15-Jun-1993

This document describes the modification state of the DECsim System and the features added in the Version 6.0 release.

VAX/VMS and Alpha AXP/OpenVMS

DECsim Release Notes

Version 6.0

Supersession/Update Information: DECSIM.DOC 10-Jun-1992

| | |
|---|---|
| Operating System Version: | VAX/OpenVMS V5.4 thru V5.5 |
| | AXP/OpenVMS V1.5 |
| Software Version: | DECsim V6.0-4406 |
| | NETPRO V6.0-2939 |
| | SX V6.0-3069 |
| | OBJMRG V6.0-141 |

```
**********************************************************************
*                                                                    *
*                  D E C S I M   V e r s i o n   6.0                 *
*                                                                    *
* 1. This DECsim Version requires VAX/VMS Version 5.4 or later       *
*    or Alpha AXP/OpenVMS Version 1.5 or later.                      *
*                                                                    *
* 2. This version of DECsim does not require recompilation of any    *
*    NET files.  However, you may wish to recompile your models      *
*    to take advantage or certain new features.                      *
*                                                                    *
**********************************************************************
```

# 1 MAJOR HIGHLIGHTS OF THE V6.0 RELEASE

This is a summary of the major new features and/or changes to the DECsim system (DECsim, NETPRO, SX). The subsequent sections describe these and other new features in detail.

1. DECsim has now been ported to the Alpha AXP system.

2. A single DECSIM$SYS directory supports both VAX and Alpha AXP processors in a dual architecture cluster.

3. PHANTLIB (vector phantom library) is now split off from GATES.NLB.

4. DECsim images are now distributed. Previously, object libraries were distributed and linked locally. Model libraries are now distributed in both source and compiled forms.

5. The keyword "DECSIM" is now provided as a DCL command. Users should remove any definition of "DECSIM" as a foreign command symbol from their LOGIN.COM files. System managers should remove any definition of "DECSIM" as a foreign command symbol from SYS$MANAGER:SYLOGIN.COM.

## 2 MAJOR BUG FIXES INCLUDED IN THE V6.0 RELEASE

A NOTES conference called DECSIM_BUGS on node DECSIM has been opened for reporting and tracking of SPR's. All future SPR's should be posted as notes in this conference.

This is a summary of the major bug fixes to the DECsim system (DECsim, NETPRO, SX) included since the V5.9 release.

1. Fixed a bug that crashed DECsim when "?#2" appeared in a DEPOSIT waveform immediately after "[".

2. Fixed a bug that crashed DECsim when DEPOSIT/FREE "signal" was given the name of a behavior state.

3. Fixed a bug in NETPRO that would incorrectly indicate a .NLB (library) file was truncated when, in fact, it had only been copied from one disk to another with a different allocation cluster size.

# 3 DECSIM COMMAND LANGUAGE AND RUN-TIME SYSTEM

## 3.1 Introduction

The DECsim command language allows you to observe and control the simulation of a logic network. You can enter commands directly from the terminal and from an already prepared command file. You can establish simulation conditions and select monitoring options as well as record results, times and commands. The language has many features similar to the behavioral modeling language.

## 3.2 Differences Between DECsim Release V5.9-4318 and V6.0-4406

This section lists the new features and/or changes for the DECsim command

language and run-time system for V6.0.

### 3.2.1 General Feature Differences

1. DECsim and its support programs have been ported to the Alpha AXP architecture. Most major features are supported, including mixed mode, behavioral modeling with non-BDS routine calls, fault simulation, SAVE/RESTORE. Just a few features are unavailable on the Alpha AXP system:

    1. No Zycad interface

    2. No Realchip/Realmodel interface

    3. No Sprite interface

    4. No Datavue interface

5. No source line debugging or single-stepping for behavior models

6. The SHARE command (AXE/RAX interface) is not yet ported.

Models (.NLB files) and networks (.NOB files) that do NOT contain behavior models are portable between the two architectures. Behavior models, which include compiled object modules, are not portable; networks containing behavior models, which include an image file, are not portable. Behavior models and networks containing behavior models must be compiled on the target architecture. Cross-compilers and cross-linkers are available on VAX processors targeted for Alpha AXP processors but not the reverse. The command procedures included with this release are not set up to support cross-compilation but the DECsim software in this release can support cross-compilation with user-written command procedures.

2. There are some file name changes in this release and, consequently, some logical name changes too. All files in the DECSIM$SYS directory that are specific to the VAX architecture have a suffix "_VV" (VMS VAX) appended to the name. Alpha AXP architecture-specific files have "_VA" (VMS AXP) appended. For example, the DECsim image for the Alpha AXP architecture is named DECSIM_VA.EXE and the vector phantom library with VAX architecture behavior models is PHANTLIB_VV.NLB. Because of this, logical names set up by DECsim V6.0 are not compatible with previous versions. The DECsim installation procedure sets things up so that a system-wide logical name "DECSIM$SYS" is created pointing to the directory where all the DECsim files are kept.

3. The verb DECSIM is now inserted in the DCL command tables. It is no longer necessary to define "DECSIM" as a foreign command. In fact, the definition of foreign command "DECSIM" used in previous DECsim releases is incompatible with V6.0 and should be removed from user's LOGIN.COM files and from the system login file, SYS$MANAGER:SYLOGIN.COM.

4. When the user runs DECsim, COMBEH, or COMNET, additional logical names are set up. To switch back to DECsim V5.9 or earlier, most of these logical names must be redefined. A command procedure, DECSIM_INIT.COM, has been provided to assist the user to switch the appropriate logical names. To switch to DECsim V5.9 or earlier after running DECsim V6.0, issue these commands from the DCL prompt:

```
$ DEFINE DECSIM$SYS <any-DECsim-directory-name>
$ @DECSIM$SYS:DECSIM_INIT
```

To switch to DECsim V6.0 after running V5.9 or earlier, issu these commands:

```
$ DEASSIGN DECSIM$SYS
$ @DECSIM$SYS:DECSIM_INIT
```

5. The system variable %ARCHITECTURE has been added so that DECsim command files (.IND files, DECSIM.INI, and DECSIM.SYS) can test for the type of computer that is running. %ARCHITECTURE evaluates to 1 on VAX and 2 on AXP.

6. A new formal parameter $LOGICAL has been added to command macros. You are expected to provide a VMS logical name as the value for this parameter type. The substitution string is the translation of the logical name.

7. Command macro VAXSTATISTICS has been changed to VMSSTATISTICS.

8. The installation procedure no longer creates a world-writable directory DECSIM$STT. This directory is now created with W:E protection and contains one file having W:RW protection. Your system manager can disable DECsim statistics gathering by deleting the DECSIM$STT directory and the one file it contains.

