

*Солдатенков Е. В.*

*магистрант*

*Московский институт электронной техники*

## **ИСПОЛЬЗОВАНИЕ СТАТИСТИЧЕСКИХ РАСПРЕДЕЛЕНИЙ В ВЕРИФИКАЦИИ ЦИФРОВЫХ БЛОКОВ**

*Аннотация: В статье рассматриваются основные аспекты использования распределительных ограничений (distribution constraints) в языке описания аппаратуры SystemVerilog. Автор объясняет, как в рамках теста с помощью ограничений задавать вероятностное распределение значений, благодаря чему можно существенно повысить гибкость тестов и эффективность верификации. В статье подробно описаны синтаксис и семантика базовых распределительных ограничений, а также приведены примеры их применения в моделировании и тестировании. Вместе с тем исследованы результаты рандомизации при комбинировании различных типов ограничений, а также получена взаимосвязь ограничений и результирующей функции распределения вероятности. Статья будет полезна инженерам-верификаторам, стремящимся улучшить качество и скорость проведения верификации.*

*Ключевые слова: System Verilog, constraints, dist, распределение вероятностей, распределение Гаусса, верификация.*

*Soldatenkov E. V.*

*Master student*

*Moscow Institute of Electronic Engineering*

*Use of statistical distributions in verification of digital blocks*

*Annotation: The article discusses the main aspects of using distribution constraints in the SystemVerilog hardware description language. The author explains how to specify probability distribution of values within a test by means of constraints, thanks to which test flexibility and verification efficiency can be significantly increased. The paper describes in detail the syntax and semantics*

*of basic distributional constraints and gives examples of their application in modelling and testing. At the same time, the results of randomisation when combining different types of constraints are investigated, and the relationship between the constraints and the resulting probability distribution function is derived. The paper will be useful to verification engineers seeking to improve the quality and speed of verification.*

*Keywords: System Verilog, constraints, dist, probability distribution, Gaussian distribution, verification.*

## I. Введение

Основным подходом в верификации сегодня является рандомизированное тестирование с ограничениями (Constraint Random Verification), сутью которого является подача рандомизированных данных в заданном диапазоне. Для определения диапазона допустимых значений используются SystemVerilog Constraints, как правило представляющие собой логические выражения. Вызов функции рандомизации запускает решатель ограничений, который подбирает случайное значение, удовлетворяющее всем соответствующим логическим выражениям.

Порядок решения ограничений может быть регламентирован инженером-верификатором с помощью solve before ограничения и dist ограничения. Последний будет рассмотрен далее.

## II. Методы исследования

В данной статье будут рассмотрены методы задания функции распределения вероятностей и эксперимент, заключающийся в проведении ста тысяч циклов рандомизации для проверки соответствия результатов эксперимента ожидаемому распределению.

## III. Основная часть

Функция распределения вероятностей – это функция, дающая вероятность того, что случайная дискретная величина принимает значение аргумента. Формальное определение выглядит следующим образом:

$$p_x(x) = P(X=x),$$

для  $-\infty < x < +\infty$ , где  $P$  – вероятностная мера.

Что бы определить такую функцию для переменной в языке SystemVerilog используют ограничение `dist`, обладающее следующим синтаксисом:

```
constraint value_c {value dist { expressions }; }
```

в котором `expressions` – выражения, определяющие вероятность появления соответствующего значения, например:

```
constraint value_c {value dist {3:/4, [5:8] :/ 7}; }
```

Оператор «`:/`» определяет вес для одного значения и `вес/n` (количество значений) для диапазона значений длиной `n`.

Оператор «`:=`» определяет вес для каждого значения в выражении.

