

The CAD/CAM Group

Engineering Capture System Release Notes

Version 2.4

February 1992

20480 Pacifica Drive, Suite B, Cupertino, CA 95014, ph 408 725-0204, fax 408 725-0207
email uunet!ecs!support

Welcome

Please take a moment to check that you have received everything needed for this 2.4 release. The contents of each release package are described below for the various platforms.

Please notice that we have included forms at the end of this section for reporting any bugs to us that you may find. We appreciate in advance your diligence in reporting bugs or problems so that we can continue to provide you with a quality, current, and bug-free software product.

SPARC Architecture Machines

You should have received the following materials. If anything is missing, please contact the CAD/CAM Group Customer Support Organization.

- 1 1/4 inch cartridge tape
- 1 set of release notes

IBM Compatible PCs

You should have received the following materials. If anything is missing, please contact the CAD/CAM Group Customer Support Organization.

- 2x 5 $\frac{1}{4}$ " 1.2MB floppy disks containing the ECS executable files
- A floppy disk containing the ECS libraries
- 1 set of release notes
- If you just purchased the extended version, one hardware protection key

Macintosh Computers

You should have received the following materials. If anything is missing, please contact the CAD/CAM Group Customer Support Organization.

- 800k floppy disks containing the ECS application files
- 800k floppy disks containing the ECS libraries
- 1 set of release notes

Contents

Welcome	2
SPARC Architecture Machines	2
IBM Compatible PCs	2
Macintosh Computers	2
Contents	3
Update Checklist	4
Supported Configurations	5
SPARC Architecture Machines	5
IBM Compatible PCs	6
ECS on Macintosh	7
ECS Installation Procedure for Sun Workstation	8
ECS Installation Procedure for IBM PC	15
New Features in Release 2.4	17
ECS Shell	17
Hierarchy Navigator	17
Schematic Editor	19
Symbol Editor	22
Waveform Tool	23
ecs.ini Editor	24
Interfaces	25
Notes For MOTIF Users	28
Further Explanation of Master Symbols	28
Bug Fixes in Release 2.4	29
Reporting New Bugs	30

Update Checklist

1. Read the entire Release Notes package carefully and note all the changes that may affect your installation.
2. Follow the Installation Instructions and install version 2.4 on your system.
3. Every user should read the entire Release Notes package *before* using version 2.4.

Supported Configurations

The Engineering Capture System is supported for the following hardware and software configurations.

SPARC Architecture Machines

The following description describes the minimum hardware and software requirements for running ECS reliably on SPARC architecture machines.

Hardware Configuration

ECS is designed to run under SunView or MOTIF on Sun workstations. A minimum configuration is suggested here to ensure reasonable performance and compatibility. This configuration includes:

- Any SPARC architecture machine
- A minimum of 8MB of memory
- A hard disk with at least 20MB available for ECS programs, symbol libraries, and data files. The installation tape includes about 50MB, much of which can be deleted depending on your needs.
- A minimum swap space of 20MB for UNIX and ECS
- A color or black-and-white console monitor
- A three-button mouse
- A 60MB 1/4-inch cartridge tape drive
- A PostScript printer such as the Apple LaserWriter II (if you want to print)

Software Configuration

ECS is designed to run using the SunView graphics environment or MOTIF under SunOS. All of the files on the distribution tape, including the example schematics, take up 50MB. You need to ensure that the target directory has access to at least this much storage space.

The following software configuration is required.

- SunOS 4.1.1 or later
- SunView graphics environment or X11R4 running with the MOTIF window manager
- The ECS software is not officially supported using the OpenLook™ window manager but the CAD/CAM Group's experience so far indicates that the ECS software runs successfully in this environment. The fonts necessary to operate in the OpenLook environment are supplied.

The ECS software can run using the MOTIF window manager. If the ECS is running on a SPARC machine, other workstations of various types may open an X window

session and access the ECS software over a network. Fonts are supplied for the MIT window manager on Sun workstations, the IBM window manager for IBM R6000 stations, DECstations, and the Silicon Graphics Indigo workstation.

IBM Compatible PCs

This section explains the necessary hardware and software requirements to run the ECS on an IBM compatible PC.

