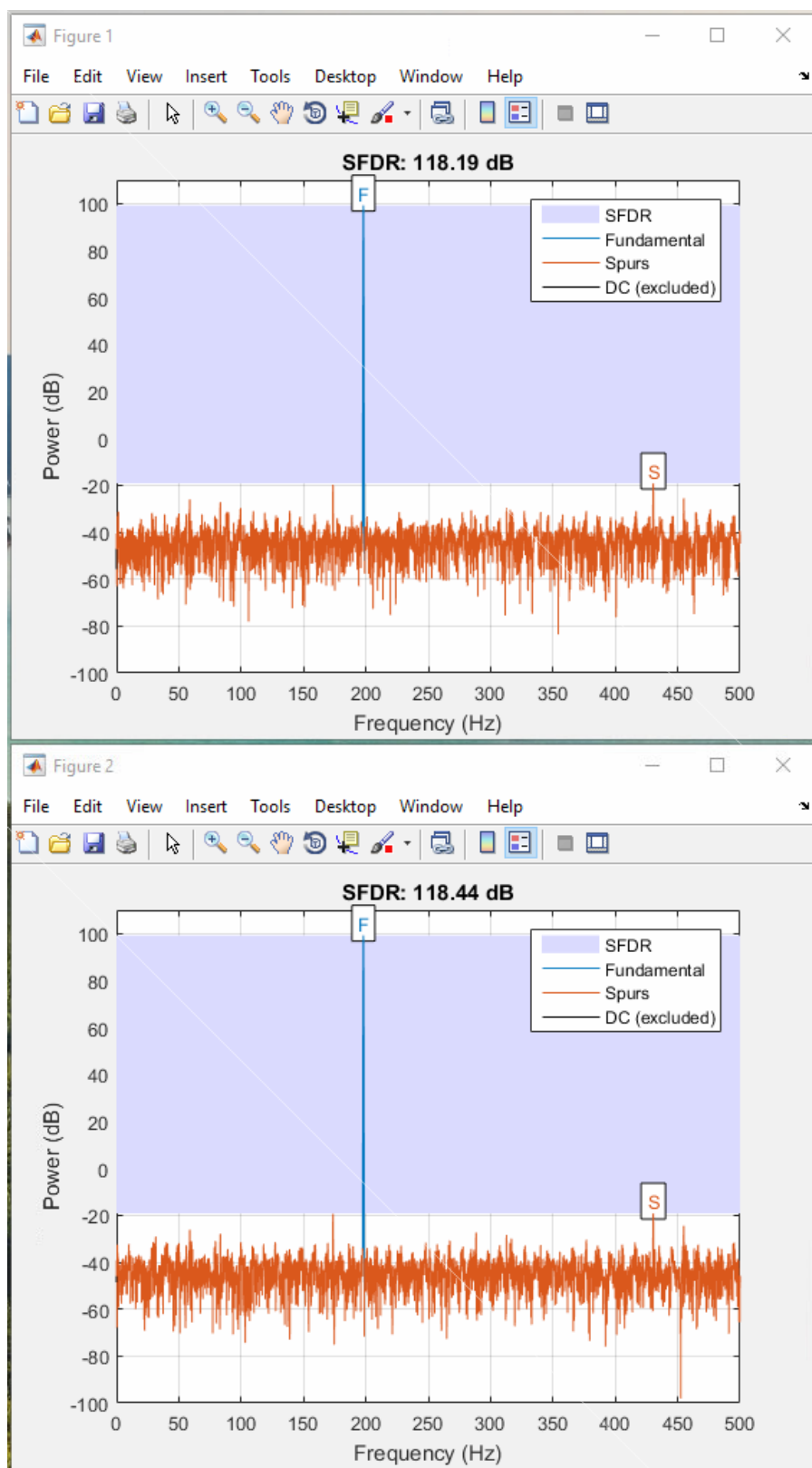


Рис. 8 – результат расчёта SFDR для схемы DDS с коррекцией на основе ряда Тейлора для \cos (Figure 1) и для $-\sin$ (Figure 2) для $f_{out} \approx 198$ Гц.

Важно отметить, что одновременное использование схемы DDS и с зашумлением фазового числа, и с коррекцией на основе ряда Тейлора даёт SFDR лишь на несколько dB лучший (около 89 dB SFDR), чем схема только с использованием зашумления (около 87 dB SFDR), что делает такую схему бессмысленной, так как использование схемы только с коррекцией на основе ряда Тейлора даёт SFDR равный примерно 118 dB.

