

FPGA-SYSTEMS.RU

сообщество FPGA разработчиков

ПОДГОТОВИЛ:

- Владимир Шевцев



Применение System Generator для моделирования HDL-блоков на примере системы автоматической регулировки усиления

2021

Оглавление

Введение.....	5
QPSK-модуляция.....	5
Принцип работы АРУ.....	6
RTL-описание АРУ.....	7
Создание Simulink модели.....	9
Результаты моделирования.....	14
Заключение.....	17
Ссылки.....	17
Приложение.....	18

Контакты автора

Владимир Шевцев

e-mail:

vshevtsev@inbox.ru

Контакты комьюнити

сайт:

fpga-systems.ru

e-mail:

admin@fpga-systems.ru



SysGen-Привет!

В статье рассматривается возможность применения Vivado System Generator для моделирования HDL-блоков на примере системы автоматической регулировки усиления. Представлен принцип работы автоматической регулировки усиления и её реализация на Verilog. Описано поэтапное построение Simulink модели. В качестве входного воздействия рассматривается смесь QPSK сигнала и белого шума.

Подготовлено для информационно-образовательного портала FPGA / ПЛИС разработчиков FPGA-Systems.ru

Приятного прочтения.

Дата	Версия	Автор	Изменения
22.07.2021	1.0	KeisN13	Начальный релиз документа

*О найденных опечатках и замечаниях просим сообщить admin@fpga-systems.ru

Введение

В данной статье рассматривается возможность применения Vivado System Generator для моделирования HDL-блоков на примере системы автоматической регулировки усиления. Представлен принцип работы автоматической регулировки усиления и её реализация на Verilog. Описано поэтапное построение Simulink модели. В качестве входного воздействия рассматривается смесь QPSK сигнала и белого шума.

QPSK-модуляция

Quadrature phase shift keying (QPSK) – вид модуляции, широко применяемый в системах передачи данных. В цифровой связи данные передаются в виде символов, каждый из которых кодирует фиксированное количество бит. При QPSK модуляции используются четыре различных символа, которые можно представить в виде комплексных чисел. Одному символу соответствует два бита данных. Пример возможного кодирования символов представлен в таблице 1.

Таблица 1. Пример кодирования бит данных QPSK символами

Биты данных	QPSK символ
00	$1/\sqrt{2} + i \cdot 1/\sqrt{2}$
01	$1/\sqrt{2} - i \cdot 1/\sqrt{2}$
10	$-1/\sqrt{2} + i \cdot 1/\sqrt{2}$
11	$-1/\sqrt{2} - i \cdot 1/\sqrt{2}$

Набор точек, который получится, если представить символы на комплексной плоскости, называется созвездием. Созвездие для QPSK представлено на рисунке 1-а).

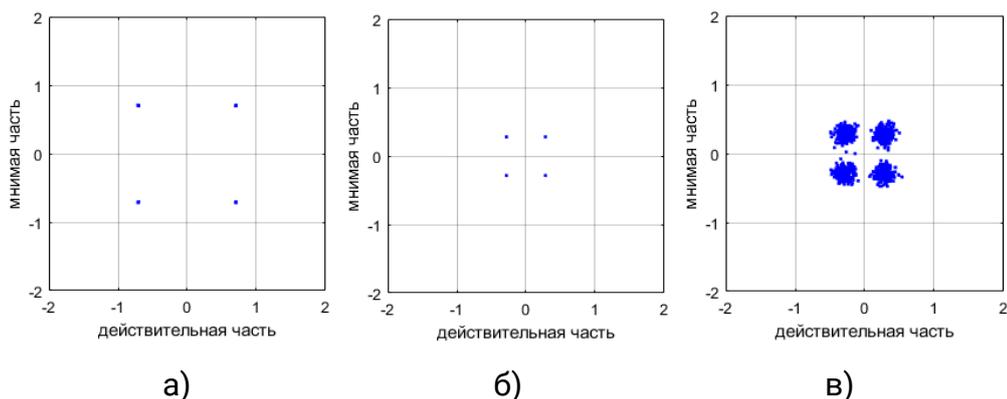


Рисунок 1. Примеры созвездий QPSK сигнала

При прохождении через канал связи символы претерпевают искажения. Ослаблению сигнала соответствует умножение значений символов на действительное число, меньшее единицы. Созвездие для QPSK при наличии затухания представлено на рисунке 1-б). Чтобы получить данные, на приемной стороне необходимо восстановить уровень сигнала. Для этого применяется система автоматической регулировки усиления.

Воздействие шума можно представить как сложение значений символов со случайным комплексным числом. Если шум белый и гауссовский, то точки на комплексной плоскости превратятся в круги, радиусы которых тем больше, чем больше мощность шума. Созвездие для QPSK при наличии шума и затухания представлено на рисунке 1-в).

Принцип работы АРУ

Автоматическая регулировка усиления (АРУ) – система, присутствующая в каждом приемном устройстве и предназначенная для поддержания постоянного уровня принимаемого сигнала. Под уровнем сигнала имеется в виду один из его параметров, таких как мощность, амплитуда, среднеквадратическое значение и т.д. В общем виде блок-схема АРУ представлена на рисунке 2. Принцип работы АРУ достаточно прост и кратко описан далее.

