



# VTRAM®

## VECTOR TRANSLATION PROGRAM SUPPORTS

### MULTIPLE SIMULATOR AND TESTER FORMATS

**VTRAN** is a program which reads the state/time information from simulation data files, performs some optional processing on this data and then re-formats it for any of over **30 popular logic simulators**. The program deals with the general class of problems involving simulation stimulus/results re-formatting encountered by designers who must deal with more than one logic simulation environment. A **TEST** option for **VTRAN** also provides interfaces to many popular device testers.

#### APPLICATIONS:

- Re-formatting existing simulation data files which were generated by one simulator into files compatible with another simulator.
- Translating state data files from logic analyzers, test programs or other data sources into stimulus files compatible with logic simulators. This includes the incorporation of pin timing.
- Modifying simulation data files including changing pin lists, pin order, bus radixes, time offsets, pin timing and time scaling.
- Convert simulation vectors into functional test vectors for physical device testers.
- Generate VHDL and Verilog testbench files.

**VTRAN** is able to handle a wide range of input data formats by essentially letting the user describe the format. This is done with a single, or a collection of format descriptors which the user customizes to describe the input data format. The input format parameters which can be specified by the user include:

- Pin names & order.
- Whitespace characters.
- Type & location of time tag, absolute or delta time.
- Location of state data & bus radixes used.

- Selection of TABULAR, SCRIPT or WAVE format.

In addition to the User-Programmable Reader, **VTRAN** also provides a growing Library of “canned” readers for popular simulator formats. Check the latest release for a current list of these “canned” readers.

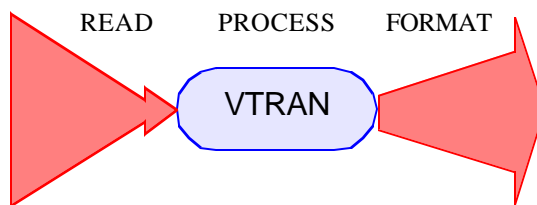
At invocation, a command file is specified which contains the **VTRAN** commands and descriptors. This file is composed of three major blocks; the **OVF\_BLOCK** which specifies the format of the input data file, the **PROC\_BLOCK** which specifies any processing to be done on the data, and the **TVF\_BLOCK** which specifies the format of the output file and the target simulator or tester.

The frustration of dealing with dissimilar simulator/tester data formats is eliminated with **VTRAN**. Files can easily be translated between most of the popular simulators in use today, as well as to many testers, with virtually any data manipulation desired. **VTRAN** can be used to automatically create test programs for physical device testing, directly from simulation results data files in a wide variety of formats with comprehensive support for scan data and multiple timesets.

#### EDA-IP:

Source III also provides pre-built and customized versions of **VTRAN** in the form of Reader IP and Writer IP blocks to interface with existing OEM products. Whether the OEM product is an ATPG tool requiring reading and writing support for WGL/STIL or a simulator requiring testbench output support, Source III’s EDA-IP blocks can satisfy the requirements. Source III’s EDA-IP blocks are very flexible and can be tailored to work with your OEM product.

Verilog VCD  
 Mentor  
 TSTL2  
 STIL  
 WGL  
 Synopsys  
 TDL\_91  
 ....



Verilog Testbench  
 VHDL Testbench  
 STIL  
 WGL  
 Tester Formats  
 Teradyne  
 Agilent/Verigy  
 Credence  
 Advantest  
 IMS  
 ....

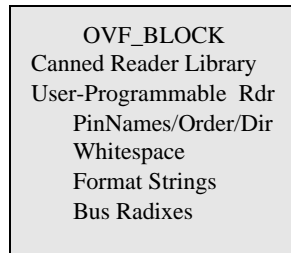
3941 Park Drive, Suite 20-342  
 El Dorado Hills, CA 95762  
 (916) 941-9403 (916) 941-9404 FAX  
<http://www.sourceIII.com>



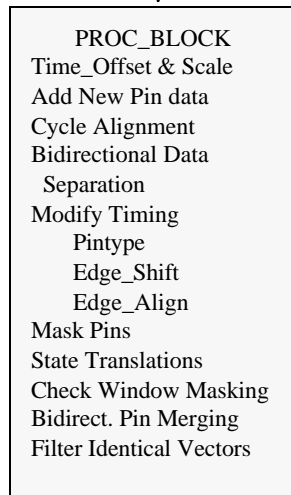
## VTRAN COMMAND FILE FLOW

### PROCESSING STEPS

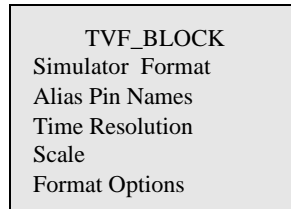
Read time/state information in the Original Vector File using either a canned reader, or the User-Programmable Reader for arbitrary vector formats.



