

 SURFACE VEHICLE RECOMMENDED PRACTICE	J2748	ISSUED OCT2006
	Issued	2006-10
VHDL-AMS Statistical Analysis Packages		

RATIONALE

Not applicable.

FOREWORD

The SAE J2748 is based on the IEEE Std. 1076.1-1999, the VHDL-AMS language definition. It provides support for modeling of statistical information that is relevant to automotive systems.

INTRODUCTION

This document specifies the interface and the behavior of the VHDL-AMS packages for use in modeling statistical behavior. These packages are useful in defining the statistical variation of parameters of electrical, electronic, and mechatronic components and sub-systems. These can then be used with simulation tools to analyze the performance and reliability of systems composed of these components and sub-systems.

SAE Technical Standards Board Rules provide that: "This report is published by SAE to advance the state of technical and engineering sciences. The use of this report is entirely voluntary, and its applicability and suitability for any particular use, including any patent infringement arising therefrom, is the sole responsibility of the user."

SAE reviews each technical report at least every five years at which time it may be reaffirmed, revised, or cancelled. SAE invites your written comments and suggestions.

Copyright © 2006 SAE International

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of SAE.

TO PLACE A DOCUMENT ORDER: Tel: 877-606-7323 (inside USA and Canada)
Tel: 724-776-4970 (outside USA)
Fax: 724-776-0790
Email: CustomerService@sae.org
<http://www.sae.org>

SAE WEB ADDRESS:

Full document details: [SAE J2748_200610](#)

TABLE OF CONTENTS

1.	SCOPE.....	3
1.1	Purpose.....	3
2.	REFERENCES.....	3
2.1	Applicable Publications	3
2.1.1	IEEE Publications.....	3
2.2	Related Publications	3
2.2.1	SAE Publications.....	3
3.	PACKAGE DEFINITIONS	4
3.1	STATISTICS	4
3.1.1	Types	4
3.1.2	Attributes	5
3.1.3	Auxiliary Functions	6
3.1.4	Regular Distribution Functions.....	9
3.1.5	Standard Distribution Functions.....	35
3.2	STATISTICS_CONTROL.....	43
3.2.1	Types	43
3.2.2	Functions.....	44
4.	USING THE STATISTICAL PACKAGES.....	45
4.1	Using Statistical Functions in VHDL-AMS Models	45
4.1.1	Library and Package Names.....	45
4.1.2	Specifying a distribution for a constant within a model.....	45
4.1.3	Assigning a distribution to a generic in a generic map	46
4.1.4	Creating a model that includes generic constants for the distribution	47
4.1.5	Correlation.....	47
4.2	Simulation Control.....	48
4.2.1	Switching Between Nominal and Statistical Analysis	48
4.2.2	Seed Management.....	49
4.2.3	Performing a Statistical Analysis on a Portion of the Design.....	49
4.2.4	Extending a Monte Carlo Experiment	49
4.3	Defining a New Statistical Distribution	49
4.3.1	Naming Conventions.....	49
4.3.2	Standard Distribution Functions.....	50
4.3.3	Attributes	50
4.3.4	Specific Requirements/Rules.....	50
4.4	Example	51
5.	PACKAGE RATIONALE	52
5.1	Package Architecture	52
5.1.1	Package STATISTICS_CONTROL	52
5.1.2	Package STATISTICS	52
5.1.3	Design Library	52
5.2	Statistical Distribution Functions	52
5.2.1	Regular Distribution Functions.....	53
5.2.2	Standard Distribution Functions.....	56
6.	NOTES	56
6.1	Key Words.....	56

1. SCOPE

This document specifies the interface and the behavior of the VHDL-AMS packages for use in modeling statistical behavior. These packages are useful in defining the statistical variation of parameters of electrical, electronic, and mechatronic components and sub-systems. These can then be used with simulation tools to analyze the performance and reliability of systems composed of these components and sub-systems.

Providing a standard definition of the package interfaces and their behavior is intended to facilitate the exchange of models between component and system manufacturers and the use of different CAE simulation tools.

The SAE statistical package supports the statistical modeling of design parameters subject to tolerances for designs described using the VHDL or VHDL-AMS languages. The performance of a design that uses this package to model parameter tolerances may be analyzed by a Monte Carlo simulation, which consists of multiple simulation runs of the design, each run with a different set of parameter values according to their statistical distributions. The Monte Carlo simulation will yield an estimate of the behavior of the design subject to the parameter tolerances. Additionally, the package may be used to perform an Extreme Value Analysis of a design, that is, an analysis that computes the worst case behavior of the design.