Для вышеуказанной схемы DDS с коррекцией на основе ряда Тейлора были проведены расчёты SFDR и для других значений $\Delta\theta$ (таблица 1). Для наглядности расчётов было принято, что $f_{clk} = 1$ кГц. Большая часть записей таблицы как для $-\sin$, так и для \cos имеет значения близкие к 118 dB SFDR. Но в таблице есть и существенные отклонения от этого значения.

Значение $\Delta\theta = 1_048_576$ показывает самый высокий результат SFDR, так как это значение подобрано таким образом, чтобы ошибка фазы всегда была равна нулю на любом этапе работы DDS. Так как значение ошибки для такого $\Delta\theta$ всегда равно нулю, значит SFDR в данном случае будет определяться только качеством отсчётов, хранящихся в ROM. Важно отметить, что в этом случае схема работает так, как если бы в ней вообще не было корректирующей логики. Значение $\Delta\theta = 10_485_760$ даёт всё ещё достаточно малую среднюю ошибку фазы, ввиду чего SFDR снизился не слишком сильно относительно $\Delta\theta = 1_048_576$.

На очень важное свойство [схем DDS на основе таблиц предрасчитанных отсчётов вообще] указывают записи для $\Delta\theta = 429_496_729, 429_496_730$ (в таблице они выделены красным). Для этих локализованных значений $\Delta\theta$ наблюдается заметное уменьшение SFDR по отношению к наиболее частым показателям 118 dB SFDR. Такие локальные просадки SFDR объясняются тем, что для значений $\Delta\theta$, которые не взаимно просты с полным значением фазы $2^{B_{\theta(n)}}$, ошибки округления являются когерентными (накапливаются), когда как для иных $\Delta\theta$ ошибки вычислений взаимно вычитаются и общая ошибка остаётся мала. На рис. 10 показана вырезка из “Xilinx DDS Compiler 5.0”, посвящённая этому вопросу. Таким образом, согласно “Xilinx DDS Compiler 5.0”, считаются нежелательными значения $\Delta\theta$, такие, что $(2^{B_{\theta(n)}} = k/n * \Delta\theta)$, где k/n простая рациональная дробь. В конкретном примере $\Delta\theta = 429_496_730$, что с высокой точностью равно $1/10$ от 2^{32} .

Known Issues

Sub-Harmonic Frequencies

The equations for SFDR rely on the assumption that rounding errors from the finite precision of phase and amplitude are incoherent. This assumption is violated for values of Phase Increment that are not mutually prime with the weight of the Phase Accumulator. The anomalies, such as spurs, will be more obvious for larger common factors between the Phase Increment Value (PINC) and the weight of the accumulator ($2^{\text{Phase_Width}}$). This is because such values may not access every location in the SIN/COS Lookup table, so the rounding errors are not randomly spread. To avoid this, do not use values of Output Frequency that are simple rational fractions of the frequency per channel, F_s , such as $3/8, 1/64$.

Рис. 10 – Вырезка из “Xilinx DDS Compiler 4.0”, касающаяся нежелательных значений $\Delta\theta$

Таблица 1. SFDR для различных $\Delta\theta$

$\Delta\theta$	f_{out} Гц	SFDR для -sin, dB	SFDR для cos, dB
1_048_576	0,244	130.34	130.30
10_485_760	2,441	126.24	126.24
50_485_760	11,755	118.19	118.13
100_485_760	23,396	118.51	117.46
200_485_760	46,679	117.59	118.39
300_485_760	69,962	117.52	117.82
400_485_760	93,245	118.05	118.03
428_496_729	99,767	117.13	118.45
429_496_729	99,9999999	108.25	107.09
429_496_730	100,000	113.03	106.31
430_496_730	100,233	118.38	117.75
450_485_760	104,887	118.24	118.62
850_485_760	198,019	118.19	118.44

Для расчётов с ещё большей точностью можно учитывать и третий элемент ряда Тейлора. Так, взяв только первые три элемента ряда Тейлора, для \sin и \cos расчёт можно представить, как:

$$\begin{aligned}\sin(x) &= \sin(a) + (x-a) \cdot \cos(a) - 1/2 \cdot (x-a)^2 \cdot \sin(a) \quad , \\ \cos(x) &= \cos(a) - (x-a) \cdot \sin(a) + 1/2 \cdot (x-a)^2 \cdot \cos(a) \quad .\end{aligned}$$

Для ежетапной генерации отсчётов вышеуказанные выражения требуют дополнительной логики относительно двух-элементной коррекции по ряду Тейлора: необходим ещё один множитель для возведения ошибки фазы в квадрат, два дополнительных множителя для умножений на $\sin(a)$ и $\cos(a)$ соответственно, а также два дополнительных сумматора.