Optional modifications to vector data read from OVF. Modification processes include state, timing, behavior and bidirectional data handling.



Format vector data for target Simulator or Tester environment. Specify format options. Output to Target Vector File.



**VTRAN** is a powerful simulation vector translator which can read the simulation data from one environment, perform a variety of sophisticated processing on this data, and produce a simulation data file compatible with almost any other simulation environment on the market today. **VTRAN** also handles the task of generating VHDL & Verilog testbench files. In addition, interfaces to many testers are available through a TEST option.

### DATA READERS

**VTRAN** reads data files from simulators, or other sources, using one of two methods. For some of those more popular simulators, **VTRAN** has a Library of canned readers which know how to read the time/state information of those files. Canned readers are currently available for the following file formats

- Verilog VCD and EVCD files
- Toshiba TSTL2 files
- Mentor LOG files, FORCE files and LSIM files
- TSSI TDS ascii files and WGL files

- Valid Tabular I/O files
- Fujitsu FTDL files
- Synopsys VSS WIF files
- IMS Test MEM format files
- TDL\_91 format files
- Test program files generated by VTRAN
- IEEE STIL (Standard Test Interface Language) files
- HP-3070 PCF files

Additional canned readers are released periodically.

Data formats for which a canned reader is not available can be read by **VTRAN** using its User-Programmable Reader Technology. Here, the user describes the format of data to be read using one or more "format strings" and **VTRAN** then uses these strings to extract the time/state information from the file. This method can be used with any tabular or script file format - whether created from a simulator, tester or logic analyzer.

### DATA PROCESSING

When translating time/logic data from one simulation format to another, or to a tester, it is often necessary to modify this data to clarify its intent and conform to the target environment's parameters. **VTRAN** can perform an array of mapping and translation operations on the simulation data it has read. These include:

- State Mapping
- Adding/removing pins
- Modifying time (scaling, offsetting, shifting edges)
- Collapsing print-on-change data to cycle-data
- Imbedding timing into cycle-data
- Separation of I/O data on bi-directional pins
- Merging of bi-directional data
- Masking data, defining strobe windows for outputs
- Pin name aliases

All of these operations are under complete user control so that the data in the output (translated) file can be customized for specific needs.

### OUTPUT FORMATTING

Once a set of simulation data has been read, and optionally processed, the output file format is selected from one of over 30 currently available interfaces, including:

EPIC	IKOS	LSI	LSIM
MAXPLUS	MENTOR	ORCAD	QSIM
SILOS	SPICE	STIL	TDL_91
TDS(TSSI)	TSTL2	VALID	VERILOG
testbench	VERILOG	VCD	VHDL testbench
VIEWSIM	WAVES	WGL	WIF

In addition to these standard formats, a UDF (user Defined Format) is also available for defining a custom output format. New formats are being added constantly and custom formats can also be made available. With the TEST option, **VTRAN** can output vectors for physical device testers, including:

CREDENCE SWAV  
TERADYNE Catalyst, J750, J971/973, FLEX  
VERIGY/AGILENT 83000/93000, Pinscale & Xmode  
ADVANTEST T66xx, T33xx  
IMS

## **VTRAN COMMAND FILE**

There are two command blocks required in every **VTRAN** command file, and a third optional command block. The first of these is the **OVF\_BLOCK**, which contains commands describing the format of data in the Original Vector File.

The second required command block is the **TVF\_BLOCK**, which contains commands that essentially tell **VTRAN** how to format the vectors for the Target Vector File.

The third (optional) command block file is the process block (**PROC\_BLOCK**) which must occur between the **OVF\_BLOCK** and **TVF\_BLOCK** in the file and is used to specify further processing to be done on the data prior to output formatting.

The **VTRAN** command file structure can be summarized as follows:

```
OVF_BLOCK
  BEGIN
  OVF_BLOCK_COMMANDS
  END
PROC_BLOCK {Optional}
  BEGIN
  PROC_BLOCK_COMMANDS
  END
TVF_BLOCK
  BEGIN
  TVF_BLOCK_COMMANDS
  END
END
```

Statement text in the **VTRAN** command file is not case sensitive. However, the case of pin names is preserved in the TVF file. Pin names can be composed of printable ASCII characters with the exception of the following: space, comma, semicolon, <, >, [, ], {, }, (, ), =, \, &, | and @.