The SAE statistical package does not support other statistical modeling needs, for example the modeling of statistical variations of the arrival of events in a queue, or any other situation where a value exhibits a statistical behavior during a single simulation.

1.1 Purpose

The VHDL-AMS Statistical Analysis Packages define a common set of standards in defining and using statistical variation of parameters on models written in VHDL-AMS for automotive electrical, electronic, and mechatronic components and sub-systems. The definition supports the use of predefined and user-defined distributions.

2. REFERENCES

2.1 Applicable Publications

The following publications form a part of this specification to the extent specified herein. Unless otherwise specified, the latest issue of SAE publications shall apply.

2.1.1 IEEE Publications

Available from ANSI, 25 West 43rd Street, New York, NY 10036-8002, Tel: 212-642-4900, www.ansi.org or <http://standards.ieee.org>.

IEEE Std 1076.1-1999 IEEE Standard VHDL Analog and Mixed Signal Extensions

2.2 Related Publications

The following publications are provided for information purposes only and are not a required part of this document.

2.2.1 SAE Publications

Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001, Tel: 877-606-7323 (inside USA and Canada) or 724-776-4970 (outside USA), www.sae.org.

SAE J2546 Model Specification Process Standard

3. PACKAGE DEFINITIONS

There are two packages which are used to define the statistical behavior: `STATISTICS_CONTROL` and `STATISTICS`. These packages are located in the library with the logical name `VHDL_UTILITY`. A reference implementation of these packages can be found on the website at <http://links.sae.org/J2748>.

3.1 STATISTICS

The `STATISTICS` package defines the types, attributes, auxiliary functions, regular and standard distributions functions. More information can be found in Section 5.2 on Statistical Distribution Functions.

3.1.1 Types

The following types are defined for use with the package.

3.1.1.1 REAL_XY_PAIR

Type **REAL_XY_PAIR**

Declaration

```
type REAL_XY_PAIR is record
    X: REAL;
    Y: REAL;
end record;
```

Record type holding abscissa and ordinate of a point in 2D

Fields

Name	Type	Description
X	REAL	abscissa value
Y	REAL	ordinate value

3.1.1.2 REAL_XY_TABLE

Type **REAL_XY_TABLE**

Declaration

```
type REAL_XY_TABLE is array (NATURAL range <>) of REAL_XY_PAIR;
```

Array type holding a sequence of `REAL_XY_PAIR`

3.1.1.3 INTEGER_XY_PAIR

Type **INTEGER_XY_PAIR**

Declaration `type INTEGER_XY_PAIR is record
 X: INTEGER;
 Y: REAL;
 end record;`

Record type holding abscissa and ordinate of a point in 2D

Fields

Name	Type	Description
X	INTEGER	abscissa value
Y	REAL	ordinate value

3.1.1.4 INTEGER_XY_TABLE

Type **INTEGER_XY_TABLE**

Declaration `type INTEGER_XY_TABLE is array (NATURAL range <>) of
 INTEGER_XY_PAIR;`

Array type holding a sequence of `INTEGER_XY_PAIR`

3.1.1.5 STAT_MODE_TYPE

Type **STAT_MODE_TYPE**

Declaration `subtype STAT_MODE_TYPE is
 VHDL_UTILITY.STATISTICS_CONTROL.STAT_CONTROL_SUBTYPE;`

Enumeration type defining the statistical modes supported by the package

3.1.2 Attributes

The following attributes are used with the regular distribution functions.

Attribute **STAT_POSITION_NOMINAL**
 STAT_POSITION_TOLERANCE
 STAT_POSITION_MINVALUE
 STAT_POSITION_MAXVALUE

Declaration `attribute STAT_POSITION_NOMINAL: INTEGER;
 attribute STAT_POSITION_TOLERANCE: INTEGER;
 attribute STAT_POSITION_MINVALUE: INTEGER;
 attribute STAT_POSITION_MAXVALUE: INTEGER;`

Position of `NOMINAL`, `TOLERANCE`, `MINVALUE`, `MAXVALUE` in formals. These attributes are used to give semantics to the formal arguments of the regular distribution functions

3.1.3 Auxiliary Functions

The following auxiliary functions are used with the package.