### 3.2.2 DECsim/ZYCAD Features

DECsim/ZYCAD is not available on Alpha AXP systems but continues to be an option on the VAX version. This feature may become available for AXP systems in the future if there is any demand for it. The DECsim startup file no longer defines logical names and/or queues for Zycad. We believe there are no Zycad simulation accelerators in Digital at this time. If this assumption is incorrect please send mail to DECSIM::DECSIM_SUPPORT right away.

### 3.2.3 DECsim-RealChip Features

The term DECsim-RealChip includes DECsim-RealModel. The interface to the two types of Valid Logic Systems hardware is the same. If you have any questions regarding DECsim-RealChip or DECsim-RealModel please contact ECADSR::HOTLINE

A current limitation of the Realchip/Realmodel server is that it may not be attached to VMS V5.0. This feature is not available on the Alpha AXP system.

## 3.3 Summary of Features Implemented in this DECsim Release

| Commands Implemented in this Release | Yes/No | With Wildcarding? (Default NO) | options not implemented |
|---|---|---|---|
| ACTIVATE | YES | YES (only with options) | |
| ALLOCATE | NO | | |
| BEGIN-END | YES | | |
| BREAK | YES | | |
| CALL | YES | | |
| COMMENT | YES | | |
| COMPILE | YES | | /FORCE, /EXTRACT |
| CONFIGURE | YES | | |
| CONTEXT | YES | | Works only in conjunction with EXAMINE, and DEPOSIT commands. |
| DCL | YES | | |
| DEBUG | YES | | /WINDOW |
| DEPOSIT | YES | YES | /PARAMETER |
| DETECT | YES | YES | /EXCEPT, /NOCONTINUOUS_VISIBILITY /MAX_TRANSITIONS |
| DROP | YES | | |
| EDIT | YES | | |
| ERROR | YES | | |
| EVALUATE | YES | | |
| EXAMINE | YES | YES | /SYMBOLIC /INVISIBLE /ALL_FAULTS |

| | | | |
|---|---|---|---|
| EXIT, QUIT | YES | | |
| FAULT | YES | YES | |
| FOR | YES | | |
| FORMAT | NO | | |
| GRIPE | NO | | |
| HALT | YES | | |
| HAZARD | YES | | |
| HELP | YES | | |
| IF-THEN-ELSE | YES | | |
| INDIRECT | YES | | /ABBREV, /MASTER, /SLAVE |
| KEY | YES | | |
| LEAVE | YES | | |
| LOAD | YES | | /ID, /OBJ, /SYMBOL, /SAGE2, /PLATO |
| LOG | YES | | /ALL, /UNIQUE and /ARROW /OVERWRITE, /TTY |
| MACRO | YES | | /access list, /SET, /ENABLE, /DISABLE, /CANCEL, /SHOW, /SWITCH, /OPTION |
| MODIFY | NO | | |
| MOS | YES | | |
| PATTERN | YES | | /SCALE, /LINEAR |

| | | |
|---|---|---|
| PRINT | YES | /SYMBOLIC, /PRECISION and a few obscure features |
| RADIX | YES | (no options) |
| RECALL | YES | |
| RESTORE | YES | |
| SAVE | YES | |
| SCOPE | YES | |
| SELECT | YES | |
| SHARE | VAX only | |
| SIMULATE | YES | /SINGLESTEP, /ZERODELAY |
| SPAWN | YES | |
| STATE | YES | |
| STATISTICS | YES | |
| SYNONYM | NO | |
| TERMINAL | YES | VT125 |
| TIME | YES | |
| TIMING | YES | |
| TRACE | YES | /AT, /CHECK |
| UNLOAD | NO | |
| USE | YES | |
| WAIT | NO | |
| WATCH | YES | /CHECK |
| WHILE, UNTIL | YES | |
| Ctrl/C | YES | A (abort from simulation), S (save) |
| ZERO_DELAY | YES | |

| | | | |
|---|---|---|---|
| Command Macros provided with this Release | Yes/No | With Wildcarding? (Default NO) | Options not implemented |

| | | |
|---|---|---|
| COMBEH | YES | |
| CONTINUE | YES | |
| EDITCOMMANDMACRO | YES | |
| EDITTEXTMACRO | YES | |
| INFORM | YES | |
| LABELS | YES | |
| PAUSE | YES | |
| RESUME | YES | |
| SPR | YES | |
| STEP | YES | |
| SYMBOLS | YES | |
| TYPE | YES | No options |
| VMS_COMMAND | YES | |
| VMSSTATISTICS | YES | |

## 3.4 Command Language Bugs and SPR's Fixed in This Release

1. Fixed a bug in the fault coverage number that DECsim prints in response to the SHOW FAULT command. The bug was due to an integer arithmetic overflow when then number of faults inserted was more than 21,474 and affected only the fractional part of the result.

2. Fixed a bug in command recall that was visible only from a window on a DECSTATION. The uparrow key was being ignored.

3. Fixed a bug in DEPOSIT. If "?#2" was the first element following a "[" in a waveform, DECsim would eventually crash. For example:

```
SIM> DEPOSIT/FORCE X = [?_#2:10, 1:10]10
```

would crash DECsim.

4. Fixed a bug in DEPOSIT/FREE. If DEPOSIT/FREE were given the name of a behavior state to "free", DECsim would crash with an access violation. Please note the "" around "free"; DEPOSIT/FORCE to a behavior state remains unimplemented.

5. Fixed a bug in reading .NOB files. These files are checked to see that the actual file size corresponds with the size that NETPRO wrote. A truncated file may be caused by an aborted copy over DECnet. However, DECsim used to compare the allocated file size rather than the written size, causing a mis-match when a file is copied from one disk to another with a different allocation cluster size.

6. Fixed a bug in cancelling a WATCH/IMMEDIATE from within the WATCH action. If one of the signals being watched was a behavior array element, DECsim would miscalculate the address(es) on which the WATCH was being removed and crash in a bug trap. This bug did not

affect normal delayed WATCHes.

7. Fixed a bug in the 6.0 field-test version where the ".n" format for command macro formals was not recognized.

8. Due to an accidental change to the NOB file format, the field-test version of DECsim was unable to load NOB files written by NETPRO V5.9 or earlier if those files contained MOS models. This problem has been fixed for this release.