### **OVF\_BLOCK COMMAND SUMMARY:**

**AUX\_FILE** [=]"filename";  
Used to specify auxiliary file for some canned readers.

**BEGIN\_LINE** [=] n ;  
Used to define the line number in the OVF file at which **VTRAN** should begin processing vectors.

**BEGIN\_STRING** [=] "string";  
Used to define a unique text string in the OVF file after which **VTRAN** should begin processing vectors.

**BIDIRECTS** [=] pin\_list;  
Defines the names and order of pins in the OVF file which are bidirectional.

**BUSFORMAT** radix; or **BUSFORMAT** pin\_list = radix;  
Specifies the radix of busses in the OVF file.

**CASE\_SENSITIVE**;  
Allows there to be more than one signal with the same name spelling but differing only in case of letters in the name.

**COMMENTS** [ON|OFF];  
Enables/Disables the passing of comments from the OVF to the TVF.

**GROUP** n [=] pin\_list;  
Together with the \$gstatesn keyword, it tells **VTRAN**

how the pin states are organized.

**INPUTS** [=] pin\_list:

Defines the names and order of input pins in OVF file.

**MAX\_UNMATCHED** [=] n [verbose]:

Specifies the number of, and information contained in, warnings for lines in the OVF which don't match a format\_string.

**MERGE\_FILE** . . . . . end\_Merge ;

**ORIG\_FILE** [=] "filename";

Used to specify the OVF file name to be translated.

**OUTPUTS** [=] pin\_list;

Defines the names and order of output pins in OVF file.

**SCRIPT\_FORMAT** [=] "format#1" [, . ."format#n"] ;

Format descriptors for User-Programmed reader.

**TABULAR\_FORMAT** [=] "format #1" [, . ."format#n"] ;

Format descriptors for User-Programmed reader.

**TERMINATE TIME** [=] n; or

**TERMINATE LINE** [=] m; or

**TERMINATE STRING** [=] "string";

Defines where in the OVF to stop processing, at a certain time, line number or when a string is reached.

**WAVE\_FORMAT** [=] "format #1" [, . ."format#n"] ;

Format descriptors for User-Programmed reader.

**WHITESPACE** [=] 'a','b', 'c', . . ,'n';

Defines characters in the OVF file that are to be treated as though they are space, (they are ignored).

### **PROC\_BLOCK COMMAND SUMMARY**

**ADD\_PIN** pinname = state1 [WHEN expr=state2  
OTHERWISE state3];

Tells **VTRAN** to add a new pin to the TVF, and allows you to define the state of this pin.

**ALIGN\_TO\_CYCLE** [-warnings] cycle pin\_list @ time,  
. . . , pin\_list @ time ;

Vectors can be mapped to a set of cycle data, the state of each pin in a given cycle is determined by its state at a specified strobe time in the OVF.

**ALIGN\_TO\_SIGNAL** [-novector] ref A->B  
SAMPLE=pinlist @ offset;

Print-on-change vectors are mapped to cycle-based vectors, with each cycle boundary determined by a reference signal transition, and the state of each pin in a given vector determined at the specified strobe time from the start of a cycle.

**ALIGN\_TO\_STEP** [-warnings] step [offset];

Forces a minimum time resolution in the TVF.

**AUTO\_ALIGN** [-warnings] cycle;

Collapses print-on-change data in the OVF to cycle data by computing strobe points from information given in the PINTYPE commands.

**BIDIRECT\_CONTROL** pin\_list = dir WHEN expr = state ;

Separates input data from output data on bidirects under control of a pin state or logical combination of pin states.

**BIDIRECT\_CONTROL** pin\_list = direction @ time ;

Separates input data from output data on bidirects based upon when the state transitions occur.

**BIDIRECT\_STATES** INPUT state\_list, OUTPUT state\_list ;

Separates input data from output data on bidirects where unique state characters identify pin direction.

**CHECK\_WINDOW** pin\_list @ t1, t2, [,t3, t4];

pin\_list@TRANSITION ctl, a->b, t1, t2;

**CYCLE** [=] n;  
Used to specify the time step between vectors in the OVF when the format of the vectors does not include a time stamp.

**CYCLE\_SHIFT** [fill\_character=c] pinlist @ n;  
**DISABLE\_VECTOR\_FILTER;**  
Can be used to disable filtering of redundant vectors.

**DONT\_CARE** 'X';  
Defines the character state to which output pins should be set outside of their check windows.

**EDGE\_ALIGN** pinlist @ rtime [,ftime] [xtime];  
Can modify pin transition times by snapping them to pre-defined positions within each cycle.