3.1.3.1 STAT_MODE

Function	STAT_MODE
Declaration	<code>pure function STAT_MODE return STAT_MODE_TYPE;</code> Function returning the current statistical mode
Return	<code>STAT_MODE_TYPE</code> The current statistical mode (<code>STAT_NOMINAL</code> or <code>STAT_STATISTICAL</code>)

3.1.3.2 NORMALIZE_TABLE

There are 2 forms of the NORMALIZE_TABLE functions with different arguments.

Function **NORMALIZE_TABLE**

Declaration `pure function NORMALIZE_TABLE(
 TBL: REAL_XY_TABLE;
 NOMINAL: REAL;
 TOL: REAL
) return REAL_XY_TABLE;`

Function to normalize a table defining a probability density function or a cumulative density function

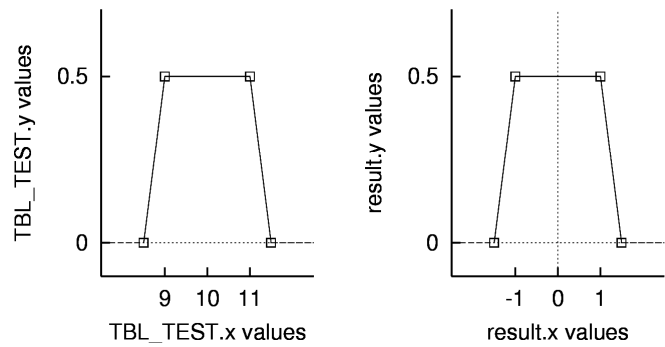
Arguments

Name	Type	Description
TBL	REAL_XY_TABLE	Table specifying the PDF or CDF. The abscissa values must increase monotonically
NOMINAL	REAL	Nominal value. Must be non-zero
TOL	REAL	Tolerance of the value. Must be > 0

Return REAL_XY_TABLE

A table normalized to use with the PWL_PDF or PWL_CDF functions

Example `NORMALIZE_TABLE(TBL=>TBL_TEST,NOMINAL=>10.0,TOL=>0.1)`



Function **NORMALIZE_TABLE**

Declaration

```
pure function NORMALIZE_TABLE(  
    TBL:      REAL_XY_TABLE;  
    NOMINAL:  REAL;  
    MINVALUE: REAL;  
    MAXVALUE: REAL  
) return  REAL_XY_TABLE;
```

Function to normalize a table defining a probability density function or a cumulative density function

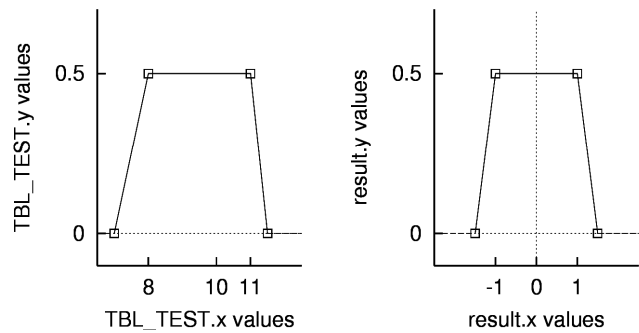
Arguments

Name	Type	Description
TBL	REAL_XY_TABLE	Table specifying the PDF or CDF. The abscissa values must increase monotonically
NOMINAL	REAL	Nominal value
MINVALUE	REAL	Minimum value. Must be < NOMINAL
MAXVALUE	REAL	Maximum value. Must be > NOMINAL

Return REAL_XY_TABLE

A table normalized to use with the PWL_PDF or PWL_CDF functions

Example NORMALIZE_TABLE(TBL_TEST,10.0,8.0,11.0)



3.1.4 Regular Distribution Functions

Regular distribution functions are statistical distribution functions that are intended to be called in a design to define the value of a design parameter. More information can be found in Section 5.2.1 on Regular Distribution Functions.

3.1.4.1 UNIFORM

There are 4 forms of the UNIFORM distribution with different arguments. These functions also have attributes defining position arguments.