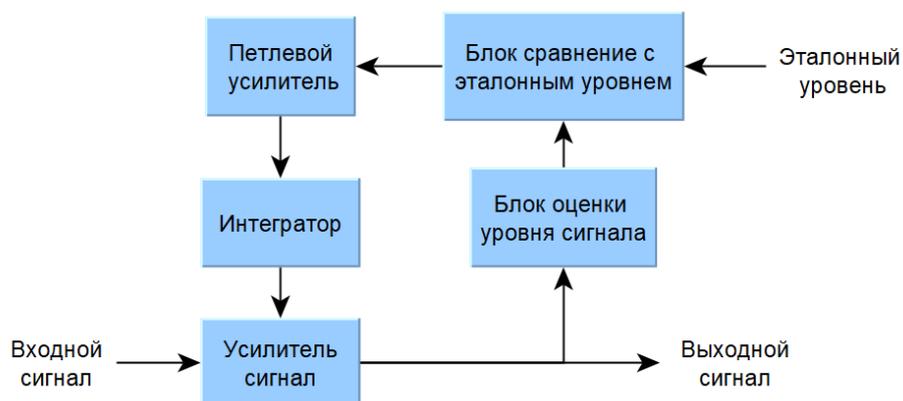


Рисунок 2. Блок-схема АРУ

Входной сигнал проходит через усилитель, коэффициент усиления которого может регулироваться. Уровень сигнала после усиления подвергается оценке. Для цифровых сигналов в качестве оценки уровня обычно выступает амплитуда принятого символа, вычисляемая по формуле

$$A = \sqrt{\operatorname{Re}(S)^2 + \operatorname{Im}(S)^2},$$

где A – уровень сигнала, S – значение символа, $\operatorname{Re}(S)$ – действительная часть, $\operatorname{Im}(S)$ – мнимая часть.

В блоке сравнения вычисляется разность полученной оценки с эталоном $D = A_{ref} - A$, где A_{ref} – эталонное значение уровня. Например, если разность положительная, значит уровень сигнала ниже эталонного значения и сигнал должен быть усилен.

Результат сравнения D умножается на петлевой коэффициент усиления μ и интегрируется. Задавая значение петлевого коэффициента усиления, можно настраивать динамические свойства системы, например, длительность переходного процесса. Для дискретных систем интегрирование заменяется суммированием, поэтому выход интегратора можно представить в виде

$$G[k + 1] = G[k] + \mu \cdot D,$$

где k – номер текущего символа, $G[k]$ – коэффициент усиления АРУ для символа с номером k .

Тогда, например, если уровень сигнала меньше эталонного значения, то разность D будет больше нуля, и коэффициент усиления АРУ G будет увеличиваться. Когда уровень сигнала сравняется с эталоном, разность D станет равной нулю, и коэффициент усиления G перестанет изменяться. Таким образом, уровень сигнала будет сохранять эталонное значение.

RTL-описание АРУ

Ниже представлен один из вариантов описания системы АРУ на языке Verilog. В данном примере основное внимание уделялось простоте реализации и читабельности кода. Никаких оптимизаций с точки зрения производительности и занимаемых ресурсов не проводилось.

Помимо стандартных портов для тактового сигнала и сигнала сброса, блок имеет вход `ce`, который в дальнейшем никак не используется. Наличие данного входа обусловлено требованиями System Generator, который ищет у блока порты, в именах

которых содержится подстроки `clk` и `ce`, и автоматически формирует для них управляющие сигналы (см. UG897 глава 6).

Регулировка уровня происходит при активном уровне на входе `enable`, иначе коэффициент усиления сигнала фиксирован и равен единице.

```

module agc(
  input clk, ce, reset,
  input enable, // '1' - включение регулировки усиления
  input signed [11:0] data_in_i, // синфазная часть входного сигнала sfix_12_9
  input signed [11:0] data_in_q, // квадратурная часть входного сигнала sfix_12_9
  output signed [11:0] data_out_i, // синфазная часть выходного сигнала sfix_12_9
  output signed [11:0] data_out_q, // квадратурная часть выходного сигнала sfix_12_9
  output reg [11:0] agc_gain // коэффициент усиления АРУ ufix_12_10
);

localparam [8:0] loop_gain = 9'h02; // петлевой коэффициент усиления ufix_9_9
localparam [11:0] ref_level = 12'h2D4; // референсный уровень сигнала ufix_12_9

reg [11:0] signal_level; // уровень сигнала ufix_12_9
reg signed [21:0] data_gained_i, data_gained_q; // сигнал после усиления sfix_22_19
  
```

Для QPSK оценку уровня сигнала можно проводить без вычисления корня и возведения в квадрат, используя только модули действительной и мнимой части значения символа. Упрощенная формула для оценки уровня имеет вид

$$A = |Re(S)| + |Im(S)|.$$

```

// вычисление уровня сигнала
always @(posedge clk)
  if (reset)
    signal_level <= 12'h000;
  else
    signal_level <= (data_out_i[11] ? -data_out_i : data_out_i) +
      (data_out_q[11] ? -data_out_q : data_out_q);
  