Схема DDS с тремя элементами ряда Тейлора была реализована в описании DDS на языке System Verilog. Для включения блоков кода, отвечающих за конфигурацию с тремя элементами ряда Тейлора в файле `\src\config.sv` должны быть задействованы ключ `TAYLOR_SERIES_CORRECTION` и ключ `TAYLOR_SERIES_THIRD_TERM`.

Ввиду того, что ошибка фазы $(x-a)$ значительно меньше единицы, то $(x-a)^2$ является очень маленьким числом. Для взятых за основу параметров DDS ($B_{\theta(n)} = 32$, $B_{\phi(n)} = 12$, $B_s = 18$) в ходе тестов обнаружилось, что округление $(x-a)^2$ до формата `UFix_18_17` во всех случаях превращает значение выражения $(x-a)^2$ в ноль. Таким образом, для данной конфигурации DDS введение дополнительной коррекции с помощью третьего элемента ряда Тейлора не имеет смысла, а показатели SFDR полностью соответствуют полученным для двухэлементной коррекции. Однако для других (более точных) конфигураций DDS третий элемент ряда Тейлора уже будет иметь вес, а значит позволит повысить точность ещё больше за счёт введения дополнительных ресурсов.

Синтез полученных схем DDS для FPGA

Результаты синтеза всех описанных выше модификаций DDS в Synopsys Synplify Pro G-2012.09 показаны ниже (рис. 11-14).

Project Settings							
Project Name	dds_prj	Implementation Name		rev_1			
Top Module	[auto]	Pipelining		0			
Retiming	0	Resource Sharing		0			
Use Xilinx Xflow	0	Fanout Guide		10000			
Disable I/O Insertion	0	Disable Sequential Optimizations		0			
Clock Conversion	1	Use Xilinx Partition Flow		0			

Run Status							
Job Name	Status				CPU Time	Real Time	Memory
Compile Input Detailed report	Complete	7	2	0	-	0m:02s	-
Premap Detailed report	Complete	3	1	0	0m:00s	0m:00s	196MB
Map & Optimize Detailed report	Complete	162	36	0	0m:02s	0m:02s	196MB

Area Summary			
I/O ports	71	Non I/O Register bits	37 (0%)
I/O Register bits	0	Block Rams	2 (416)
DSP48s	0 (768)	LUTs	81 (0%)
Detailed report		Hierarchical Area report	

Timing Summary			
Clock Name	Req Freq	Est Freq	Slack
dds_topclk	260.0 MHz	370.8 MHz	1.149
Detailed report			

Optimizations Summary	
Combined Clock Conversion	1 / 0 more

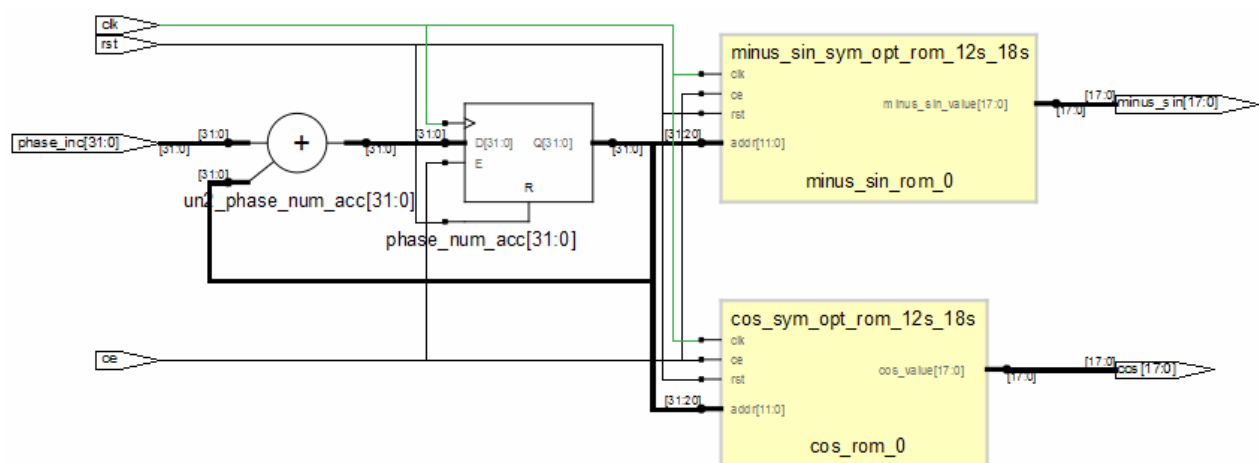


Рис. 11 – Результаты синтеза схемы DDS с усечением значения фазового аккумулятора для Xilinx Virtex 6 : XC6VLX240T : FF784 : -1