Function **UNIFORM**

Declaration

```
impure function UNIFORM(
    NOMINAL: REAL;
    TOL: REAL;
    MODE: STAT_MODE_TYPE := STAT_MODE
) return REAL;
```

Uniform distribution with `REAL` nominal value and tolerance

Arguments

Name	Type	Default	Description
NOMINAL	REAL		Nominal value
TOL	REAL		Tolerance of the value. Must be ≥ 0
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value in the range between the minimum and maximum values. The default is determined based on the value of constant <code>STAT_CONTROL</code> in package <code>STATISTICS_CONTROL</code>

Attributes

Name	Value
<code>STAT_POSITION_NOMINAL</code>	1
<code>STAT_POSITION_TOLERANCE</code>	2

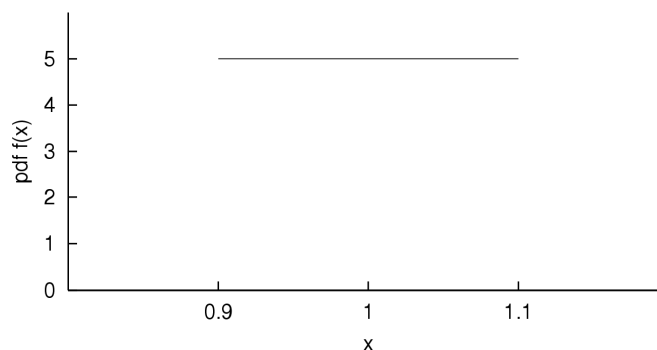
Return

`REAL`

See argument `MODE`

Example

`UNIFORM(NOMINAL => 1.0, TOL => 0.1)`



Function **UNIFORM**

Declaration `impure function UNIFORM(
 NOMINAL: TIME;
 TOL: REAL;
 MODE: STAT_MODE_TYPE := STAT_MODE
) return TIME;`

Uniform distribution with `TIME` nominal value and tolerance

Arguments

Name	Type	Default	Description
NOMINAL	TIME		Nominal value
TOL	REAL		Tolerance of the value. Must be ≥ 0
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value in the range between the minimum and maximum values. The default is determined based on the value of constant <code>STAT_CONTROL</code> in package <code>STATISTICS_CONTROL</code>

Attributes

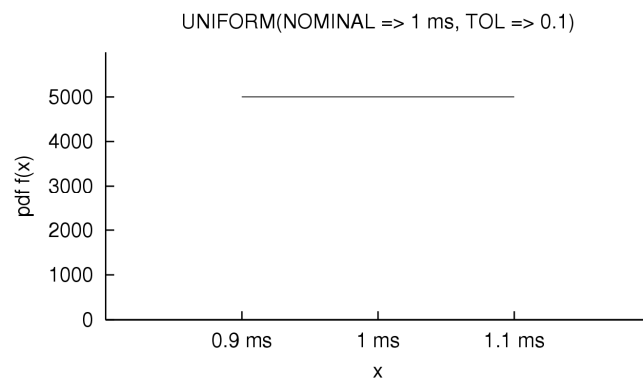
Name	Value
<code>STAT_POSITION_NOMINAL</code>	1
<code>STAT_POSITION_TOLERANCE</code>	2

Return

`TIME`

See argument `MODE`

Example



Function **UNIFORM**

Declaration `impure function UNIFORM(
 NOMINAL: REAL;
 MINVALUE: REAL;
 MAXVALUE: REAL;
 MODE: STAT_MODE_TYPE := STAT_MODE
) return REAL;`

Uniform distribution with `REAL` nominal value and bounds

Arguments

Name	Type	Default	Description
NOMINAL	REAL		Nominal value
MINVALUE	REAL		Minimum value. Must be $\leq$ NOMINAL
MAXVALUE	REAL		Maximum value. Must be $\geq$ NOMINAL
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value in the range between the minimum and maximum values. The default is determined based on the value of constant <code>STAT_CONTROL</code> in package <code>STATISTICS_CONTROL</code>

Attributes

Name	Value
<code>STAT_POSITION_NOMINAL</code>	1
<code>STAT_POSITION_MINVALUE</code>	2
<code>STAT_POSITION_MAXVALUE</code>	3