## 3.5 Known Bugs, Restrictions, and Deficiencies in DECsim

### 3.5.1 Restrictions on the Alpha AXP Version of DECsim

1. DECsim is not set up to work with translated shareable images. In particular, EDT on the Alpha AXP system is presently a translated image. Therefore, the EDIT command with the default editor, EDT, does not work. However, other editors, such as LSE or TPU (EVE),

work fine.

2. Using the SHARE command crashes DECsim.

3. The DATAVUE, ZYCAD, Sprite, and RealChip interfaces do not work.

4. Behavior model source-line stepping does not work.

5. Due to restrictions in OpenVMS for Alpha AXP systems, once a behavior image has been activated, you cannot recompile and activate the new image. To work around this problem you must either exit and rerun DECsim or supply a unique file name for the .EXE file using the /EXECUTABLE option of the compile command.

### 3.5.2 General DECsim Restrictions

1. The special behavior wait module that is used by some non-BDS routines must be upgraded for V6.0. If you need this module send mail to DECSIM::DECSIM_SUPPORT.

2. DECsim crashes if you supply a fourstate actual value to a routine that is defined with a wide twostate parameter. (mixed 4/2 state expressions)

   Workaround:
   assign the fourstate value to a twostate temporary, then pass the

value of the temporary to the routine.

3. Remaining Problems with VECTOR <--> VECTOR Interconnect

In order for you to make a good decision about whether you should model using direct VECTOR <--> VECTOR interconnect, the following list documents the remaining major known problems. We are working on solutions -- however these problems are not fixed in the current version.

- DECsim can not handle the naming of a VECTOR OUTPUT PORT using different signal names for different bits of the VECTOR (described using the concatenate operator (ampersand) in your network description). It also can not handle bringing part of a VECTOR OUTPUT out through a MODEL header and connecting the remaining part of the VECTOR OUTPUT local to that MODEL.

- If your signal is a multi-bit bus, but it is modeled at the SCALAR or MOS level, you can not reference the multi-bit bus as one larger entity, you must reference it on a bit by bit basis.

- You can not DEPOSIT or DEPOSIT/FORCE to the VECTOR PORTS (can only use pattern, which does not support forcing of OUTPUTS).

4. Restrictions for Command Language Hazards
   - Hazard checkers are not allowed on vector outputs (behavior or memory).
   - Hazard checkers are ignored in ZYCAD networks.
   - Hazard checkers are ignored in fault simulations.

5. The DEPOSIT command has not been enhanced to accommodate vector interconnect (behavior and memory primitive multi-bit ports). EVALUATE or PATTERN can be used instead.

6. Recursive macros are not detected and cause infinite recursion.

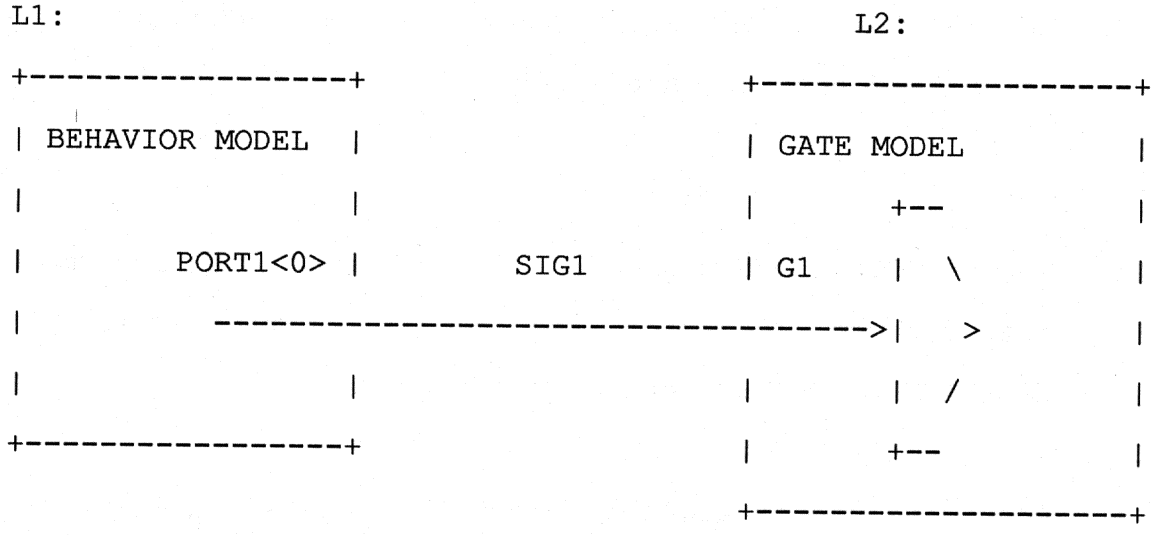
7. The PATTERN and TRACE commands for soft and ZYCAD simulations accept up to 65,532 signals. However, depending on the amount of physical memory on a system, using large numbers of signals in a single command causes the operating system to page fault to a point that may require many hours of CPU time to process the command. It is recommended that large numbers of signals be broken down to a number of commands, each containing approximately 300 to 500 signals. The most efficient number of signals per command is dependent on the physical memory available as well as the maximum working set (WSextent) allowed by the user authorization file.

8. For a number of commands (like TRACE, etc), if you give a command

and the signal name given in the command is an input which is driven by a SPLIT or MERGE, there may be problems, including crashes or infinite loops. To get around this, you must use either the signal name of the output driving the signal or a higher level signal name.

AN EXAMPLE:

In this network, a SPLIT is placed between the L1 BEHAVIOR PORT output and the L2 Gate Model input.




Commands:

TRACE L1.PORT1 - this is OK, the BEHAVIOR PORT output

is TRACED.

TRACE SIG1 - this is also OK. Again, the BEHAVIOR PORT output is TRACED.

TRACE L2.G1 - this does not work, because the input name "G1" was given.

9. When trying to EXAMINE/FULL memory connections, with a certain model hierarchy, user sees label.{?Nameless_Signal?}. Work around: Don't use EXAMINE/FULL in these cases, and you'll just see the symbol at the current scope.

10. Fault Simulation

- SET DETECT/THRESHOLD=x is operational with only one detector when doing fault simulation. If you specify multiple detectors with /THRESHOLD, you will get a warning and DECsim may produce erroneous cancellations.

- DETECT/TOGGLE and DETECT/OSCILATION are not implemented for the faulty machine. These qualifiers work on the good machine only.

- The NOFAULT declaration in NETPRO, the SET FAULT/OVERRIDE, the SHOW FAULT/BLOCKER and SHOW FAULT/COUNT_FAULT_EFFECTS are implemented for gate level networks only.

- Fault dictionaries - This capability is not scheduled and will remain unscheduled until a user group asks for it and can define what they want.