Project Settings							
Project Name	dds_prj	Implementation Name		rev_1			
Top Module	[auto]	Pipelining		1			
Retiming	1	Resource Sharing		1			
Use Xilinx Xflow	0	Fanout Guide		10000			
Disable I/O Insertion	0	Disable Sequential Optimizations		0			
Clock Conversion	1	Use Xilinx Partition Flow		0			

Run Status							
Job Name	Status				CPU Time	Real Time	Memory
Compile Input Detailed report	Complete	9	3	0	-	0m:01s	-
Premap Detailed report	Complete	3	1	0	0m:00s	0m:00s	196MB
Map & Optimize Detailed report	Complete	163	38	0	0m:02s	0m:02s	196MB
						Date/Time	
						14.03.2016 10:42:49	
						14.03.2016 10:42:51	
						14.03.2016 10:42:54	

Area Summary			
I/O ports	71	Non I/O Register bits	87 (0%)
I/O Register bits	0	Block Rams	2 (416)
DSP48s	0 (768)	LUTs	129 (0%)
Detailed report		Hierarchical Area report	

Timing Summary			
Clock Name	Req Freq	Est Freq	Slack
dds_topclk	230.0 MHz	372.0 MHz	1.659
Detailed report			

Optimizations Summary			
Retiming	0 / 4 more	Combined Clock Conversion	1 / 0 more

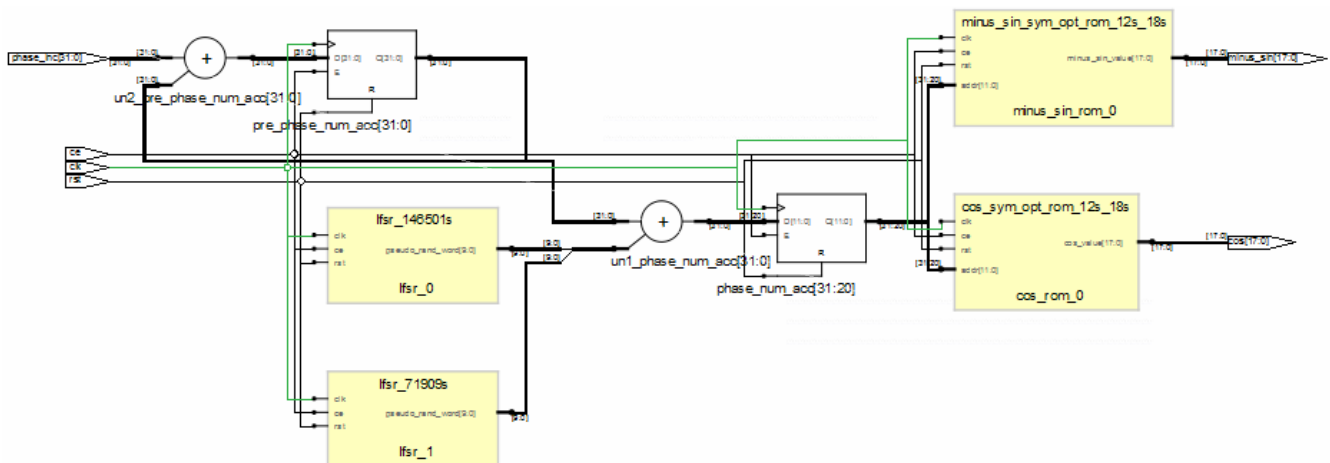


Рис. 12 – Результаты синтеза схемы DDS с зашумлением значения фазового аккумулятора для Xilinx Virtex 6 : XC6VLX240T : FF784 : -1

Project Settings							
Project Name	dds_prj	Implementation Name		rev_1			
Top Module	[auto]	Pipelining		1			
Retiming	1	Resource Sharing		1			
Use Xilinx Xflow	0	Fanout Guide		10000			
Disable I/O Insertion	0	Disable Sequential Optimizations		0			
Clock Conversion	1	Use Xilinx Partition Flow		0			

