

**Tempus Text Command Reference**  
**Product Version 21.12, December 2021**

## read\_activity\_file

```
read_activity_file name
[-block block_name ]
[-cell cell_name ]
[-end { time1 time2 ...timen } ]
[-format { VCD | TCF | SAIF | FSDB | PHY | SHM}]
[-hier_separator separator ]
[-name_mapping_rule file]
[-reset]
[-rtl {true|false}]
[-scale_duration scalefactor ]
[-scope scope_name ]
[-set_net_freq {true | false}]
[-start { time1 time2 ...timen } ]
[-start_time_shift value ]
[-weight value]
[-zero_delay {true|false}]
```

Specifies the name and type of activity file to be used as input. This command must be used when specifying activity files for static or dynamic power calculation.

### Parameters

|                          |                                                                                                                                                                                                            |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -block <i>block_name</i> |                                                                                                                                                                                                            |
|                          | Specifies the name of the FSDB/SAIF/TCF/VCD block to map a sub-block activity file with a top level Verilog file. Provides the ability to support multiple activity files based upon block instance names. |

The block argument is added to the FSDB/SAIF/TCF/VCD name before comparing it to the Verilog file.

`-cell cell_name`

Specifies the master cell name for the activity file. This parameter allows you to support master-cell-based activity specification. That is, you need to specify only the master cell name for the activity file, instead of specifying all the instances that were instantiated from a master cell.

#### Example

Let us consider a design that has a cell named "super\_filter" with four block instances "super\_filter1", "super\_filter2", "super\_filter3", and "super\_filter4":

##### Instance-Based Use Model:

```
read_activity_file -format VCD -block super_filter1
read_activity_file -format VCD -block super_filter2
read_activity_file -format VCD -block super_filter3
read_activity_file -format VCD -block super_filter4
```

##### Master-Cell-Based Use Model:

```
read_activity_file -format VCD -cell super_filter
```

`-end {time1 time2 ...timen }`

When you use the vector-driven method of performing dynamic instance-based power-consumption calculation, you can specify the part of the activity file from which you want power analysis to use for the power calculation. This option specifies the end time of this range and is used in conjunction with the `-start` option.

Specifies the end time as time-value pairs to report the average power across non-overlapping multiple windows specified in the activity file for a block, level of hierarchy, or other part of the design that you want to analyze. Units are in seconds (s), milliseconds (ms), microseconds (us), nanoseconds (ns), or picoseconds (ps).

*Default* : The time unit is defined in the first .lib file read during design import.

You can specify the `$worst_power_window_end` variable with the `-end` parameter to take the worst case end time from the last vector profiler run. Not required for TCF and SAIF.

**Note:** You can read multiple windows in a single VCD file by specifying multiple pairs of the start time and end time for the non-overlapping multiple windows specified in the activity file. The use model for reading multiple windows is:

```
read_activity_file -format VCD design.vcd -start {start1 start2 start3} -end {end1 end2 end3}
```

The `start1/2/3` and `end1/2/3` values should be in the ascending order.

Multiple current waveforms representing multiple VCD windows are merged into a single continuous current waveform. To capture the waveform for signals at the edge of user windows, a buffer time is added between the windows being stitched using the `set_power_analysis_mode -settling_buffer` parameter. This buffer time ensures that the stitched waveform is contiguous.

The generated waveform hence has the duration of user windows and the intervening settling buffers. The default settling buffer period is 400ps.

`-format { VCD | TCF | SAIF | FSDB | PHY | SHM}`

Specifies the net activity of the design using an activity file in the VCD, TCF, FSDB, SAIF, PHY, or SHM format. The arguments specified with the `-format` parameter are case-insensitive.

- The VCD file format uses the toggle count information as the basis for the power consumption calculation. These files can be in compressed (zipped) format using the `.gz` extension.
- The TCF file format contains data on the switching activity of the nets in the design. The switching activity includes the toggle count information and the probability of the net or pin being in the logic 1 state. These files can be in compressed (zipped) format using the `.gz` extension.