С помощью представленных конструкций напишем ограничения для переменной `value` (рисунок 1 и рисунок 2):

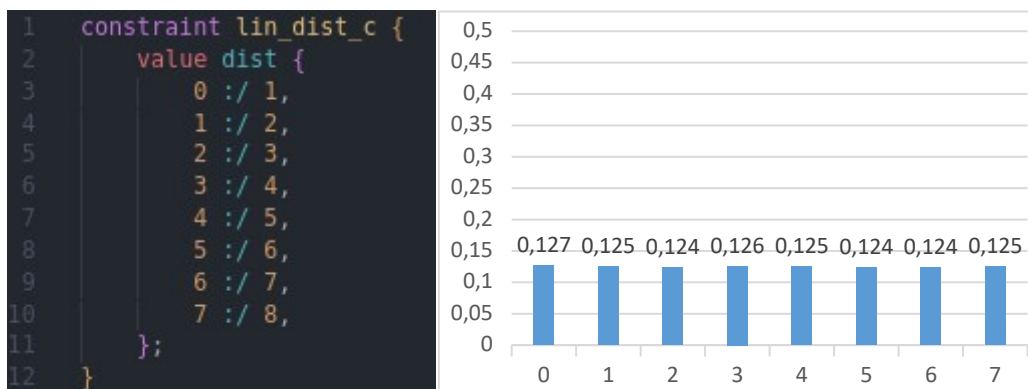


Рисунок 1. Функция линейного распределения

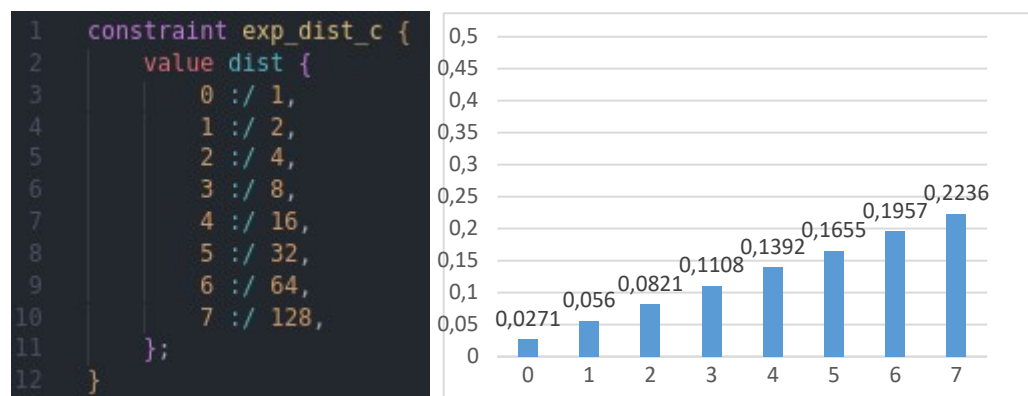


Рисунок 2. Экспоненциальная функция распределения

В обоих этих случаях вероятность того, что решатель ограничений выберет заданное значение, равна весу этого значения, делённому на сумму весов. Например, вероятность выбора значения 0 в нашем линейном распределении равна  $1/(1+2+\dots+8) = 1/38 = 0.02631$ .

Для создания более сложных ограничений можно комбинировать несколько constraint блоков, однако стандарт SystemVerilog не дает четкого определения обработки комбинации ограничений.

Для того, чтобы найти связь результирующей функции распределения вероятности величины с её ограничениями, дополним их равномерным распределением и проведем исследование в симуляторе xselium, направленное на получение графиков  $p(x)$  с различными комбинациями ограничений (рисунок 3).