- o The use of DETECT points in memory cells which are LOADed from an external file MAY produce inconsistent results if the LOAD command is done after the DETECTs are inserted.

  + WORKAROUND: Perform all LOADs into RAM models prior to inserting DETECT points.

11. Expressions with wide data are not implemented.

12. Initialization and assignment to wide ports is not completed or tested.

13. EVALUATE, EXAMINE and PRINT display wide data in hexadecimal if the radix is other than a power of 2.

14. Do not attempt to perform more than one SHARE operation within one DECsim session -- should the first one fail, subsequent SHAREs may also fail. To prevent this failure, DECsim will give an error message if SHARE is retried. Note that the global section file must be deleted when all programs mapping it exit.

15. The VT125 graphics control characters are not recognized by DECsim.

16. Concatenation of signal names is not generally available in the command language.

17. The SAVE command does not save indirect files.

18. If there are two or more SAVE commands, the command block around the SAVE command may get corrupted. Don't use SAVE/EVERY and then another SAVE command.

19. The RESTORE command has the following restrictions:

    1. You cannot RESTORE a network using a version of DECsim that is different than the one under which it was saved.

    2. You cannot RESTORE a network using a version of VMS different than the one under which it was saved, even if the DECsim versions are the same.

    3. RESTORE of a saved file on a machine other than the one the save file was created on may cause DECsim to crash.

    4. If you redirect SYS$INPUT (either by DECSIM/INPUT=file or with the VMS command ASSIGN/USER file SYS$INPUT) RESTORE will kick you back to the beginning of the file. This is likely to cause infinite looping.

# 4 NETWORK DESCRIPTION LANGUAGE

## 4.1 Introduction

This is the structural modeling language of DECsim - it allows you to define EQUATION, MOSFET, and MEMORY primitives and to interconnect gate-level, MOS-level, memory-level, and/or behavior-level models to create a model or a network. All model instances, whether they are instantiating structural or behavioral MODELs, appear in the network description language.

NETPRO is the Network Description Language compiler.

## 4.2 Differences Between General Release NETPRO V5.9-2892 and V6.0-2939

1. NETPRO has been ported to Alpha AXP OpenVMS. It is a native image on AXP. There is no difference in output file format between VAX and AXP versions. In fact, except for behavior models (which contain machine code), .NLB and .NOB files are fully portable.

2. NETPRO now flags an error if it detects that a network (or any one model in a model library) contains a mixture of behavior models compiled for the VAX and for the AXP. NETPRO also flags an informational message if it detects a behavior model that was compiled for the "other" architecture (i.e. not the one NETPRO is running on).

3. The algorithm for calculating the resolution has been changed slightly to guaranty that the VAX and the AXP versions of NETPRO will compute the identical result. Because of this, some delays (especially MOS delays) may differ very slightly from the numbers calculated by V5.9.

4. Due to an accidental change in the NOB file format, NOB files written by the field-test version of NETPRO are readable only by the field-test version of DECsim if the file contains any MOS models. NLB files are not affected. If you have any such NOB files created by the field-test (or beta-test) NETPRO, you must recompile them.

## 4.3 Summary of Features Implemented in this NETPRO Release

| Features implemented in this release | Yes |
|---|---|
| CAPACITANCE declaration | YES |
| Concatenation of signal names | YES (restricted by DECsim) |
| DEFAULT_TIMEOUT declaration | YES |
| DELAY declaration | YES |
| DELTA_L and DELTA_W for MOS | YES |
| Directives | NO |
| EQUATION statement | YES (see restrictions) |
| EQUATION statements with Feedback | YES |
| Error listing | YES |
| EXPLICIT_NODE declaration | YES |
| /EXTRACT compile switch (extract loadable network from model library) | NO |
| Generation and usage of model libraries | YES |
| GLOBAL declaration section | YES |
| INFORM statement and "!" comments | YES |
| Instances of models of any level complexity | YES |
| Instantiating and connecting Behavior models to other elements in a network. | YES (See restrictions) |
| LIBRARY and REQUIRE statements | YES |
| MACROs (Text) | YES |
| MAXRES declaration | YES |
| Mixed-MOS simulation | YES |
| Models defined within models to any depth | YES |
| MOSFET statement | YES |

| | | |
|---|---|---|
| Multi-section models | NO | |
| NO_CHARGE_SHARE declaration | YES | |
| OPERATOR statement | NO | |
| Parameters -simple use | YES | (no param expressions) |
| PARASITIC declaration | YES | |
| PCAP declaration | YES | |
| Phantom models (scalar)-Automatic insertion | YES | |
| Phantom models (scalar)-User re-definable | YES | |
| Phantom models (vector) | YES | |
| PHYSICAL statement | NO | |
| POWER statement | YES | |
| PRECISION,RESOLUTION statements (time-step values) | YES | |
| OUT_STUB,UNCONNECTED_OK,INPUT declarations | YES | |
| OUTPUT,IN_STUB,BIDIRECTIONAL declarations | YES | |
| RESISTOR statement | YES | |
| REVISION statement | YES | |
| Signals -handling of implicit "label.pin" | YES | |
| STRUCTURE sub-blocks- Any number in a model | YES | |
| Subscripts -full use of bit- and word-subscripts | YES | |
| SYNONYM statement | YES | |

| | | |
|---|---|---|
| THRESHOLD declaration | YES | |
| TIMEOUT declarations (for Gate-level) | YES | |
| TIMEOUT declaration (for MOS) | YES | |
| Wire Delay | NO | (See AP010.TXT) |
| Wirenames with embedded spaces, assertion and polarity | YES | (But see Commands) |
| Wire-tying signal declarations: WIREOR, WIREAND, TRISTATE, WIRETIE, PULLUP, PULLDOWN | YES | |

## 4.4 NETPRO Bugs and SPR's Fixed in this Release

NOB and NLB files do not need to be recompiled to be used with this version of DECsim.

1. Fixed a bug in reading .NLB files. These files are checked to see that the actual file size corresponds with size that NETPRO or SX originally wrote. A truncated file may be caused by an aborted copy over DECnet. NETPRO used to compare the allocated file size rather than the written size, causing a mis-match when a file is copied from one disk to another with a different allocation cluster size.

## 4.5 Known Bugs, Restrictions, and Deficiencies in NETPRO

1. There is no guarantee of upward compatibility for NETPRO output file(s) format.

   The files that NETPRO writes (library files and network object files) are not guaranteed to be upward compatible. In other words, future releases of NETPRO may require that these files be re-created by re-compiling them from their source files.