**EDGE\_SHIFT** pinlist @ rtime [,ftime] [xtime];  
Can modify pin transition times by shifting them by fixed amounts.

**MASK\_PINS** [mask\_character ='X'] [pin\_list] @ t1, t2 [-CYCLE] ; or  
@ CONDITION expr = state ; or  
@ TRANSITION A->B [-m] [,+n]; or  
@ CONTROL\_TRANSITION ctl\_pin A->B [,-m][,+n];  
Masks the state of specified pins to the mask\_character within the time range between t1 and t2, or when a specified logic conditions exists on other pins or for a number of cycles surrounding the pin's transition.

**PINTYPE** pintype pin\_list @ start1 end1 [start2, end2] ;  
Defines the behavior and timing to be applied to input and/or output pins during translation.

**POIC;**  
Specifies that vectors in the OVF file should be translated to the TVF only when at least 1 input pin has changed in the vector.

**SCALE** [=] nn;  
Used to linearly expand or reduce the time line of the OVF. Happens prior to any timing modifications.

**SEPARATE\_TIMING;**  
Tells **VTRAN** not to incorporate pin timing and behavior into the vectors themselves.

**STATE\_TRANS** [=] [dir] 'from1'->'to1', . . . ;  
Defines a mapping from pin states in the OVF file to states in the TVF file.

**STATE\_TRANS\_GROUP** pin\_list = 'from1'->'to1', . . . ;  
Supplements the **STATE\_TRANS** command by providing state translations on an individual pin or group basis.

**TEMPLATE\_CYCLIZATION** [params] ;

**TIME\_OFFSET** [=] n ;  
When reading the vectors from the OVF, the time stamp can be offset by an arbitrary amount.

## TVF\_BLOCK COMMAND SUMMARY

**ADD\_VTB\_TEXT** = "text", Before|After, <Location>;  
**ALIAS** ovf\_name = tvf\_name, . . . ; , or  
"ovf\_string"="tvf\_string";  
Provides a way to change the names of pins listed in the OVF, for listing in the TVF.

**BIDIRECTS** [=] pin\_list;  
Defines the names and order of pins to be listed in the TVF file which are bidirectional.

**BUSFORMAT** radix; or **BUSFORMAT** pin\_list = radix;  
Specifies the radix of busses in the TVF file.

**COMMAND\_FILE** = "filename" ;  
command file for the target simulator, in addition to the vector data file.

**DEFINE\_HEADER** [=] "text string";  
Allows you to inhibit the automatic generation of headers and replace it with a custom text string.

**FORCE\_SEQUENTIAL\_BUSSES** [+ | -] ;

**HEADER** [=] n;  
Causes a vertical list of the pin names to appear as comments in the TVF every n vector lines.

**INPUTS** [=] pin\_list ;  
Defines the names and order of pins to be listed in the TVF file which are inputs.

**INPUTS\_ONLY;**  
Causes only input and the input versions of bidirectional pins to be listed in the TVF.

**LOWERCASE;**  
Forces all pin names in the TVF to use lower-case letters.

**MERGE\_BIDIRECTS** state\_list;  
Merges the input and output state information of a bidirectional pin to a single pin after it has been split and processed.

**OUTPUTS** [=] pin\_list ;  
Defines the names and order of pins to be listed in the TVF file which are outputs.

**OUTPUTS\_ONLY;**  
Causes only output and the output versions of bidirectional pins to be listed in the TVF.

**RENAME\_BUS\_PINS** buslist = format;  
Provides a way of modifying specific bus names in the TVF.

**RESOLUTION** [=] n;  
Specifies the resolution of time stamps in the output vector file (n = 1.0, 0.1, 0.01 or 0.001).

**SCALE** [=] nn ;  
Linearly scales all times to TVF.

**SIMULATOR** [=] name [param\_list];  
Defines the target vector file format to be compatible with the simulator named.

**STOBE\_WIDTH** [=] n;  
Used with several of the simulator interfaces to define the width of an output strobe window.

**SYSTEM\_CALL** ". . .text . . .";  
Upon completion of translating vectors from the OVF to the TVF, **VTRAN** sends this text string to the system just prior to termination.

**TARGET\_FILE** [=] "filename";  
Specifies the name of the output Target Vector File.

**TESTER\_FORMAT** [=] tester [params];  
Specifies target tester format and parameters.

**TITLE** [=] "title";  
Specifies a special character string to be placed in the header of certain simulator vector files.

**UPPERCASE;**  
Forces all pin names in the TVF to uppercase letters.