Hardware Configuration

ECS is designed to run under the Microsoft Windows 3.xx operating environment on IBM compatible PCs. A minimum configuration is suggested to ensure reasonable performance and compatibility. This configuration includes:

- An 80286, 80386 or 80486-based CPU with at least 640k bytes of memory. Additional memory can improve performance by configuring it as a RAM disk, disk cache, and/or expanded memory. See the *Windows Users' Manual* for instructions on using these features.
- A hard disk with at least 6MB available for the ECS programs, symbol libraries, and data files. This can be reduced accordingly if the PCB libraries and/or some of the utility programs and netlisters are not required.
- A monitor. Monochrome displays must operate in a graphics mode. The IBM monochrome adapter or equivalent is not acceptable.
- A two- or three-button mouse.
- A hardcopy output device (if you want to print). Driver programs for some devices are included in the Windows system. Driver programs for other devices are available from hardware vendors. Most dot matrix printers and laser printers are supported. Plotters using Hewlett Packard Graphics Language (HPGL) interface are also supported.

Both the standard and extended versions of ECS on the PC require the above configuration. The standard version supports designs that range in size between 3k and 5k primitives, depending on how many pins, attributes, instances and other memory consuming items are used in a design.

The extended version has modifications that support much larger designs. Note that a minimum of 3MB of RAM and 3MB of free disk space is recommended for large designs.

It is recommended that you run Windows in protected mode. The performance increase over real mode is substantial. Running in protected mode requires a 386 or later machine.

Software Configuration

ECS runs under DOS using Microsoft Windows. The following software configuration is required.

- DOS or MS_DOS Version 3.xx or higher. A higher version may be required for hardware compatibility (PS/2). DOS 5 generally provides better performance by freeing more of the bottom 640k of memory.
- Microsoft Windows Version 3.xx.
- A Windows Driver program for your graphics adapter and display. Microsoft includes several drivers in the Windows system. Drivers are also available from the vendor of your graphics adapter board.

ECS on Macintosh

This section describes the hardware and software requirements necessary to run the ECS on a Macintosh computer

Hardware Configuration

ECS is designed to run under MacOS on Macintosh computers. A minimum configuration is suggested to ensure reasonable performance and compatibility. This configuration includes:

- Any Mac SE or Mac II (Mac II is recommended)
- A minimum of 2MB of memory
- A hard disk with at least 10MB available for ECS programs, symbol libraries, and data files
- A color or black-and-white monitor
- A single-button mouse
- A 3-1/2-inch floppy disk drive
- An Apple compatible printer such as the Apple LaserWriter II or ImageWriter

Software Configuration

ECS is designed to run under MacOS. The following software configuration is required.

- System Tools 6.0.1 or later
- MultiFinder 4.3 or later
- Finder 6.1 or later
- ECS is compatible with System 7 from Apple. For best performance do not use *Calculate Folder Sizes* from the **Views** Control panel. Use of this option leads to long cursor response times in the ECS.

ECS Installation Procedure for SUN Workstation

This software runs on Sun 4 machines running SunOS 4.1.1 or later.

This tape includes the following directories:

bin_sunv	sunview version of ECS for Sun 4 machines
bin_motif	motif version of ECS for Sun 4 machines
data	the data directory including message files, help files and the ecs.ini configuration file and simulator.ini files.
examples	some example design files
mod_libs	ECS model libraries (schematic files)
sym_libs	ECS symbol libraries
dots	some initialization files for X windows
fonts	the fonts used by ECS under sunview
xfonts	the fonts used by ECS under motif
xfont???	the fonts used by ECS under motif, specific to certain servers. The ??? are replaced by MIT, XNW or the appropriate characters taken from the display server string.
license	contains the license file and license manager daemons
pc	contains the latest release of software on the PC (Included only if you have ECS on PCs)
pc_pik	contains the PIK libraries and sample code for the PC (Included for PIK customers only)
pik4	contains the UNIX PIK libraries (for PIK customers only)
pik_src	contains sample code for the PIK (for PIK customers only)
Xfiles	Some files needed when running X-windows. Specifically XKeysymDB should be put in the libraries directory (i.e., the path pointed by LD_LIBRARY_PATH env variable) if one does not exist already.

Some of these directories may not be included on your tape depending on the software you purchased.

The following instructions describe the installation procedure for ECS running under sunview. Many of the procedures are common for the X Windows version. In addition, there is a file called README.motif with instructions specific for motif.