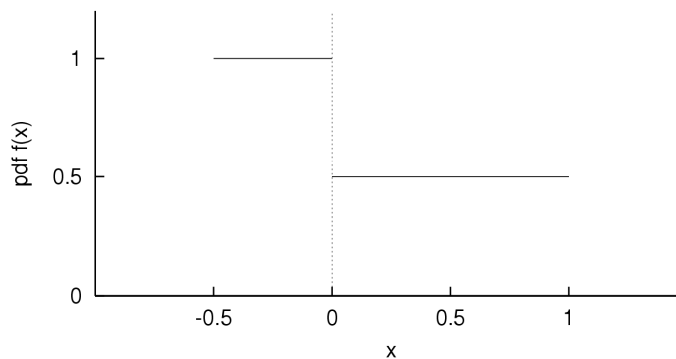
Return

`REAL`

See argument `MODE`

Example

`UNIFORM(NOMINAL=>0.0,MINVALUE=>-0.5,MAXVALUE=>1.0)`



Function **UNIFORM**

Declaration

```
impure function UNIFORM(  
    NOMINAL: TIME;  
    MINVALUE: TIME;  
    MAXVALUE: TIME;  
    MODE: STAT_MODE_TYPE := STAT_MODE  
) return TIME;
```

Uniform distribution with `TIME` nominal value and bounds

Arguments

Name	Type	Default	Description
NOMINAL	TIME		Nominal value
MINVALUE	TIME		Minimum value. Must be $\leq$ NOMINAL
MAXVALUE	TIME		Maximum value. Must be $\geq$ NOMINAL
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value in the range between the minimum and maximum values. The default is determined based on the value of constant <code>STAT_CONTROL</code> in package <code>STATISTICS_CONTROL</code>

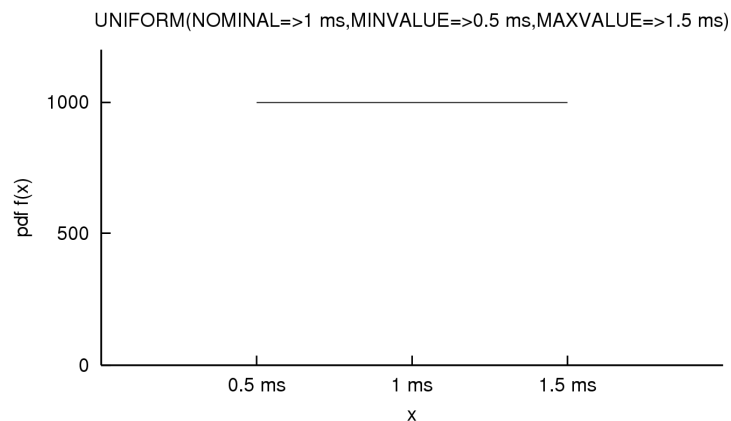
Attributes

Name	Value
<code>STAT_POSITION_NOMINAL</code>	1
<code>STAT_POSITION_MINVALUE</code>	2
<code>STAT_POSITION_MAXVALUE</code>	3

Return `TIME`

See argument `MODE`

Example



3.1.4.2 NORMAL

There are 4 forms of the NORMAL distribution with different arguments. These functions also have attributes defining position arguments.

Function **NORMAL**

Declaration

```
impure function NORMAL(
  NOMINAL: REAL;
  TOL: REAL;
  TRUNCATE: BOOLEAN := FALSE;
  ZSCORE: REAL := 3.0;
  MODE: STAT_MODE_TYPE := STAT_MODE
) return REAL;
```

Normal distribution with REAL nominal value and tolerance

Arguments

Name	Type	Default	Description
NOMINAL	REAL		Nominal value
TOL	REAL		Tolerance of the value. Must be ≥ 0
TRUNCATE	BOOLEAN	FALSE	If TRUE, truncate normal distribution at minimum and maximum values.
ZSCORE	REAL	3.0	Defines how many standard deviations are between minimum and nominal value and between nominal and maximum value
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value with minimum and maximum values at $\text{NOMINAL} \pm \text{ZSCORE} * \text{std.deviation}$. The default is determined based on the value of constant <code>STAT_CONTROL</code> in package <code>STATISTICS_CONTROL</code>

Attributes

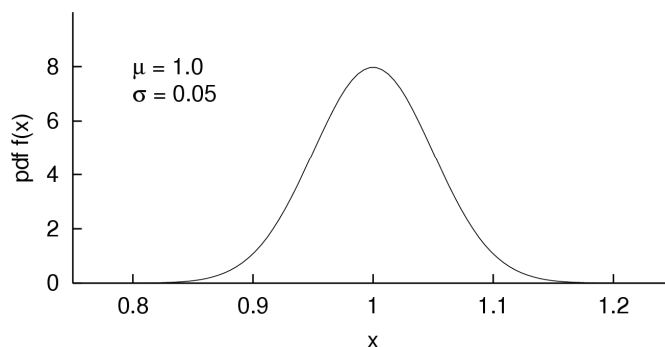
Name	Value
STAT_POSITION_NOMINAL	1
STAT_POSITION_TOLERANCE	2