```

Реализация интегратора и блока сравнения также была упрощена. Вычисление коэффициента усиления АРУ `agc_gain`, выполняется с помощью выражения

$$G[k + 1] = \begin{cases} G[k] + \mu & \text{при } A_{ref} > A, \\ G[k] - \mu & \text{при } A_{ref} \leq A. \end{cases}$$

Эталонный уровень A_{ref} рассчитан для значений символов, представленных в таблице 1, и равен $\sqrt{2}$. Эталонный уровень A_{ref} задается константой `ref_level`, а петлевой коэффициент усиления μ - константой `loop_gain`.

```
// коэффициент усиления АРУ
always @(posedge clk)
  if (reset | !enable)
    agc_gain <= 12'h400; // при сбросе и выключеном режиме коэффициент
усиления равен единице
  else
    if (ref_level > signal_level)
      agc_gain <= agc_gain + loop_gain;
    else
      agc_gain <= agc_gain - loop_gain;
```

Регулировка уровня осуществляется умножением действительной и мнимой части входного сигнала на коэффициент усиления АРУ.

```
// усиление входного сигнала
always @(posedge clk) begin
  data_gained_i <= data_in_i * $signed({1'b0, agc_gain});
  data_gained_q <= data_in_q * $signed({1'b0, agc_gain});
end

// формирование выходных сигналов
assign data_out_i = data_gained_i[21:10],
       data_out_q = data_gained_q[21:10];

endmodule
```

Создание Simulink модели

После создания новой Simulink модели необходимо настроить параметры симулятора. По умолчанию Simulink при численном решении системы дифференциальных уравнений, полученной из модели, динамически подстраивает шаг дискретизации. Для моделирования HDL-блоков это не подходит, так как период тактового сигнала постоянный. Максимальный шаг дискретизации удобно установить равным единице, так как тогда одной секунде модельного времени будет соответствовать один такт работы блока. Окно настроек симулятора представлено на рисунке 3.

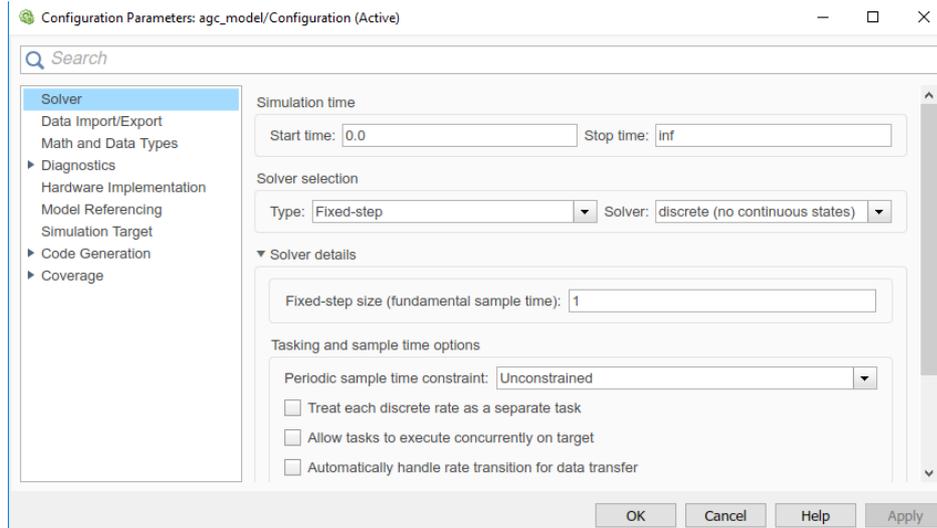


Рисунок 3. Окно настроек симулятора

Для создания QPSK сигнала и имитации канала используются следующие блоки:

- **Random Integer Generator** – генерирует целые случайные числа от нуля до трех;
- **QPSK Modulator Baseband** – преобразует полученные числа в QPSK символы в соответствии с таблицей 1;
- **Gain** – умножает значение символа на заданную величину, имитируя затухание канала;
- **AWGN Channel** – добавляет белый шум заданной мощности к сигналу.

Окна параметров блоков, настройки которых отличаются от настроек по умолчанию, представлены в приложении.

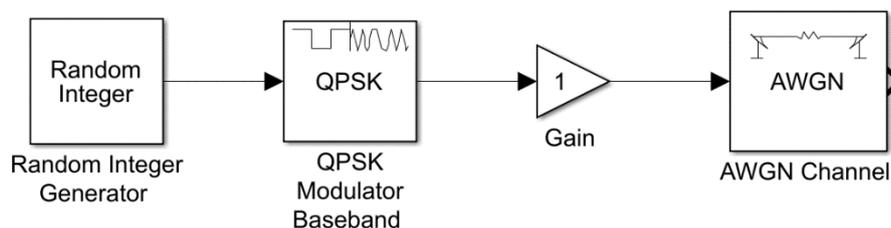


Рисунок 4. Модель после добавления блоков формирования QPSK сигнала

Далее к модели добавляются интерактивные блоки и источники управляющих сигналов:

- **Step** – формирует активный уровень сигнала сброса в начале моделирования;
- **Constant** – источник для сигнала включения АРУ;

- **Slider** – привязывается к блоку Gain и позволяет интерактивно задавать ослабление сигнала в канале;
- **Switch** – привязывается к блоку Constant и дает возможность интерактивно включать и выключать АРУ.