Run Status							
Job Name	Status				CPU Time	Real Time	Memory
Compile Input Detailed report	Complete	8	4	0	-	0m:01s	-
Premap Detailed report	Complete	3	1	0	0m:00s	0m:00s	197MB
Map & Optimize Detailed report	Complete	59	16	0	0m:11s	0m:11s	198MB
						14.03.2016 11:45:37	

Area Summary			
I/O ports	71	Non I/O Register bits	258 (0%)
I/O Register bits	0	Block Rams	0 (416)
DSP48s	3 (768)	LUTs	803 (0%)
Detailed report		Hierarchical Area report	

Timing Summary			
Clock Name	Req Freq	Est Freq	Slack
dds_topclk	230.0 MHz	167.0 MHz	-1.640
Detailed report			

Optimizations Summary			
Retiming	28 / 67 more	Combined Clock Conversion	1 / 0 more

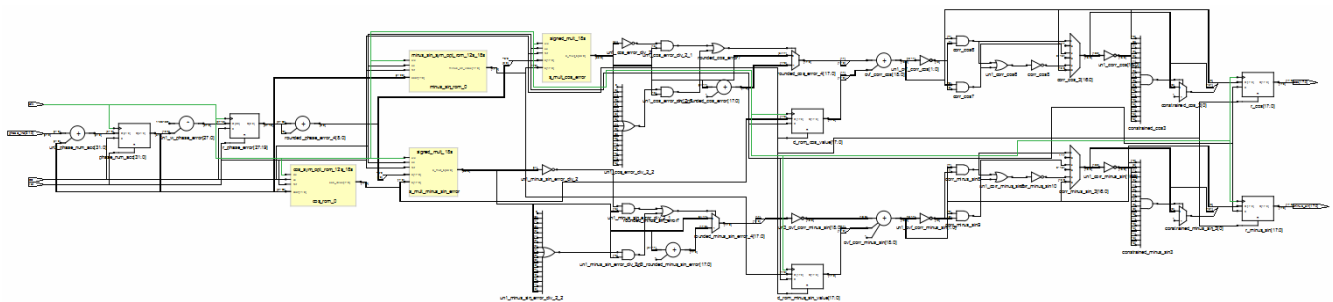


Рис. 13 – Результаты синтеза схемы DDS с коррекцией на основе ряда Тейлора для Xilinx Virtex 6 : XC6VLX240T : FF784 : -1