Return REAL

See argument `MODE`

Example

`NORMAL(NOMINAL=>1.0,TOL=>0.1,ZSCORE=>2.0)`



Function **NORMAL**

Declaration

```

impure function NORMAL(
  NOMINAL:  TIME;
  TOL:      REAL;
  TRUNCATE: BOOLEAN := FALSE;
  ZSCORE:   REAL := 3.0;
  MODE:     STAT_MODE_TYPE := STAT_MODE
) return TIME;

```

Normal distribution with `TIME` nominal value and tolerance

Arguments

Name	Type	Default	Description
NOMINAL	TIME		Nominal value
TOL	REAL		Tolerance of the value. Must be ≥ 0
TRUNCATE	BOOLEAN	FALSE	If TRUE, truncate normal distribution at minimum and maximum values
ZSCORE	REAL	3.0	Defines how many standard deviations are between minimum and nominal value and between nominal and maximum value. Must be > 0 .
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value with minimum and maximum values at $\text{NOMINAL} \pm \text{ZSCORE} * \text{std.deviation}$. The default is determined based on the value of constant <code>STAT_CONTROL</code> in package <code>STATISTICS_CONTROL</code>

Attributes

Name	Value
<code>STAT_POSITION_NOMINAL</code>	1
<code>STAT_POSITION_TOLERANCE</code>	2

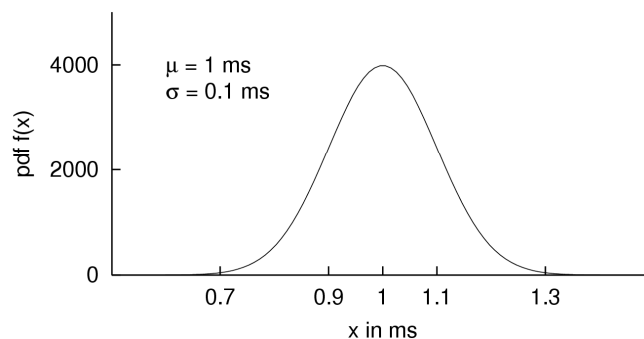
Return

`TIME`

See argument `MODE`

Example

`NORMAL(NOMINAL=>1 ms,TOL=>0.1,ZSCORE=>1.0)`



Function **NORMAL**

Declaration

```

impure function NORMAL(
  NOMINAL: REAL;
  MINVALUE: REAL;
  MAXVALUE: REAL;
  TRUNCATE: BOOLEAN := FALSE;
  ZSCORE: REAL := 3.0;
  MODE: STAT_MODE_TYPE := STAT_MODE
) return REAL;

```

Normal distribution with **REAL** nominal value and bounds

Arguments

Name	Type	Default	Description
NOMINAL	REAL		Nominal value
MINVALUE	REAL		Minimum value. Must be $\leq$ NOMINAL
MAXVALUE	REAL		Maximum value. Must be $\geq$ NOMINAL
TRUNCATE	BOOLEAN	FALSE	If TRUE, truncate normal distribution at minimum and maximum values
ZSCORE	REAL	3.0	Defines how many standard deviations are between minimum and nominal value and between nominal and maximum value. Must be > 0 .
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value with minimum and maximum values at $\text{NOMINAL} \pm \text{ZSCORE} * \text{std.deviation}$. The default is determined based on the value of constant STAT_CONTROL in package STATISTICS_CONTROL

Attributes

Name	Value
STAT_POSITION_NOMINAL	1
STAT_POSITION_MINVALUE	2
STAT_POSITION_MAXVALUE	3

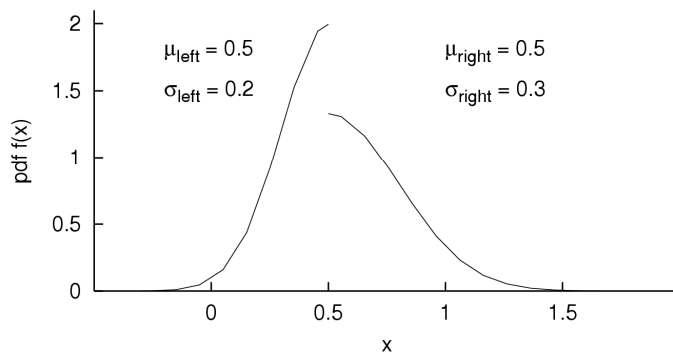
Return

REAL

See argument **MODE**

Example

NORMAL(NOMINAL=>0.5,MINVALUE=>-0.1,MAXVALUE=>1.4)