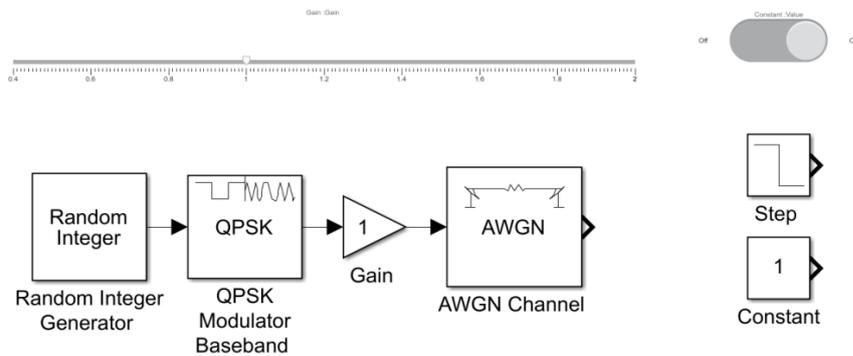


Рисунок 5. Модель после добавления интерактивных блоков и источников управляющих сигналов

После этого в модель добавляются блоки отображения результатов и вспомогательные блоки:

- **Complex to Real-Imag** – раскладывает комплексное число на мнимую и действительную части;
- **Real-Imag to Complex** – формирует комплексное число из мнимой и действительной части;
- **Constellation Diagram** – названы в модели **Input Signal** и **Output Signal**, отображают созвездия входного и выходного QPSK сигналов;
- **Display** – выводит текущее значение коэффициента усиления АРУ;
- **Scope** – позволяет наблюдать изменение коэффициента усиления АРУ во времени.

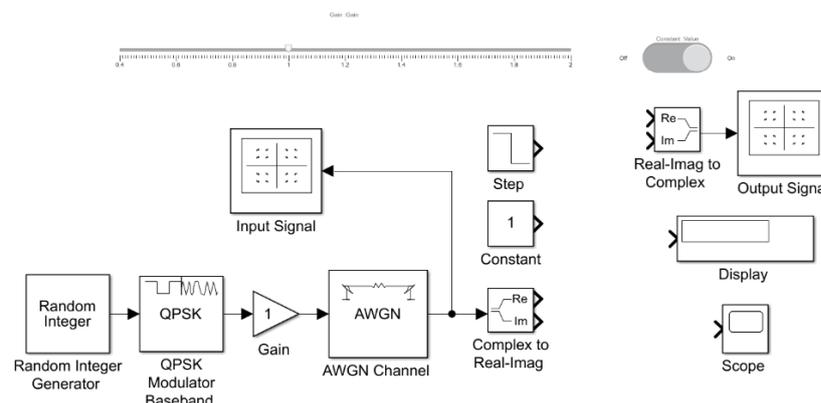


Рисунок 6. Модель после добавления блоков отображения результатов

На этом этапе в модель внесены все Simulink блоки. Для передачи данных между блоками Simulink и блоками System Generator необходимо добавить блоки **Gateway In** и **Gateway Out**. Настройки блоков **Gateway In**, управляющих портами АРУ `reset` и `enable`, представлены на рисунке 7-а). Настройки блоков для управления портами `data_in_i` и `data_in_q` показаны на рисунке 7-б). Основные параметры – это формат входного сигнала, квантование, обработка переполнения и шаг дискретизации.

