

```

/*Verilog code sample */
module top;
/* (1) global format definitions */
`define TWOS_COMPLEMENT 1 /*...*/
/* (2) global rounding definitions*/
`define TO_INF 6 /*...*/
/* (3) global overflow definitions*/
`define SATURATION 1 /*...*/
/* (4) associate varprec attribute*/
(* varprec = data *)
reg [0:511] in1; /*...*/
(* varprec = descriptor *)
reg [0:1] d1; /*...*/
/* other Verilog declarations */
real fl1; integer i; reg b1;
initial begin
/* (5) set descriptor info */
$VpSetDescriptorInfo(d1, 256, 96,
`TWOS_COMPLEMENT,
`TO_NEAREST_INTEGER_IF_TIE_TO_ZERO,
`SATURATION, 1);
/* (6) set descriptor defaults */
$VpSetDefaultOptions (40, 96,
`FLOATING,
`TO_NEAREST_INTEGER_IF_TIE_TO_ZERO,
`NORMAL, 1);
/* associate descriptor to data */
$VpAssocDescrToData(in1, d1_FIX)
$VpAssocDescrToData(VpFL, d2_FLOAT)
#10; /* advance time by 10 */
/* (7) computations using Vp */
VpFl = 2**(-80);
in1 = $VpGetPi()/50;
in2 = $VpSin(in1) ** 2;
in3 = $VpCos(in1) ** 2;
in3 = in2 + in3 - 1;
$display("in3= %y\n", in3);
if(vpFl>in3)$display("Error \n");
else $display("OK \n");
/* (8)dynamic change of descriptor*/
$VpSetDescriptorInfo(d1, 256, 6,
`TWOS_COMPLEMENT,
`TO_NEAREST_INTEGER_IF_TIE_TO_ZERO,
`SATURATION, 1);
/* (9) continue simulation using
newer format */
in1 = $VpGetPi()/50; /*...*/

```

SUPPORTED VP FORMATS

FIXED POINT


any_size_integer.any_size_fraction

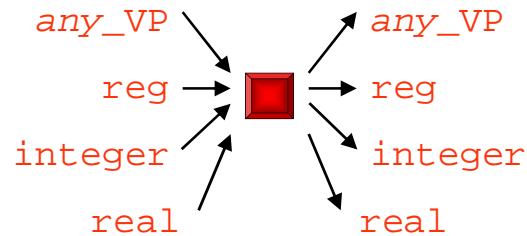
FLOATING POINT

sgn any_size_exp any_size_mantissa

sin	sh	asin	ash	+
cos	ch	acos	ach	-
tan	th	atan	ath	*
ctan	cth	actan	acth	/
ln	exp	sqrt	recast	**

SUPPORTED IN VERILOG

any_VP =  (any_VP)



For further information contact:

Fintronic USA, Inc.
Dr. Alec Stanculescu
alec@fintronic.com
Tel: 650-349-0108
Fax: 650-349-1150
www.fintronicusa.com



Fintronic USA

Opportunities to Improve Chip Design Involving Complex Arithmetic Computations

FOR

- DSP
- PROCESSOR
- EMBEDDED CORE
- MILITARY IC DESIGNERS

VP Floating and fixed point variable precision formats

Arithmetic, trigonometric, hyperbolic, logarithm, power, and exponential
Inside Verilog

Features:

- Pure Verilog® syntax
- Variable Precision (VP) fixed point and IEEE Std™ 754/854 floating point objects and functions and operators
 - +, -, *, **, and '/' for any combination of the following operands and result formats:

VP	arbitrary precision fixed point
	arbitrary precision floating point
Verilog	integers
	reals
	registers
	supported constants

- All trigonometric and hyperbolic direct and inverse functions called using any format and precision
- Power, logarithm, square root available for any format
- Accurate rounding and special condition reporting
- Automatic as well as user controlled sub-expression width size and format inference

Advantages:

- Choice of variable precision fixed point and floating point formats integrated into (System)Verilog
- Run time swapping formats for easy design space exploration and optimizations at the ESL level
- Straightforward implicit conversion between objects using different formats aiming at numeric value invariance
- Reduced overhead compared to other multi language, multi application, non EDA native approaches
- Intuitive Verilog syntax shortens learning time
- Ideal for DSP peak value estimation, and realistic word length optimization
- VP enables true ESL space exploration, smoothness in the design of next generation processors, and the R&D of custom arithmetic systems

Fintronic Super FinSim® 8.0

- First FinSim was sold on Linux in 1993
- Thousands of licenses in use
- Runs on all platforms: Solaris, Linux, Windows, 32 and 64 bit OS
- Supports Verilog 2001
- Supports OSCI's SystemC
- Integrated with Novas' Verdi and Debussy, Veritools' Undertow, Denali's memory modeling products
- NEW: Variable precision (VP) floating point and fixed point operations in Verilog.