Function **NORMAL**

Declaration

```

impure function NORMAL(
    NOMINAL: TIME;
    MINVALUE: TIME;
    MAXVALUE: TIME;
    TRUNCATE: BOOLEAN := FALSE;
    ZSCORE: REAL := 3.0;
    MODE: STAT_MODE_TYPE := STAT_MODE
) return TIME;

```

Normal distribution with `TIME` nominal value and bounds

Arguments

Name	Type	Default	Description
NOMINAL	TIME		Nominal value
MINVALUE	TIME		Minimum value. Must be $\leq$ NOMINAL
MAXVALUE	TIME		Maximum value. Must be $\geq$ NOMINAL
TRUNCATE	BOOLEAN	FALSE	If TRUE, truncate normal distribution at minimum and maximum values.
ZSCORE	REAL	3.0	Defines how many standard deviations are between minimum and nominal value and between nominal and maximum value. Must be > 0 .
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value with minimum and maximum values at $\text{NOMINAL} \pm \text{ZSCORE} * \text{std.deviation}$. The default is determined based on the value of constant <code>STAT_CONTROL</code> in package <code>STATISTICS_CONTROL</code>

Attributes

Name	Value
<code>STAT_POSITION_NOMINAL</code>	1
<code>STAT_POSITION_MINVALUE</code>	2
<code>STAT_POSITION_MAXVALUE</code>	3

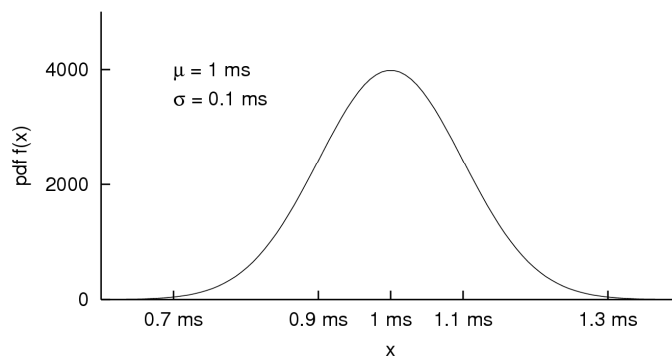
Return

`TIME`

See argument `MODE`

Example

`NORMAL(NOMINAL=>1 ms,MINVALUE=>0.7 ms,MAXVALUE=>1.3 ms)`



3.1.4.3 PWL_CDF

There are 4 forms of the PWL_CDF distribution with different arguments. These functions also have attributes defining position arguments.

Function **PWL_CDF**

Declaration

```
impure function PWL_CDF(
    NOMINAL: REAL;
    TOL:      REAL;
    CDF:      REAL_XY_TABLE;
    MODE:     STAT_MODE_TYPE := STAT_MODE
) return REAL;
```

Piecewise linear distribution with **REAL** nominal value and tolerance, specified as cumulative density function (CDF)

Arguments

Name	Type	Default	Description
NOMINAL	REAL		Nominal value
TOL	REAL		Tolerance of the value. Must be ≥ 0
CDF	REAL_XY_TABLE		Table specifying the CDF. The abscissa values must be defined on a normalized scale where -1, 0, 1 correspond, respectively, to the minimum, nominal, and maximum values. They must increase monotonically. The first abscissa value must be ≤ 0 , the last abscissa value must be ≥ 0 . The ordinate values must increase monotonically, the first ordinate value must be 0, the last ordinate value must be 1
MODE	STAT_MODE_TYPE	STAT_MODE	Specifies whether the function should return the nominal value or a random value in the range defined by mapping the CDF to the nominal value and tolerance. The default is determined based on the value of constant STAT_CONTROL in package STATISTICS_CONTROL

Attributes

Name	Value
STAT_POSITION_NOMINAL	1
STAT_POSITION_TOLERANCE	2

Return

REAL

The nominal value, or a random value in a range obtained by mapping the abscissa values -1, 0, 1 of the CDF to the minimum, nominal and maximum values

Example

PWL_CDF(NOMINAL=>3.0,TOL=>0.05,CDF=>CDF_TEST)