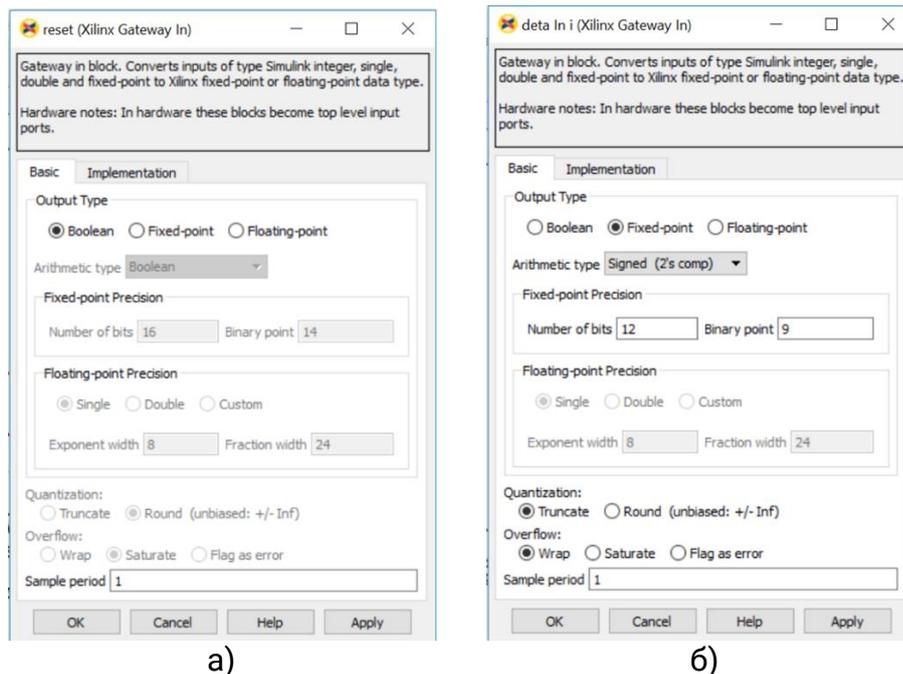


Рисунок 7. Окна настроек блоков Gateway In

Также необходимо добавить блок **System Generator Token**, с помощью которого выполняется настройка параметров синтеза HDL-кода из модели (см. UG897 глава 3). Помимо этого, **System Generator Token** отвечает за формирование тактовых сигналов и сигналов `clock enable`. Если System Generator используется только для моделирования готовых HDL-блоков, то все настройки **System Generator Token** можно оставить по умолчанию.

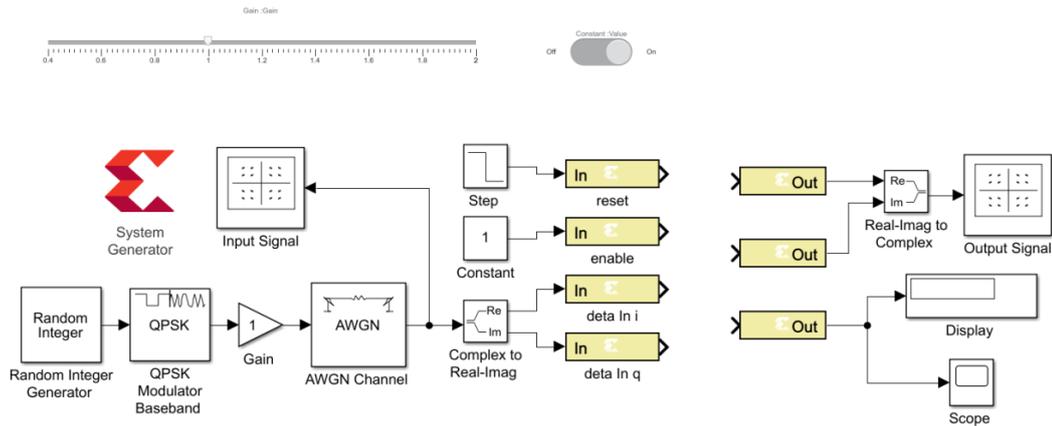


Рисунок 8. Модель после добавления блоков System Generator

Для присоединения HDL-модулей к Simulink модели используется блок, который называется **Black Box**. После добавления данного блока автоматически запустится мастер его конфигурации, который попросит указать HDL-файл.

После этого будет сгенерирован файл **"имя_модуля"_config.m**, настраивающий внешний вид и параметры **Black Box**. По умолчанию всем выходным портам **Black Box** присваивается формат целых числе без знака. Задать требуемые параметры портов можно, вручную отредактировав конфигурационный m-файл, так как показано ниже (см. UG897 глава 6).

```
data_out_i_port = this_block.port('data_out_i');
data_out_i_port.setType('Fix_12_9');
data_out_q_port = this_block.port('data_out_q');
data_out_q_port.setType('Fix_12_9');
agc_gain_port = this_block.port('agc_gain');
agc_gain_port.setType('UFix_12_10');
```

Black Box не имеет входных портов **clk** и **ce**, так как их управляющие сигналы формирует **System Generator Token**. На этом создание модели завершено. Полностью собранная модель представлена на рисунке 9.