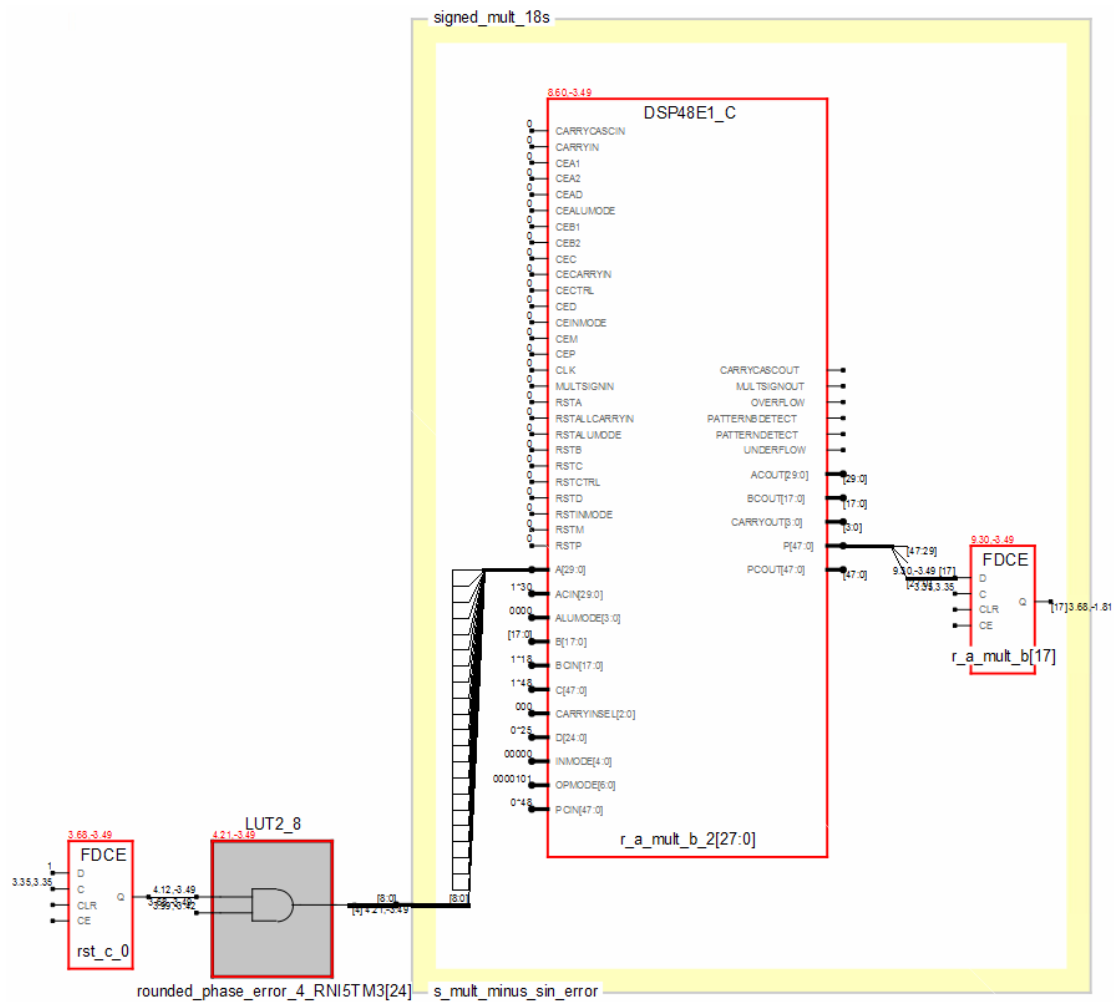


Рис. 14 – Результаты синтеза схемы DDS с коррекцией на основе ряда Тейлора для Xilinx Virtex 6 : XC6VLX240T : FF784 : -1: критический путь

Таким образом (рис. 14), максимальная частота схемы DDS с коррекцией на основе ряда Тейлора ограничивается скоростью срабатывания комбинаторной логики умножителя.

Результаты для схемы DDS, полученной в Xilinx DDS Compiler 4.0

Ниже представлены результаты для схемы DDS с коррекцией на основе ряда Тейлора, полученной в Xilinx DDS Compiler 4.0 (рис. 15-17).