2. Restrictions on Hazards defined in NETPRO:

   - Hazards cannot be defined alone in a model (the model must contain some structural primitives).
   - The signals associated with the checker must exist in the context of the model.
   - Parameters are name-based only, signals can be name-based or order based.
   - Hazard checkers are not allowed on multibit vector signals bu are allowed on single bit vectors. Single bit vectors will automatically insert a split phantom between the vector and the hazard. You can make all vectors single bit vectors with WIRETIE_MODEL_LEVEL signal_name = GATE.

- o Hazard checkers are ignored in ZYCAD.
- o Hazard checkers are ignored in fault simulation.

3. Recursive macros are not detected and cause infinite recursion.

4. Syntax error correction attempts may cause infinite looping, regenerating the same sequence of attempted corrections.

5. Restrictions on interconnection of primitives: DECsim allows the interconnection of different types of primitives (EQUATIONs, MOSFETs, BEHAVIOR models, MEMORY models, etc.). There are some restrictions however:

   - o The use of concatenation ("A&B&C" etc.) in a network description to indicate the splitting or merging of different width signals must be used carefully. It is preferable to use subscripts (A<3:0>, etc.). See the appropriate Applications Note for a discussion of what is working and what is not working.

6. Conflicting down-level MOS declaration warnings can only be turned-off by a current level declaration that appears BEFORE the two down-level instances, not AFTER them. This contradicts the basic NETPRO philosophy that ordering of statements doesn't matter.

7. Semantics Error Messages.

Current Semantics Error messages have several problems:

- Not enough information is given in the message. For example, a semantics error message may indicate that a parameter i undefined, but it will not give the parameter name.

- The source line that is printed along with a semantics error may not be the line that caused the error. Some errors are not discovered until an entire sub-block is parsed. In this case, a set of error messages will be given, followed by the last line of the sub-block, followed by this message:

```
SUMMARY: The previous ERRORS or WARNINGS occurred in the
         statement, sub-block, or model ended by this line:
   In file DSU$:[HEILMAN.MISC]FOO.NET;5, line 1.
   EQN O=I;
   .......^
```

- In errors that deal with signals (INPUT declaration missing for FOO signal, for example), the signal name is printed. However, if the signal name has bit subscripts, the message will be replicated for every bit in the subscript range.

- In errors for label.pin format signals, the signal is printed in double quotes. The user should note that this would not be a valid source statement because the double-quoting would cause the dot to be interpreted as part of the signal name.

## 8. EQUATION statement

An equation must not have more than 6 inputs. Equations with more than 6 inputs must be factored by the user.

Every equation must evaluate to U#2 (undefined) when all inputs are U#2. An error message is given if this restriction is not met. This restriction is most likely to be violated when constants appear in the equation or when the CONVU1 or CONVU0 operators are used. Note that this makes an EQUATION expression where the output is set to a constant illegal: "OUT = Z#2" for example.

## 9. Quoting Labels

If a label on an instance needs to be quoted because it contains special characters, single quotes must be used. However, if the user needs to quote the same label in a label.pin format signal, he or she must use double quotes.

10. Unstable equations with feedback produce a warning stating that they may not function as expected. (The user may expect the equation to be re-evaluated when the output changes.) Equations with feedback are only evaluated ONCE PER INPUT TRANSITION in accordance with the given model. If we define a feedback equation as F(Out), then an unstable feedback equation is an F(Out) in which :

( F(Out) NEQ Out ) AND ( F(F(Out)) NEQ F(Out) )

is true for 1 or more input combinations. Example of an unstable equation:

```
EQUATION Out = In1 OR (NOT Out);
If Out is 0 then
F(0) is 1  ->  F(Out) NEQ Out
F(1) is 0  ->  F(F(out)) NEQ F(Out)
```

Thus this is an unstable equation which will produce a compilation warning.

# 5 BEHAVIOR MODELING LANGUAGE (BDS) COMPILER (SX)

## 5.1 Introduction

Behavioral modeling allows for more abstract definitions of the functions of a logic network. The language consists of both hardware description constructs (such as control structures, data transfers and operations) and simulator utilities (such as printing data and accessing VMS event flags).

SX is the compiler for behavior models written in BDS. SX generates BLISS code which in turn is compiled by the BLISS-32 compiler into VAX machine code or the BLISS32EN compiler into AXP machine code. SX also generates binary data which is used by NETPRO to instantiate models. Because Bliss is not always installed on Alpha AXP systems, your system manager may have to install Bliss from the kit located (at the time of this writing) at TLE::BLSALPHAN$V11_KIT:BLSALPHAN011.

## 5.2 Differences Between SX General Release V5.9-3047 and V6.0-3069

SX has been ported to Alpha AXP OpenVMS as a native image. The .NLB or .SLB files created by SX via the COMBEH command procedure are not portable between architectures. However, the intermediate output files written by SX are fully portable and may be used as the basis of a cross-compilation system for any user willing to work out the details.

## 5.3 Summary of Features Implemented in this SX Release

| Features implemented and supported in this release | Yes/No |
|---|---|
| ACTIVATE | YES |
| CONSTANTS: 2-state less than 32 bits | YES |
| 2-state greater than 32 bits, 4-state | NO |
| 4-state (0,1,U,Z) data from ports | YES |
| 4-state numbers | YES |
| 4-state arrays | NO |
| Conversion from 2-state to 4-state | YES |
| Conversion from 4-state to 2-state | YES |
| Delayed assignments (time delay directive) | YES |
| Directives: TIMESCALE | YES |
| %CARRYIN, %CARRY, %OVERFLOW (add, subtract) | YES |
| US (unsigned - default) | YES |
| TC (two's complement - 32 bits or less) | YES |
| TC (two's complement - wide, fourstate) | NO |
| WIDTH (all operations but shift and extension) | YES |
| HALT | YES |
| Model statistics printed at end of compilation | YES |

| | |
|---|---|
| Modifiers: INPUT, OUTPUT, | YES |
| NOENTRY, ENTRY, | YES |
| CONVU1, CONVU0, CONVZ1, CONVZ0 (2state ports) | YES |
| CONVU1, CONVU0, CONVZ1, CONVZ0 (4state to 2state) | YES |
| FOURSTATE, TWOSTATE | YES |
| BIDIRECTIONAL | YES |
| PULLUP, PULLDOWN | NO |
| TRANSPORT, FAULT, NOFAULT, DRIVER | NO |
| TRISTATE, WIREAND, WIREOR, WIREMOS | NO |
| INITIAL (2-state less than 32-bits) | YES |
| INITIAL (4-state, 2-state greater than 32-bits) | NO |
| INITIAL (states with word subscripts) | NO |
| INITFILE (file containing initial data) | NO |
| INITIALIZE (routine to initialize model) | YES |
| Calling non-BDS (BLISS, PL/I, C, etc.) routines | YES |
| %PRINT | YES |
| Print routine name during BLISS compilation (/DEB) | YES |
| ROUTINE, PORT, REVISION, STATE, SYNONYM | YES |
| Source-level debugging (WATCH/STATEMENT support) | YES |
| Time constants & units | YES |
| Time parameters in MODEL headers | YES |
| Time variables & expressions | YES |
| VMS synchronization built-in routines: | |
| %READ_EVENT_FLAG | YES |
| %SET_EVENT_FLAG, %CLEAR_EVENT_FLAG | YES |
| %WAIT_FOR_EVENT_FLAG | YES |
| WAIT time | YES |
| WAIT (time, FOREVER) (WHILE, UNTIL) condition | YES |