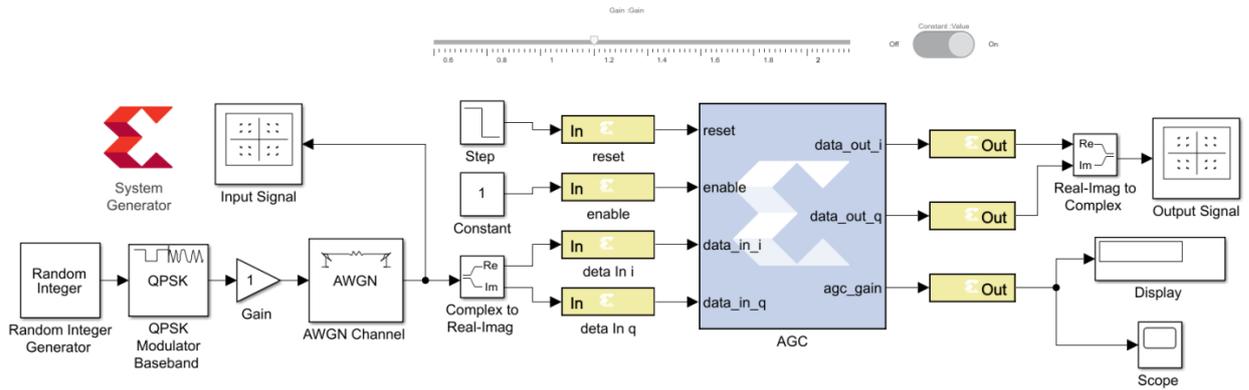


Рисунок 9. Модель после добавления HDL-модуля

Результаты моделирования

Ниже представлены результаты моделирования АРУ при различных уровнях сигнала на входе. На рисунке 10 представлены созвездия сигналов на входе и выходе, элементы интерактивного управления и блок, отображающий коэффициент усиления АРУ. При прохождении через канал связи сигнал умножается на 0.6, что соответствует ослаблению на 4.4 дБ. Система АРУ выключена, поэтому ее коэффициент равен единице. Красными отметками обозначены точки, соответствующие таблице 1.

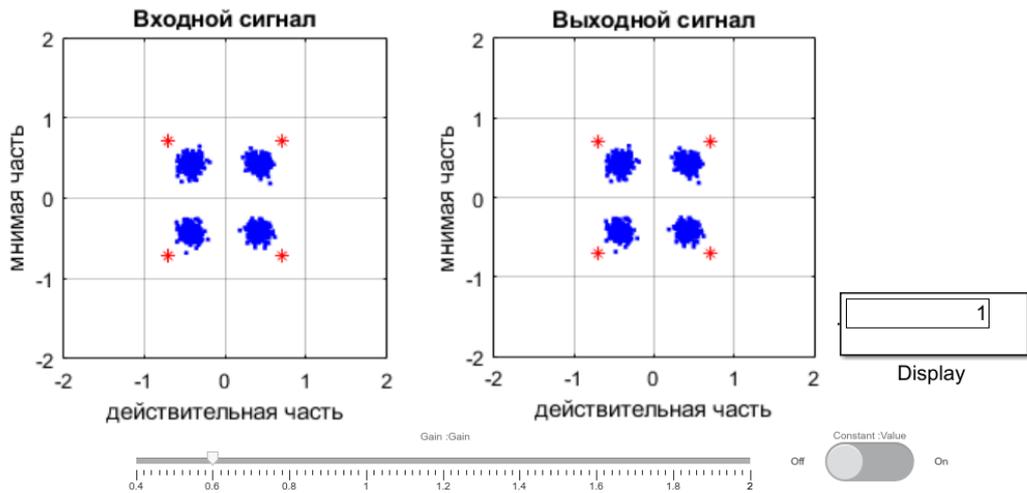


Рисунок 10. АРУ выключена, сигнал ослаблен на 4.4 дБ

На рисунке 11 представлены те же элементы, но теперь сигнал умножается на 1.6 или, что тоже самое, усиливается на 4 дБ. Система АРУ по-прежнему выключена.

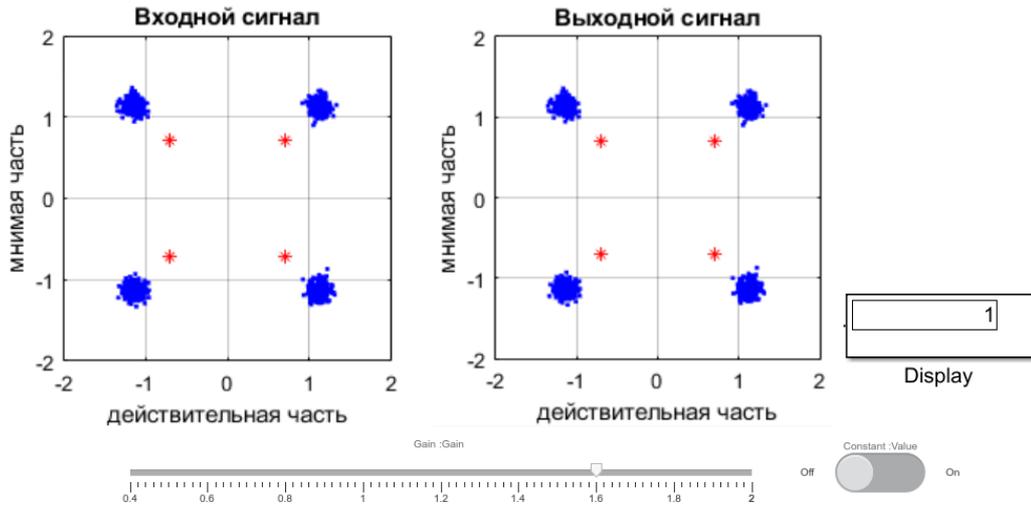


Рисунок 11. АРУ выключена, сигнал усилен на 4 дБ

На рисунках 12 и 13 показаны результаты моделирования в случае, когда система АРУ включена. Можно увидеть, что центры окружностей совпадают с точками из таблицы 1. Также умножив коэффициент усиления АРУ на коэффициент усиления канала, получим значение близкое к единице, что соответствует корректной работе АРУ.