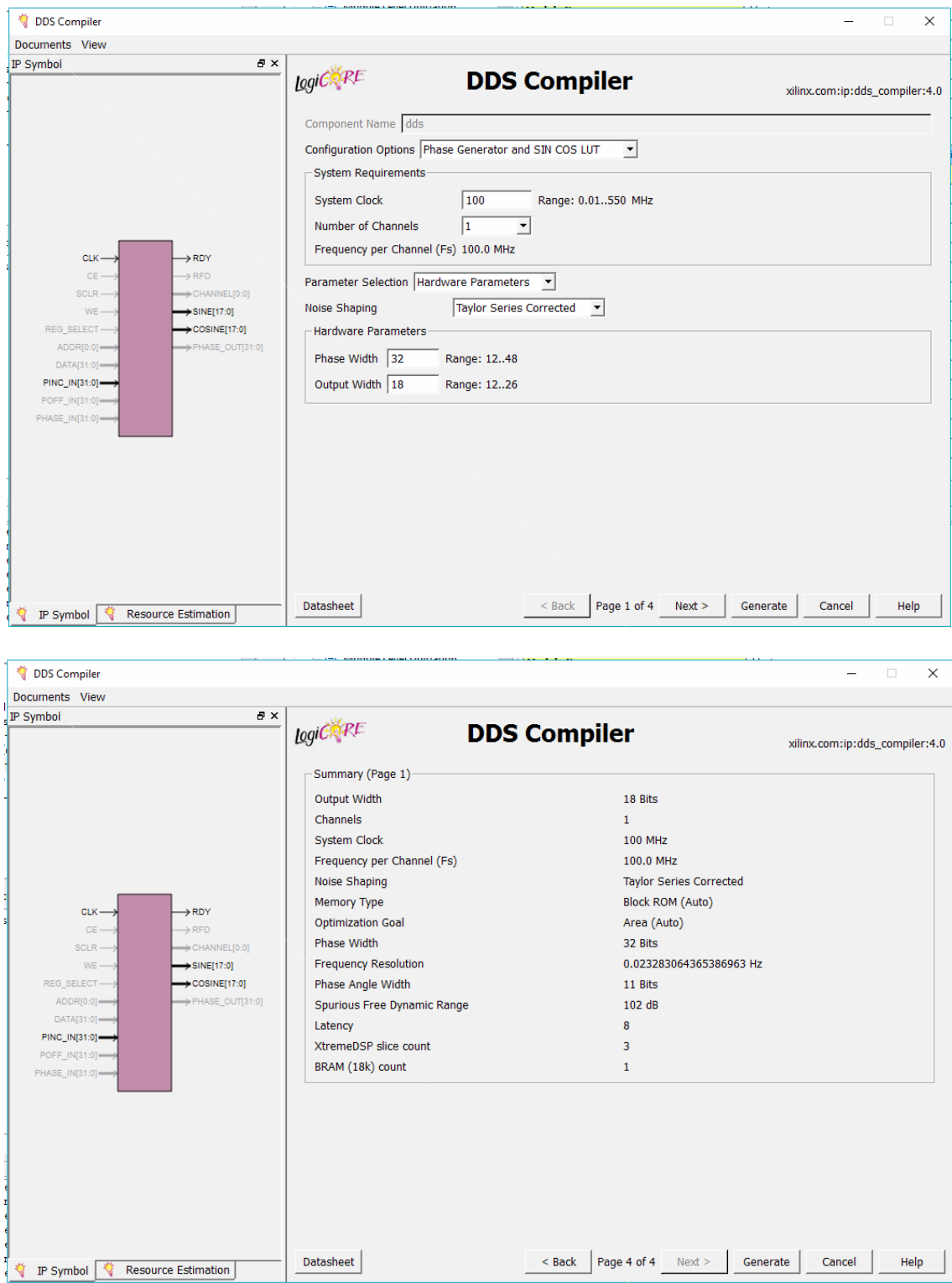


Рис. 15 – параметры для генерации схемы Xilinx DDS с коррекцией на основе ряда Тейлора