Supports extended TCF which contains transition density for X and Z as shown in the following extended TCF example:

```
"out" : "0.0 0" "0.005000 2" "0.0 0"
```

The fields are defined as follows:

Field 1: net or pin name

Field 2: duty and transition density for transition of 0-1

Field 3: duty and transition density for transition of X(0-X-1)

Field 4: duty and transition density for transition of Z(0-Z-1)

- The SAIF file format contains aggregate switching activity information on the nets and/or ports in the design. It does not contain time-based information. This file format is used for quick average power analysis of the design.
- The FSDB file format specifies the net activity of the design using an activity file in FSDB format. Voltus supports the FSDB version up to 5.8.

**Use Model**

```
read_activity_file -format FSDB fsdbfile
```

```
report_power
```

(or)

```
report_vector_profile
```

**Example**

```
read_activity_file -format FSDB test.fsdb -scope top
```

- The PHY (Palladium database format generated in the Palladium environment) format contains the simulation activity data generated during emulator-based simulation. The PHY format can be used in all vector-based power analysis flows (vector profiler, static power analysis and dynamic vector-based power analysis).

**Use Model**

```
read_activity_file -format PHY <path to PHY database directory>
```

**Example**

```
read_activity_file -format PHY ./phy2/trace.phy -scope test
```

**Notes:**

- The xeDebug software must be present in the PATH environment. Add xeDebug in the path using the following command:  

```
set path = (<install_dir>/bin $path)
```

 The xeDebug software version should be the same as the one using which the PHY database is generated.
- The PHY database generated from the following UXE/VXE version is supported:  
 UXE18.6.0 onwards  
 VXE18.6.0 onwards
- When reading a PHY format activity file, you require a DPA (DPA\_ToggleCountFormat) license.
- For information on error messages displayed while reading the PHY database, refer to the `xe.msg` file in the PHY database directory.
- Make sure that the PHY database is not locked before invoking the Voltus software.
- Make sure that the xeDebug software is not already running on the machine.
- Ensure that the following directories are present in the top-level PHY database directory: `*.phy`, `PDB`, `cellList`, `QTDB`, `dbFiles`, and `.design`.
- It is recommended that the write permission is assigned on the entire PHY database. In case the write permission cannot be granted, you must set the `SWFV_FREEZE_PHY` environment variable to 1.
- If you want to run parallel Voltus executables on the same top-level PHY directory with multiple `*.phy` databases, it is suggested to create local copies of the multiple top-level directories, with links to the original PHY database. This will allow local copies of the PHY database to acquire separate locks.
- The Simulation History Manager (SHM) database format is a record of the data signal changes that occur during design simulation. The SHM database consists of a directory, typically with the `.shm` suffix. The directory contains two disk files, namely `*.trn` and `*.dsn`. For more information about the SHM format, refer to the *SimVision Analysis Environment* manual available on the Cadence Online Support (COS) website.

**Example**

```
read_activity_file -format SHM ../shm/ADDER.trn -scope adder_e/u1 -start 0ns -end 60ns
```

`-hier_separator` *separator*

Specifies the separator character in the hierarchical net names, bus names, and pin names for the block or other part of the design that you want to analyze. It must be the same as the separator character in the design netlist.

*Default* : slash (/)

`-name_mapping_rule` *file*

Specifies to enable rule-based mapping to map RTL vectors to gate-level netlist. This parameter helps in resolving name mismatch between RTL and gate level netlists.

The name mapping rule file format is:

```
Name_Mapping_Rule_file
# RTL      # Gate
<rtlstring> <gatestring>
```

where, each line is a rule. The string in the first column(RTL) is what we want to replace with the string in the second column(Gate). The rules in every line are added onto the previous rules.

For example, to replace all the square [ ] brackets with under score \_ for RTL-to-Gate matching, the rule file will be:

```
#RTL      #GATE
[          _
]          _
```

**-reset** If **-reset** is specified, all previous activity files specified using the `read_activity_file` command will be reset and ignored.  
*Default* : All the previous activity files specified using the `read_activity_file` command are loaded and used (They are additive).

**-rtl {true|false}**

Specifies to process the vector as an RTL vector and perform state propagation. The propagation behavior is global and the full design is propagated. RTL vectors are zero-delay vectors so delay annotation is always done for signals from this vector file.