TABLE OF UNSIGNED BEHAVIORAL ARITHMETIC OPERATORS

| | TWOSTATE | | FOURSTATE | |
|---|---|---|---|---|
| OPERATOR | <33bits | wide | <33bits | wide |
| = (assign) | yes | yes | yes | yes |
| & (concat) | yes | yes | yes | yes |
| <>, () | yes | yes | yes | yes |
| BUF | nyi | nyi | nyi | nyi |
| NOT | yes | yes | yes | yes |
| AND | yes | yes | yes | yes |
| NAND | yes | yes | yes | yes |
| OR | yes | yes | yes | yes |
| NOR | yes | yes | yes | yes |
| XOR | yes | yes | yes | yes |
| EQV | yes | yes | yes | yes |
| + (add) | yes | yes | yes | yes |
| - (sub) | yes | yes | yes | yes |

| | | | | |
|---|---|---|---|---|
| * (mult) | yes | yes | yes | yes |
| / (div) | yes | yes | yes | yes |
| MOD | yes | yes | yes | yes |
| EQL | yes | yes | yes | yes |
| NEQ | yes | yes | yes | yes |
| LSS | yes | yes | yes | yes |
| LEQ | yes | yes | yes | yes |
| GTR | yes | yes | yes | yes |
| GEQ | yes | yes | yes | yes |
| EQL{QUOTE} | n/a | n/a | yes | yes |
| NEQ{QUOTE} | n/a | n/a | yes | yes |
| TS | n/a | n/a | yes | yes |
| MOSTS | n/a | n/a | yes | yes |
| TSCON | n/a | n/a | yes | yes |
| CONV0Z | n/a | n/a | yes | yes |
| CONV1Z | n/a | n/a | yes | yes |
| CONVZ0 | n/a | n/a | yes | yes |
| CONVZ1 | n/a | n/a | yes | yes |
| CONV0U | n/a | n/a | yes | yes |
| CONV1U | n/a | n/a | yes | yes |
| CONVU0 | n/a | n/a | yes | yes |
| CONVU1 | n/a | n/a | yes | yes |
| CONVUZ | n/a | n/a | yes | yes |
| CONVZU | n/a | n/a | nyi | nyi |
| CONV4 | n/a | n/a | nyi | nyi |
| MAP | n/a | n/a | yes | yes |
| WIDTH | nyi | nyi | nyi | nyi |

| | | | | | |
|---|---|---|---|---|---|
| ZXT | yes | yes | yes | yes | |
| SXT | yes | yes | yes | yes | (ZXT for unsigned) |
| OXT | yes | yes | yes | yes | |
| SL0 | yes | yes | yes | yes | |
| SR0 | yes | yes | yes | yes | |
| SR1 | yes | yes | yes | yes | |
| SLR | yes | yes | yes | yes | |
| SRR | yes | yes | yes | yes | |
| SLA* | nyi | nyi | nyi | nyi | |
| SRA* | nyi | nyi | nyi | nyi | |
| + (unary) | nyi | nyi | nyi | nyi | |
| - (unary) | nyi | nyi | nyi | nyi | |
| AND unary | nyi | nyi | nyi | nyi | |
| NAND unary | nyi | nyi | nyi | nyi | |
| OR unary | nyi | nyi | nyi | nyi | |
| NOR unary | nyi | nyi | nyi | nyi | |
| XOR unary | nyi | nyi | nyi | nyi | |
| EQV unary | nyi | nyi | nyi | nyi | |

---

NOTES:

yes tested and debugged on small models

n/a not applicable

nyi not yet implemented

### 5.3.1 SX Bug's and SPR's Fixed in this Release

- A bug was fixed where SX used to put UNIVERSAL state names in the DECsim symbol table, causing DECsim to crash.

- A bug in code generation was fixed. When two fields extracted from a wide result were multiplied (or concatenated) to produce a 32-bit or less result, SX would generate code that crashed. For example:

```
STATE X<63:0>, Y<32:0>;
ROUTINE Z;
   Y = (X)<63:48> _& (X)<15:0>;
ENDROUTINE Z;
```

### 5.3.2 Known Bugs, Restrictions, and Deficiencies in Behavior Simulation

1. Restrictions on UNIVERSAL STATEs

- Only simple assignments are supported.

  For example:

```
bds_state = %GLOBAL_VALUE(universal_state_name);
%GLOBAL_TARGET(universal_state_name) = bds_state;
```

- %GLOBAL_VALUE may not be used in a WAIT expression and cannot be a trigger for the WAIT condition expression.
- %GLOBAL_xxx is not allowed within a %PRINT statement.
- Bit subscripts are not allowed in the %GLOBAL_xxx expression (the full width is assumed).
- Bit extracts are not allowed.

  Workaround:

  Example:

```
UNIVERSAL STATE "%NET.I1.I2.X"[0:5]<31:0>;
UNIVERSAL SYNONYM syn_x [0:5]<31:0> = "%NET.I1.I2.x"[0:5]<31:
STATE temp<>, foo<>, bar<>;

    ROUTINE ...
        ...
```

```
! for read
        temp = %GLOBAL_VALUE(syn_x[foo]);
        out_port_name = {delay} temp<(bar sr0 2):4> + 7;
        ...

! for write
        temp = %GLOBAL_VALUE(syn_x[4]);
        temp<31:4> = 123#16;
        %GLOBAL_TARGET(syn_x[4]) = temp;
         ...
```

2. Restrictions on NONBDS STATEs and NONBDS CONSTANTs:

   - Both NONBDS STATEs and NONBDS CONSTANTs must be two-state and 32 bits wide.
   - There is no support for bit extract.
   - There is no access to non-bds states or constants from the DECsim command language.
   - %PTR is not allowed with non-bds states.
   - %LOC and %PTR are not allowed with non-bds constants.
   - A non-bds constant cannot appear in the declaration of a BDS state as its initial value.
   - Names for non-bds states and non-bds constants may not be the same as a BLISS reserved keyword.
   - If a non-bds constant is declared as a globalvalue in 'C', it must be declared as a NONBDS CONSTANT in BDS and not as a NONBDS STATE.