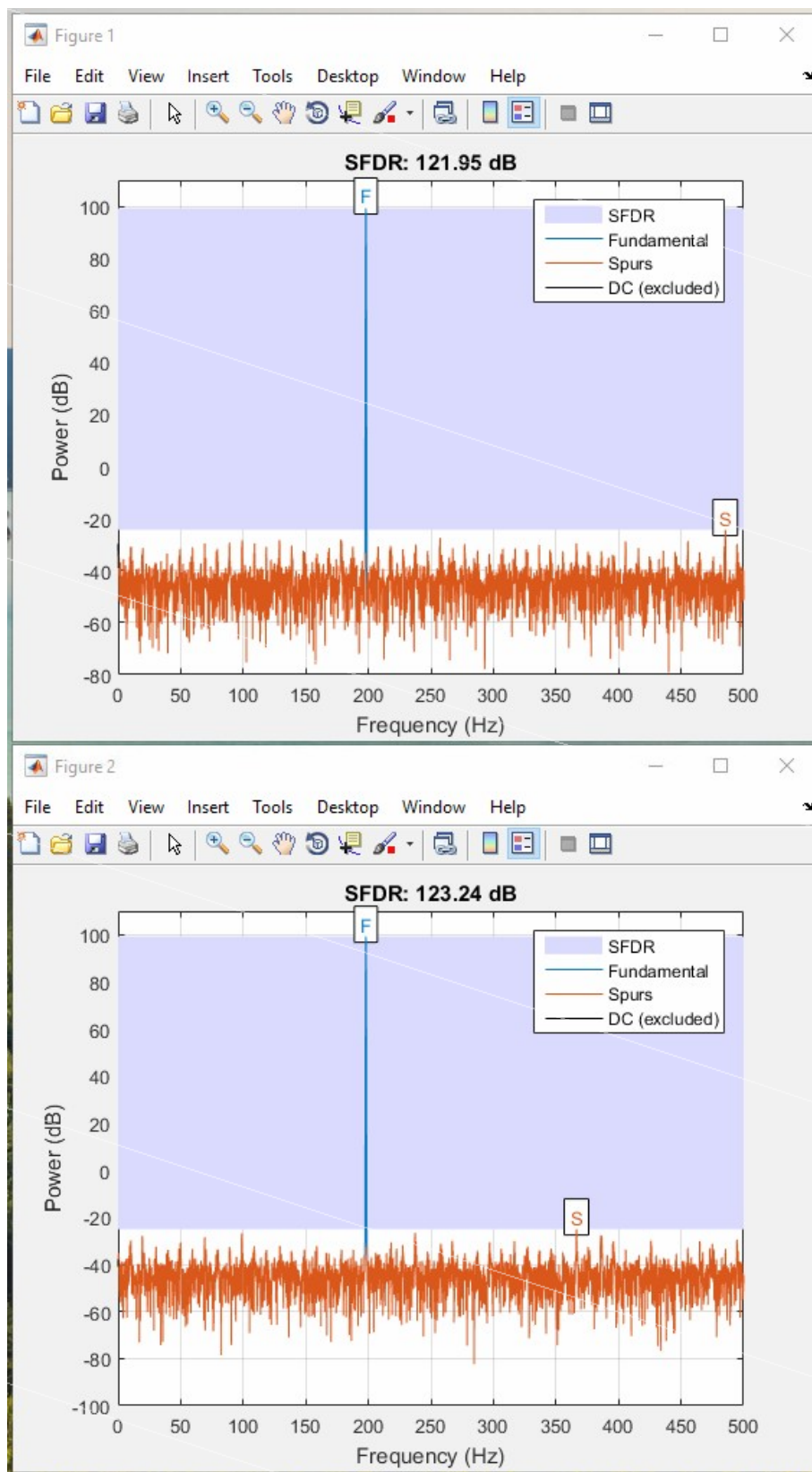


Рис. 16 – результат расчёта SFDR для схемы Xilinx DDS с коррекцией на основе ряда Тейлора для \cos (Figure 1) и для $-\sin$ (Figure 2) для $f_{out} \approx 198$ Гц.

Project Settings							
Project Name	dds_top	Implementation Name		xilinx_dds			
Top Module	dds_top	Pipelining		1			
Retiming	0	Resource Sharing		1			
Use Xilinx Xflow	0	Fanout Guide		100			
Disable I/O Insertion	0	Disable Sequential Optimizations		0			
Clock Conversion	1	Use Xilinx Partition Flow		0			

Run Status							
Job Name	Status				CPU Time	Real Time	Memory
Compile Input Detailed report	Complete	15	0	0	-	0m:04s	-
Premap Detailed report	Complete	28	1	0	0m:01s	0m:02s	197MB
Map & Optimize Detailed report	Complete	34	1	0	0m:12s	0m:17s	196MB
						Date/Time	
						14.03.2016 14:43:28	
						14.03.2016 14:43:32	
						14.03.2016 14:43:50	

Area Summary			
I/O ports	70	Non I/O Register bits	148 (0%)
I/O Register bits	0	Block Rams	1 (416)
DSP48s	3 (768)	LUTs	55 (0%)
Detailed report		Hierarchical Area report	

Timing Summary			
Clock Name	Req Freq	Est Freq	Slack
dds_topclk	682.0 MHz	579.7 MHz	-0.259
Detailed report			

Optimizations Summary	
Combined Clock Conversion	1 / 0 more

Рис. 17 – Результаты синтеза схемы Xilinx DDS с коррекцией на основе ряда Тейлора для Xilinx Virtex 6 : XC6VLX240T : FF784 : -1

Как видно, при попытке сгенерировать схему DDS с помощью DDS Compiler 4.0 с параметрами аналогичными проектируемой разработке ($B_{0(n)} = 32$, $B_{\Theta(n)} = 12$, $B_S = 18$), получаются результаты чуть лучшие (~ 122 - 123 dB SFDR, рис. 16), чем у спроектированного DDS (~ 118 dB SFDR, рис. 8). Данный результат также лучше результата, заявленного в документации на DDS Compiler 4.0 (~ 118 dB SFDR) для параметров $B_{0(n)} = 32$, $B_{\Theta(n)} = 12$, $B_S = 18$. Можно предположить, что внутренняя структура сгенерированного ядра использует большую разрядность отсчётов [в таблице $-\sin$ и \cos], чем выходная разрядность, или же использует таблицу большей глубины, что может обусловить лучший SFDR.

Полученное ядро Xilinx DDS при синтезе для FPGA также имеет значительно более высокую максимальную тактовую частоту, и при этом потребляет меньше ресурсов типа Block RAMs.

В любом случае не представляется возможным определить, в чём заключаются архитектурные отличия между сгенерированным ядром Xilinx DDS и спроектированным ядром DDS, так как исходный код для ядер Xilinx является закрытым.