#### IV. Выводы и результаты исследования

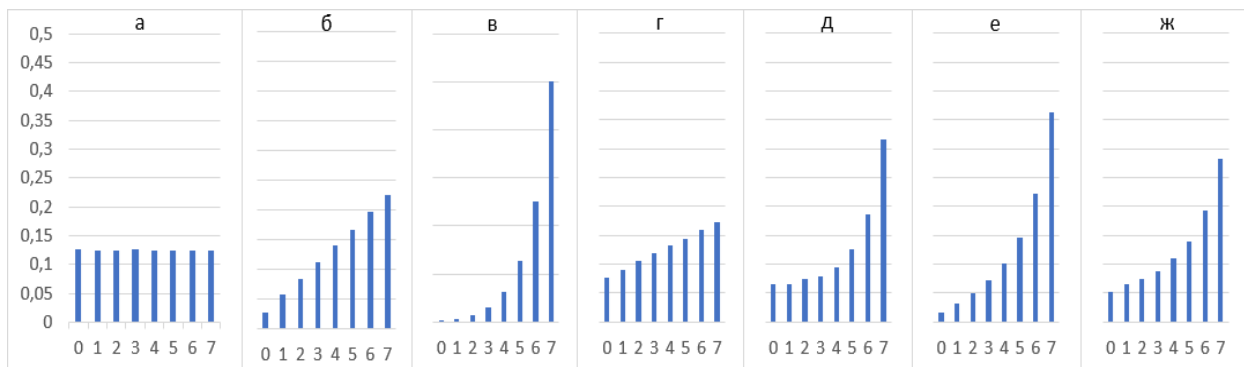


Рисунок 3. Результаты эксперимента: а - равномерное распределение, б - линейное распределение, в - экспоненциальное распределение, г - равномерное и линейное распределение, д - равномерное и экспоненциальное распределение, е - линейное и экспоненциальное, ж - равномерное, линейное и экспоненциальное распределение

По результатам проведенных экспериментов была выведена формула:

$$p(x) = \frac{\sum_{n=1}^N \frac{weight_n(x)}{\sum_{m=-\infty}^{\infty} (weight(m))}}{N}$$

где N – количество блоков ограничений.

Формула удовлетворяет результатам эксперимента в случае, когда вес каждого значения ненулевой. Проведем дополнительный эксперимент с значением веса 1 и 6 равным нулю (рисунок 4):

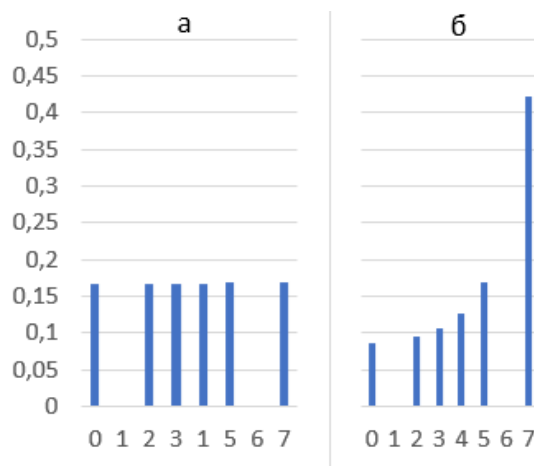


Рисунок 4. Результат эксперимента: а – равномерное распределение с нулевым весом чисел 1 и 6, б – равномерное с нулевыми весами и экспоненциальное распределение

$$p(x) = \frac{\sum_{n=1}^N \frac{weight_n(x) * (\prod_{i=1}^N weight_n(i) > 0)}{\sum_{m=-\infty}^{\infty} (weight(m) * (\prod_{i=1}^N weight_n(i) > 0))}}{N}$$

В формуле учитывается комбинированное действие ограничений на переменную, а также случай, когда вес переменной равен нулю.

## V. Заключение

В данном исследовании рассмотрены методы задания функции распределения случайной дискретной величины в языке описания аппаратуры SystemVerilog, а также получена формула, отражающая функцию распределения вероятности для величины, описанной в блоке constraint. С помощью этой формулы можно задавать любое желаемое распределение и достичь более высокой эффективности верификации.

### Использованные источники:

1. Stewart, William J. (2011). Probability, Markov Chains, Queues, and Simulation: The Mathematical Basis of Performance Modeling;

2. Интернет-ресурс по современным методологиям верификации и языкам описания аппаратуры. - <https://vlsiverify.com/system-verilog/>;
3. Hardware Design Verification: Simulation and Formal Method-Based Approaches, William K. Lam, Prentice Hall PTR, March 03, 2005;
4. IEEE Std 1800-2017 “IEEE Standard for SystemVerilog–Unified Hardware Design, Specification, and Verification Language”, 2017;