3. SELECTONE and SELECTALL do not work for wide data.

4. When the network processor links (with the VAX/VMS LINK utility) the object code for those models will generate errors (multiple definitions of global symbols) if:

   - A behavior model is instantiated twice in a network, and the 2 copies were compiled at different times of the day or on different days.

   - There are 2 models that access the same module.

5. The loop variable in a FOR loop statement must be 2-state.

6. 4-state arrays are not yet implemented. If declared an error message will be issued.

7. The module name and the source file name (not including device, extension, etc.) containing the module must be the same. When declaring the module name in the master model a full VMS file specification is allowed. If just the module name is used, DECsim

assumes that the module resides in the same directory as the main model. VMS logical names may also be used if it is necessary to point to another directory.

8. Recursive macros are not detected and cause infinite recursion.

9. A WATCH on wide data will trigger as each longword state gets written - which means that the first wakeup will have intermediate data.

10. Syntax error correction attempts may cause infinite looping, regenerating the same sequence of attempted corrections.

11. The CONSTANT declaration does not work for 4-state data.

12. Subscripts may not be specified in CONSTANT declarations or accesses. (An error message is generated.)

    Suggested work around: Bits of the constant may be accessed with the (expr)<bit_sub> construct. The low bit is assumed to be <0>, the high bit is assumed to be <width-1>. For example:

```
CONSTANT MAGIC_NUMBER = 76543210#16;
STATE MAGIC_NUMBER_FIELD<31:0>;
  ...
MAGIC_NUMBER_FIELD = (MAGIC_NUMBER)<19:12>; ! 8 bits, 43#16
MAGIC_NUMBER_FIELD = (MAGIC_NUMBER)<4>;     ! 1 bit, 1#2
```

13. The %PRINT directives WIDTH, RADIX, ZEROFILL, NOZEROFILL for

4-state data, and for 2-state data greater than 32 bits wide are ignored.

14. All expressions and states may be up to 256 bits wide with the following exceptions:

    SELECT value FROM .... value should be <= 32 bits.
    FOR variable FROM .... variable should be <= 32 bits.
    Values in bit and word subscripts should be <= 32 bits.

15. Ports must be declared with bit subscripts. Word subscripts are not allowed on ports.

16. Delayed assignments may only be made to an entire port and/or global state. An error will be given if bit subscripts are found on the destination of a delayed assignment.

17. Routines may not have the same name as some other global objecu such as a port, state, constant or synonym.

    Suggested solution: Either change the name of the routine or

change the name of the other global object.

18. If no behavior subblock name is given, the model name must be distinct from all other identifiers (except macro names) declared in the subblock. Otherwise, NETPRO will crash when the model is instantiated. If the model name or subblock name is the same as a macro name, the ENDMODEL or ENDBEHAVIOR construct will have syntax errors or a mismatching name semantics error.

19. There is a limit on routine size and nesting (parentheses in expressions). The limit is not determined by SX but is determined by BLISS. With version 4.4 of BLISS, routine size has increased from around 300 statements to around 750 statements.

# 6 REPORTED SPR'S STILL OPEN

The following section describes known bugs and deficiencies in DECsim. There is a NOTES conference on node DECSIM called DECSIM_BUGS. You may now report SPR's by adding notes to this conference. Before adding a new note please check to make sure that your problem has not already been reported. If you have more information about a reported bug you may wish to post a reply. Suggested workarounds may also be posted as replies.

The following is a list of reported open SPR's. The numbers correspond to the note number.

#7 When scoping down in your network, some or all of the signals in the access list may be reported twice when you type examine "*".

#15 Negative numbers are not allowed in behavior word subscripts
for example: STATE www[-2:4]<31:0>; ! not allowed

Suggested Workaround: STATE www[FFFFFFFE#16:4]<31:0>;

#20 When compiling in NETPRO, a delay parameter triple out of order gives no indication of which model is incorrect.

#21 Request for operators on wide data in the command language

#22 Inconsistency with LOAD/EXE and LOAD/TEXT for subscripted variable limits.

#24 LOAD/EXE range checking error.

#31 Enhancement request for the command language-- User requests the ability to apply PATTERN's, trigger TRACE/WHEN's and DETECT/STROBE's off command language states.

#32 Enhancement request for DEPOSIT/FORCE on fourstate PORTs.

#35 Request to define command language macro's within macro's

Suggested Workaround: Have the secondary macro definitions in a .ind file and have the .ind file called from the primary macro.

#36 Request for expression calculation for NETPRO's delay parameters.

REPORTED SPR'S STILL OPEN

Currently they must be constants.

#38 The LOAD/MICRO command ignores the /INITIAL qualifier.

#40 Signed arithmetic in behavior models does not work.

#41 Behavior print width problem.

#46 A divide by 1 in a behavior model causes a BLISS compile time error.

#49 Request for separate compile and link when making changes to nonbds code surrounding network.

#50 Request for DECsim behavior to work with object libraries for nonbds code.

#53 Request access to the simulator clock to find out how long things have been running.

#61 Request for run-time evaluation of the command language macro template $EXPRESSION.

#62 Problem with behavior and magnitude of delayed assignments.

#72 Behavior bug on division of wide-data delay.

#73 Bad 4-state BLISS code generated from behavior model.

#74 Behavior models do not seem to perform bit field insertion and extraction when reading and writing to ports in the network.

#75 In behavior, %PRINT{RADIX=10}... does not work for local 4-state variables.

#77 Macro delay expression crashes SX.

#83 Behavior print does not allow composed characters in a text string.

#94 The "timescale" directive in SX doesn't work.

#100 The SLR and SRR should work with negative rotations.

#103 Mixed mode state observation problem.

#106 EXAMINE/Separator does not apply to between array elements.

#109 BDS symbols of more than 135 characters are arbitrarily truncated to 135 characters without any warning or error message from SX.