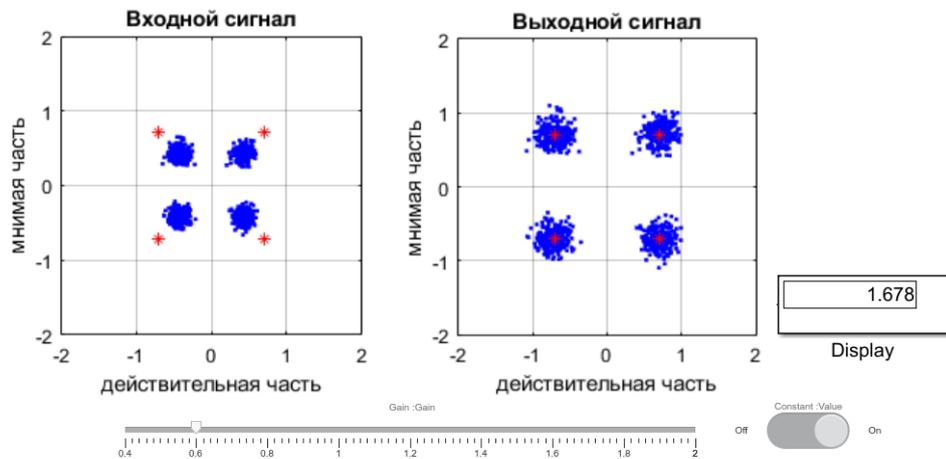


Рисунок 12. АРУ включена, сигнал ослаблен на 4.4 дБ

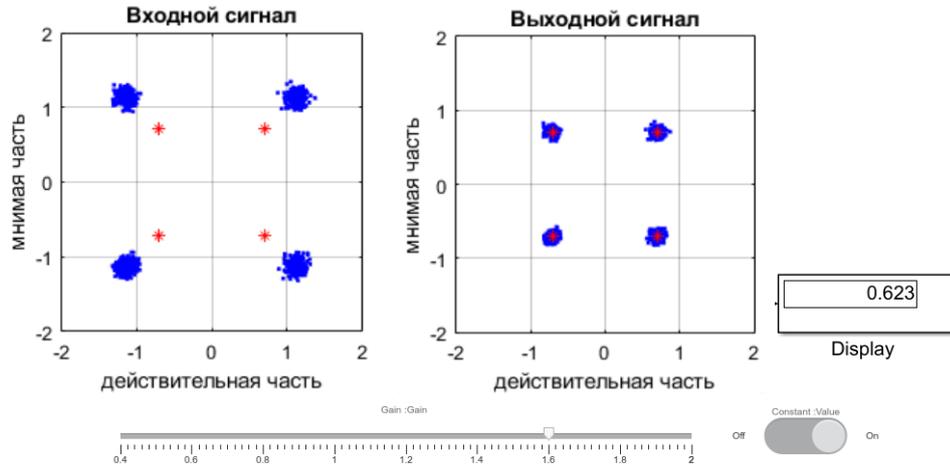
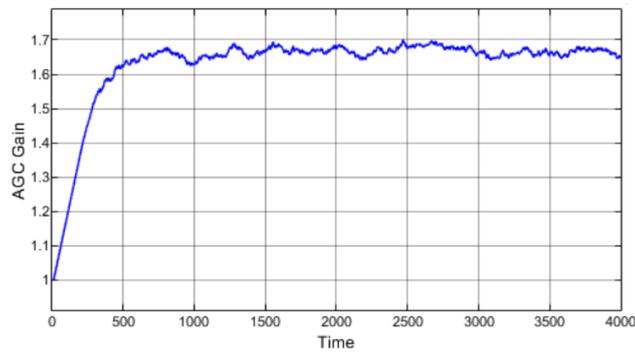
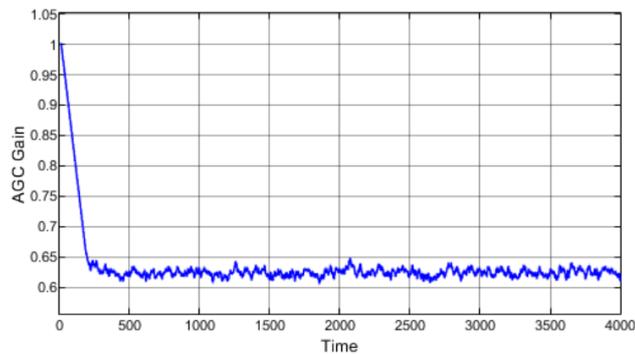


Рисунок 13. АРУ включена, сигнал усилен на 4 дБ

В заключении на рисунках 14 а) и б) с помощью блока Scope представлен переходной процесс, отражающий изменение коэффициента усиления АРУ от времени. Рисунок а) соответствует случаю ослабления сигнала на 4.4 дБ, рисунок б) – случаю усиления на 4 дБ.



а)



б)

Рисунок 14. Зависимость коэффициента усиления АРУ от времени в тактах

Заключение

При моделировании HDL-модулей в ряде случаев бывает полезно применять System Generator, так как это открывает доступ к широкому набору блоков для создания входных сигналов и их визуализации. Для задач из области связи удобно использовать возможности Simulink для имитации каналов передачи с шумами, помехами и замираниями. Также, если при разработке тракта передачи данных была создана его Simulink модель, можно заменить отдельные узлы блоками, реализованными на HDL, и исследовать поведение данных блоков в рамках всей системы.