*Default*: false

**-scale\_duration scalefactor**

Scales the FSDB/SAIF/TCF/VCD duration value by the specified scale factor. For example, an FSDB scale of 2 will double the FSDB duration value.

**Note**: The **-start** and **-end** parameters of the `read_activity_file` command are post scaled.

**-scope scope\_name**

Specifies the FSDB/SAIF/TCF/VCD scope, that is, the name of the module within the activity file associated with the block or other part of the design that you want to analyze.

The scope argument is removed from the FSDB/SAIF/TCF/VCD name before comparing it to the Verilog file.

If the scope value has backslash '\ ' or square brackets '[]', you must specify the scope within curly brackets {{.....}} to ensure that '\ ' or '[]' are correctly processed by the Tcl Interpreter. In the following example, the scope value is specified within curly

brackets as it contains '\' and '[':

```
read_activity_file -scope {{reg_based_buffer.reg_inst\sv_xpo1_trans_q_reg[0][14][7]
[sv_re][4]}}
```

If '\' and '[' are not there, curly brackets are not required.

`-set_net_freq {true | false}`

This option should be set when an activity from a TCF or VCD file needs to be used to capture net frequency for signal EM analysis (verify\_AC\_limit). In addition, you must use the verify\_AC\_limit -use\_db\_freq option to read the toggle rates from the activity file.

*Default* : false

**Example**

The following example reads a TCF file which is used to capture net frequency for signal EM analysis:

```
read_activity_file -format TCF -set_net_freq true design.tcf
verify_AC_limit -use_db_freq -report signal_em.rpt
```

`-start { time1 time2 ... timen }`

When you use the vector-driven method of performing dynamic instance-based power-consumption calculation, you can specify the part of the activity file from which you want power analysis to use for the power calculation. This option specifies the start time of this range and is used in conjunction with the -end option.

Specifies the start time as time-value pairs to report the average power across non-overlapping multiple windows specified in the VCD file for a block, level of hierarchy, or other part of the design that you want to analyze. Units are in seconds (s), milliseconds (ms), microseconds (us), nanoseconds (ns), or picoseconds (ps).

*Default* : The time unit is defined in the first .lib file read during design import.

You can specify the \$worst\_power\_window\_start variable with the -start parameter to take the worst case start time from the last vector profiler run. Not required for TCF and SAIF.

**Note:** You can read multiple windows in a single VCD file by specifying multiple pairs of the start time and end time for the non-overlapping multiple windows specified in the activity file. The use model for reading multiple windows is:

```
read_activity_file -format VCD design.vcd -start {start1 start2 start3} -end {end1 end2 end3}
```

The start1/2/3 and end1/2/3 values should be in the ascending order.

Multiple current waveforms representing multiple VCD windows are merged into a single continuous current waveform. To capture the waveform for signals at the edge of user windows, a buffer time is added between the windows being stitched using the set\_power\_analysis\_mode -settling\_buffer parameter. This buffer time ensures that the stitched waveform is contiguous.

|                                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                                       | The generated waveform hence has the duration of user windows and the intervening settling buffers. The default settling buffer period is 400ps.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| <code>-start_time_shift value</code>  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|                                       | Specifies to shift the start time of the specified activity files.<br>When you specify the <code>-scale_duration</code> parameter with the <code>-start_time_shift</code> parameter, it scales the start time shift along with the start/end time.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <i>name</i>                           | The name of the activity file.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>-weight value</code>            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|                                       | Specifies the weight number of the TCF file. 1.0 is the default value. The value should be a floating number that is smaller than 1.0. This parameter allows you to merge multiple TCF files at different function modes for static power estimation of a design.<br>In the following example, TCF <i>file1</i> , <i>file2</i> , and <i>file3</i> has the weight numbers 0.3, 0.2, and 0.4, respectively.<br><code>read_activity_file -format TCF file1 -weight 0.3</code><br><code>read_activity_file -format TCF file2 -weight 0.2</code><br><code>read_activity_file -format TCF file3 -weight 0.4</code><br>Here, <i>file1</i> , <i>file2</i> , and <i>file3</i> will be merged together to represent the toggle and duty for the nets/pins in the design. The relative weight of 0.3/0.9, 0.2/0.9, and 0.4/0.9 would be used for static power calculation. In this example, 0.9 is the sum of weights. |
| <code>-zero_delay {true false}</code> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|                                       | Specifies to process the vector as a zero-delay vector and perform delay annotation.<br><br><i>Default</i> : false                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