#112 The following model causes DECsim to access violate in the ARTH$WID2US_VALUE routine. The result of the BEGIN-END block is a 32 bit result, but SX thinks that it is 64 bits because of the first expression.

```
MODEL bug=;
BEHAVIOR;
STATE
     m<>, p<>, wide<63:0>;

ROUTINE r;
m =      BEGIN
         p<31:0> = wide;
         %PRINT ('Before assignment');
         NOT (p<1> XOR p<0>);
         END;
```

```
ENDROUTINE r;

ENDBEHAVIOR;
ENDMODEL bug;
```

#114 The %LOC operator in SX is not consistant between arrays and states when calling Non-BDS routines.

#117 If the same routine name is used in two different modules, SX will crash when compiling the top module.

#118 If the first module in a modular model does not contain any routines compiling the top modular model gets an error.

#121 If the line: %print (,'foobar'); appears in an SX source, SX crashes (because of the illegal comma).

REPORTED SPR'S STILL OPEN

#126 The LOAD command doesn't seem to handle hierarchial destinations properly.

#134 Request for DECsim to enhance SHOW CALL to display the names of the WATCH commands currently executing.

#135 The SPR command resets the window size to 24 lines

#137 An SX dump - caused by a problem with allocation of temporaries. SX is trying to use one that is not available. A workaround is posted in the reply to this note.

#138 Enhancement Request to allow users to find out if states have U's and Z's in them. exit

#144 Request for SHOW STATE to accept an argument so that User may limit the display to one or more states.

#148 Request to implement EXAMINE/PARAMETER for Behavior models.

#152 SX should catch non-function calls.

#156 Bug in using the value of a WAIT statement as the value for a delayed assignment. For example, X = {0} WAIT 1;

Workaround:

```
y = WAIT 1;
x = {0} y;
```

#160 Bug in subscripting an operation result in SX. If you have a statement of the form:

```
state_variable_name = (port_symbol XOR 3)<3:0>;
```

no compile time error is issued, but DECsim will crash when this statement is executed.

#167 Wide constants use the wrong radix. When you examine a behavior language name defined as a CONSTANT in your BDS file, the value is printed in decimal radix by default if the width is 32 bits or less. However, it is printed in radix 16 if the width exceeds 32 bits and no indication is given that the radix is NOT decimal (unless A-F shows up as a digit).

#168 SX doesn't print the source line properly with nested macros. When

you have nested macros in BDS code, and there's an undefined identifier, SX does not point to the macro expansion line -- making it difficult to track down.

#170 Enhancement Request: The user would like arguments to a macro to be optional such that any unspecified at macro call time would default to some known null value.

#171 Enhancement Request: would like an active IND file after a RESTORE command.

#183 Crash if too many SELECTs trigger.

#187 BLISS warning caused by (x SL0 0)

#189 Problem with DCSCMOS4.PRC -- FET resistance values are lower than the SPICE result.

#195 SET WATCH/ONCE/EVERY in a watch body results in the /ONCE getting deleted before the WATCH is executed.

WORKAROUND: Take off the /ONCE and then add a CANCEL WATCH within the one time WATCH body.

#203 Problem with a BDS wide array used within parentheses.

#213 DECSIM/INPUT=filename goes into an infinite loop (loops from the first RESTORE command back to the beginning of the file).

#215 NETPRO doesn't fully accept "$" in labels. NETPRO accepts labels that begin with "$" but does not accept labels that begin with a digit followed by a dollar followed by more stuff.

# 7 USER DOCUMENTATION

## 7.1 Introduction

The DECsim documentation includes hardcopy documentation, online text files, Bookreader files, and NOTES file conferences. To get hardcopy documentation, send requests to CADSYS::DOCUMENT. All online text files are located in DECSIM$SYS on any node that has an installation of DECsim. The NOTES file conferences are located on node DECSIM.

Bookreader versions of the DECsim documentation are now available in CADBKS, the new CAD Tool Bookreader kit. See DECSIM note #38 for current information and how to get this kit. CADBKS includes the following DECsim manuals:

"DECsim User's Guide"
"DECsim Reference Manual"
"DECsim Messages"

There is also a DECsim course taught upon sufficient demand. Information about course dates and registration is available from SHARE::REGISTRAR or you can contact your local training group.

The following is a list of the DECsim documentation:

1. "Introduction to DECsim" (hardcopy)
   This manual gives you a basic idea of what DECsim is all about. It introduces the first time user to the concepts and conventions of digital logic simulation. It explains how to set up a simple

network, get it ready for simulation, and do some testing.

Manual last issued -- Jan 1985 -- superseded 3.0
VAX/VMS version 3.6 or higher, DECsim version 4.1

2. "DECsim User's Guide" (hardcopy and Bookreader)
This manual presents more detailed information and strategies for applications of DECsim to a design project.

Manual last issued -- May 1990 -- superseded 5.2
VAX/VMS version 4.4 or higher, DECsim version 5.6

3. "DECsim Reference Manual" (hardcopy and Bookreader)
The reference manual is everything you ever wanted to know about DECsim but have to know how to ask for. This manual gives specific information about the language syntax and semantics and some typical applications. It is organized by feature and not by task.

Manual last issued -- May 1990 -- superseded 5.4
VAX/VMS version 4.4 or higher, DECsim version 5.6

4. "DECsim Messages" (hardcopy and Bookreader)
This manual lists DECsim Command Language and SX messages. Each message is accompanied by an explanation and, where applicable, a suggested recovery procedure. (Note that this information is also in DECsim help (HELP MESSAGES).)

Manual last issued -- May 1990 -- superseded 5.4
VAX/VMS version 4.4 or higher, DECsim version 5.6 or higher

5. "DECsim Pocket Reference Guide" (hardcopy)
This manual provides a synopsis of DECsim commands and shows syntax in the form of railroad charts.

6. DECSIM.DOC or DECSIMxx.RELEASE_NOTES (online text file)
Where xx is the current version number.
This is a file of release notes and information for the current version of DECsim. This file is issued and updated with every release. To read this file type DECSIM$SYS:DECSIM.DOC.

7. ANINDEX.TXT (online text file)
This file is an index of the available DECsim Application Notes. For a list of application notes type DECSIM$SYS:ANINDEX.TXT

8. DECSIM NOTES CONFERENCE
This is a VAX/VMS NOTES Conference which serves as a forum for the exchange of information among DECsim users and developers. This includes answering questions, documenting work-arounds, release

announcements, etc.

You can add this conference via the NOTES command:

```
NOTES> ADD ENTRY DECSIM::DECSIM
```

9. DECSIM_BUGS NOTES CONFERENCE

This is a VAX/VMS NOTES Conference to address software problem reports (SPR's). Solutions and/or workarounds are posted as replies.

You can add this conference via the NOTES command:

```
NOTES> ADD ENTRY DECSIM::DECSIM_BUGS
```