Ссылки

1. https://www.xilinx.com/support/documentation/sw_manuals/xilinx2015_1/ug897-vivado-sysgen-user.pdf - user guide по System Generator (UG 897).
2. https://www.xilinx.com/support/documentation/sw_manuals/xilinx2020_1/ug948-vivado-sysgen-tutorial.pdf - tutorial по System Generator (UG 948).
3. https://github.com/VSHEV92/AGC_System_Generator_Model - модель и исходные коды, представленные в статье.

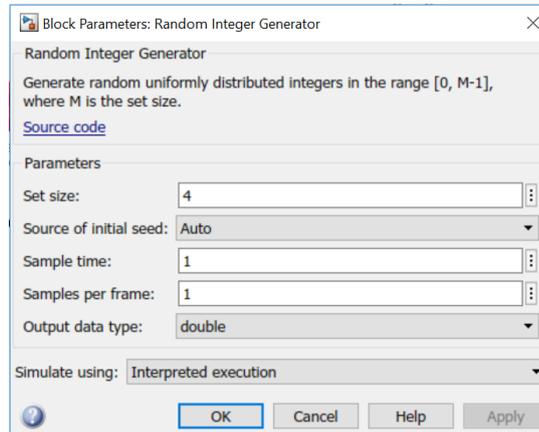
[Мотивировать автора](#)

[Поддержать проект FPGA-Systems.ru](#)

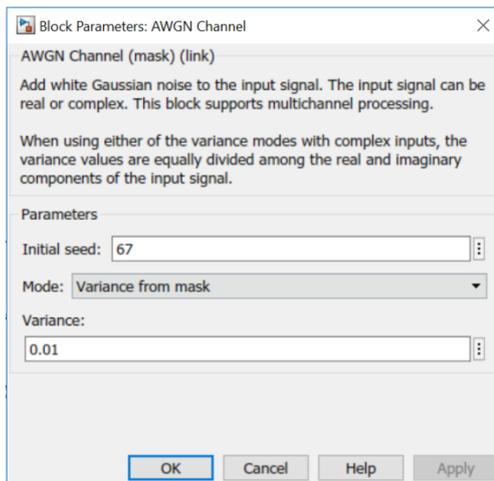
[Оставить комментарий/отзыв](#)

Приложение

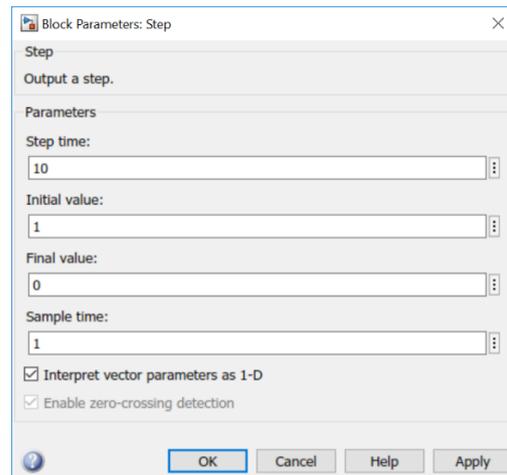
На рисунке П1 представлены окна параметров Simulink блоков, настройки которых отличаются от настроек по умолчанию.



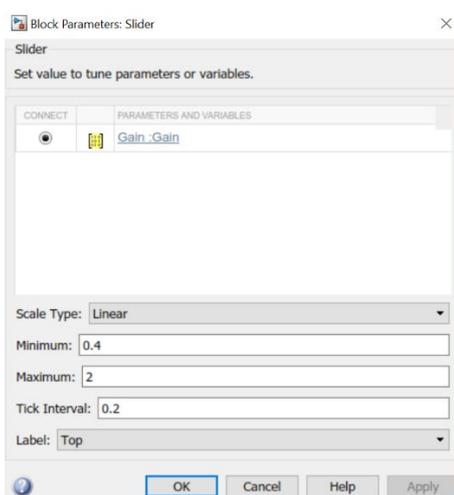
а)



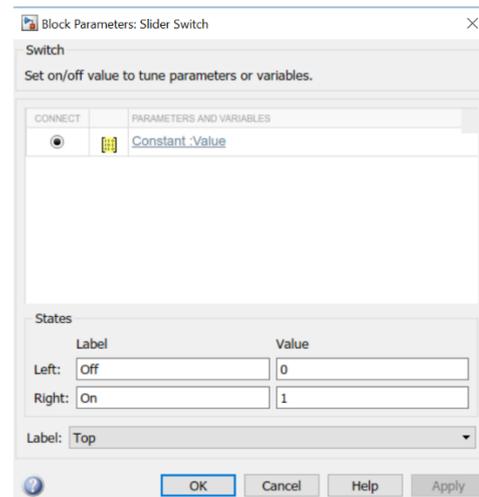
б)



в)



г)



д)

Рисунок П1. Окна настройки параметров блоков модели