## Activity Precedence

The following commands can define activity:

```
read_activity_file file
  [-format { VCD | TCF | SAIF | FSDB }]
  [-hier_separator separator ]
  [-start time ]
  [-end time ]
  [-reset]
```

[-scope *scope\_name* ]  
[-block *block\_name* ]

set\_default\_switching\_activity

[-input\_activity *factor* ]  
[-seq\_activity *factor* ]  
[-period *value* ]  
[-duty *value* ]  
[-global\_activity *factor* ]  
[-hier *hierarchy\_name*]  
[-clock\_gates\_output *activity\_factor* ]

set\_switching\_activity

[-reset]  
[-clock *clock\_name*]  
[-period *value* ]  
[-unclocked]  
[-activity *factor* | -density *transition\_density* ]  
[-net *net\_name* | -port *port\_name* | -pin *pin\_name* ]  
[-duty *value* ]  
[-inst]

The precedence of activity is as follows:

1. User-Defined: Activities applied to specific nets/pins/ports
2. Activity File: Activity specification through a VCD/FSDB/SAIF/TCF format
3. Clock Gates Output Activity: Activity specification to the output of all clock gates
4. SDC/TWF constants: Set constants to a list of pins or ports for use by the timing engine (timing analysis)
5. Hierarchical Global Activity: Activity specification for a specific hierarchy
6. Global Activity: Activity specification for all instances that are part of the data network.
7. Sequential Element Activity: Activity specification applied to the output of all sequential elements.
8. Primary Input Activity: Activity specification for all primary inputs

If you specify multiple activity data for the same items, the power engine will use the one with the higher precedence.



## Examples

- The following command reads the compressed ap\_wait\_test\_pll1\_mod .vcd.gz VCD file that contains the toggle count information for the power consumption calculations, specifies the crm\_ap module within the VCD file as the module associated with the part of the design being analyzed, specifies the slash character (/) as the hierarchical separator for hierarchical net names, bus names, and pin names for the part of the design being analyzed, and specifies the start (180002ns) and end (189802ns) times from the VCD file for the part of the design being analyzed:

```
read_activity_file -format VCD \  
-scope testbench/top/ap/mcu_platform/crm_ap \  
-hier_separator \  
-start 180002ns \  
-end 189802ns \  
ap_wait_test_pll1_mod.vcd.gz
```

- The following command reads the compressed A.vcd.gz VCD file that contains the toggle count information for the power consumption calculations, specifies A1 as the block to map a sub-block VCD file with a block instance at the top-level, and specifies the start (180002ns) and end (189802ns) times from the VCD file for the part of the design being analyzed:

```
read_activity_file -format VCD \  
-block ap/A1 \  
-start 180002ns \  
-end 189802ns \  
A.vcd.gz
```

- The next example specifies a VCD activity file called dmac\_mac.vcd for the scope top/dma\_dut.

```
read_activity_file -format VCD -start 10ns -end 20ns \  
-scope top/dma_dut dmac_mac.vcd
```

- The following command specifies multiple pairs of the start time and end time for the multiple windows in the VCD file :

```
read_activity_file \  
-format VCD \  
-scope adder/u1 \  
-start {0ps 2000ps} \  
-end {1100ps 3100ps} \  
-block ../vcd/dut_5buf_full_chip.vcd
```

- The following command specifies to read the RTL-level VCD activity file:

```
read_activity_file -format VCD -scope scope_top -start 300ns -end 700ns /DesignData/VCD/rtl.vcd -block
```

```
inst_C1 -rtl true
set_power_analysis_mode -method vector_profile -vector_profile_mode transient -worst_window_type full
-power_grid_library /DesignData/PGV/library_pv.cl
```

## Related Topics

- "[Dynamic Power and IRDrop Analysis](#)" in the *Voltus User Guide*

## Related Commands

- [set\\_power\\_analysis\\_mode](#)
- [report\\_power](#)

*propagate\_activity*

*report\_instance\_power*