Setup and Installation Procedures for ECS

1. Create the ECS_ROOT environment

The ECS system is installed under a directory structure which localizes all of the application files as a tree structure below the main installation directory which is referred to as ECS_ROOT. The executables, symbol & model libraries and user designs need not be a part of this tree structure, but it is highly recommended that this

structure be followed during the installation. However, the "data" and appropriate font directories MUST be located under the ECS_ROOT directory.

As SUPERUSER (root), create a directory to contain the ECS system. The name and location of this directory is optional. Set the permissions on this directory to permit access. You may want to check available disk space using the "df" command. ECS installation requires about 30 MB of disk space and the libraries require about 10MB.

Typical command sequences might include:

```
cd /home  
mkdir /home/ecs  
chmod 775 ecs
```

Note: '/home' represents an arbitrary path.

If you already have a previous version of ECS and want to overwrite it, you may choose the same directory for installation. However, the names of some files may have changed, so some of the old files will be left behind. It is advisable to delete these files or install ECS under a different directory

If you are overwriting the data directory and have customized your ecs.ini file and other simulator.ini files, back them up and merge in the new features provided in the .ini files which accompany this version of software.

2. Loading the tape

Set the current directory to the directory created in step 1.

```
cd /home/ecs
```

Use the tar command to load the tape into this directory

```
tar xvbf 4000 /dev/rstn
```

(Replace rstn with the device name for your tape drive.)

The tape should load creating a directory structure of:

```
/home/ecs  
/home/ecs/license  
/home/ecs/data  
/home/ecs/fonts  
/home/ecs/bin_sunv  
/home/ecs/bin_motif  
/home/ecs/examples  
/home/ecs/examples/...  
/home/ecs/sym_libs
```

```
/home/ecs/mod_libs
```

```
.  
.   
.
```

If you also purchased the Programmers interface kit, the structure also contains the directories:

```
/home/ecs/pik_src
```

```
/home/ecs/pik4
```

3. Preparing to run ECS

Before running ECS you must set an environment variable to point to the directory structure containing the system. The .login or .cshrc file in your home directory may be modified to include the following command: `setenv ECS_ROOT /home/ecs`

The path must also be modified to contain the appropriate directory for the system architecture. Add the path `/home/ecs/bin_xxx` to the `set path=` line in the .login or .cshrc file in the home directory.

```
set path=(. ... /home/ecs/bin_sunv) # for sunview version
```

```
set path=(. ... /home/ecs/bin_motif) # for x-windows version
```

Check that you have removed earlier versions of ECS referenced in the `path` before this directory. Once these modifications are made, these commands may be invoked by logging out and logging back in or by using the **source** command:

```
source ~/.cshrc
```

```
source ~/.login
```

As an alternative, the `setenv` and `set path` commands may be issued on the command line in one of windows in which you plan to run `ecs` or in the login shell . It may also be necessary to issue the 'rehash' command if this method is used.

4. Unpacking the Model and Symbol Libraries

The symbol and model libraries are transmitted in 'ar' format to conserve space and loading time. The 'listall' and 'expand' scripts in the `mods` and `libs` directories are used to list and setup the libraries. Read the comments in the script files for additional information.

Once the libraries are set up, the configuration file, 'ecs.ini' must be updated to reflect the library's locations. After starting the ECS Shell, use the `Setup/ecs.ini` command to invoke the editor. A default `ecs.ini` file supplied with the software has library paths set up relative to `ECS_ROOT` and may be modified with the editor.

5. Security

The ECS security has changed as of 12/20/91. Old keys will not work for the new software. You must load the new license file.

The ECS security system on the Sun uses Highland Digital Network license software. If you are using any other software that uses the Highland Digital license manager, you must merge the ECS license file *license.dat* with that from your other software (i.e., concatenate the two in any order.) You need to remove replicated SERVER lines and leave only one.

The ECS system is shipped with a license directory. This directory contains five files:

lmgrd	license manager daemon
ccgd	version 2.4 ECS licensing daemon
ccgid	version 2.21 ECS licensing daemon
README	describes security installation
license.dat	the encrypted license file

The file *license.dat* should be copied to the directory */usr/local* on each of the machines which will be running ECS. If you would rather locate it elsewhere, add a *setenv* command in the *.login* or *.cshrc* file which defines an environment variable as:

```
setenv LM_LICENSE_FILE full_path_name_of_license_file
```

i.e., */home/myhome/license.dat*

Install the license manager daemons on the server machine. The server machine's *hostid* should be listed on the 'SERVER' line of the *license.dat* file. (Note that "server" here denotes the network server and not the X display server). For this, you need to:

1. Copy the appropriate daemons to the */etc* directory on the server machine. You will have to be logged in as *root* to write in this directory.

```
cp -p lmgrd /etc/lmgrd      (sun4 server)
cp -p ccgid /etc/ccgid
```
2. Start the daemon by issuing the command:

```
/etc/lmgrd > /dev/null &
```
3. Edit the startup file */etc/rc.local* and add the following lines at the end so that the daemon starts up every time the machine is rebooted:

```
if [ -f /etc/lmgrd ]; then
    /etc/lmgrd > /dev/null & echo "Starting License Manager"
fi
```

Exit *root* and login as a user. Start the ECS system.

Note that the 2.21 version of ECS uses a different daemon (*ccgid*) than the 2.4 software (*ccgd*). This allows both versions of software to be licensed at the same time.

Running ECS under sunview

ECS (sunview version) requires the Sunview environment to operate. Start sunview by issuing the command **sunview** from the unix shell. This command may also be added to the .login script.

Start the ECS shell by typing **ecsshell**

Note that this process automatically runs in the background, leaving the parent shell free for the user.

Running ECS under X-Windows

ECS (Motif version) requires the X window System Release 4 or later to operate. ECS is designed to run under the OSF Motif window manager. This is the window manager under which the ECS software is officially supported by the CAD/CAM Group.

We have found in preliminary testing that the ECS also works with other managers including the Sun OpenLook window manager, the Tom's window manager and the 4-D window manager from Silicon Graphics.

Before running ECS, there are a few things that you need to set up which differ depending on your X-server. Some of these are mentioned below. It is assumed, at this point, that X11 System Release 4 or later (or a window system like X11/NeWS which follows the X11 protocol) is already installed and you are running Motif or a similar window manager.

(A) If your window manager is Motif and you are using the X11R4 server:

1. Add a 'setenv' of:

setenv LD_LIBRARY_PATHMotif/lib4

This should point to the directory where the Motif libraries are stored. The X11 binaries should also be referenced in your path.

2. Copy the files in 'dots/mwm' into your home directory:

.xinitrc startup file for 'X'

.Xdefaults Default settings for applications

.ccglogo Background for screen

If these files already exist in your directory, you may want to make a backup so that you can merge the commands from the shipped files with your existing ones.

3. If your display server is not a Sun machine, the fonts supplied with ECS may need to be rebuilt.

ECS uses its own set of fonts which are placed under the ECS_ROOT directory.

The raw fonts are supplied in the Bitmap Distribution Format (".bdf" extension files) in the "xfonts" directory. The working fonts need to be in the

Server Natural Format (".snf" extension files) for your particular server. The compiled snf fonts for some vendors are shipped in the directories xfontMIT, xfontSG, xfontXNW, etc.

The directory names are generated from the X ServerVendor string. The first 3 uppercase letters in the string are appended to the fixed 'xfont' string.

On Sun platforms where there is a choice of X server, ECS uses the directory named xfontMIT if the standard MIT server is being used and the directory xfontXNW if Openlook (X11/NeWS server) is being run. (See OpenLook set up below.)

On the Silicon Graphics Iris display server, the directory xfontSG is used.

If you do not have any machine of a particular manufacturer in your network, then the corresponding directory may be deleted.

If you have a custom server on your machine other than the ones for which compiled fonts have been shipped, you should still be able to create another directory with the name built up as above, and install the fonts using the "bdf2snf" and "mkfontdir" X processes.

A sample script called INSTALL is provided for this purpose in the xfonts directory. Note that the directory with the ".snf" fonts you have created should be named appropriately as xfontXXX (where XXX are the first 3 uppercase letter in the X ServerVendor string). The script does not do this. You may have to use the vendors equivalent procedures if the "bdf2snf" and "mkfontdir" are not part of your version of X. If there is no change in font formats, you may just link the directory name (xfontXXX) to the existing directory for one of the compiled fonts.

(B) If your window manager is TWM / TVTWM and you are using the X11R4 server:

1. Add a 'setenv' of:

```
setenv LD_LIBRARY_PATH ....lib/X11
```

This should point to the directory where the std X libraries are stored. The X11 binaries should be referenced in your path.

2. Copy the files in 'dots/twm' into your home directory:

.xinitrc startup file for 'X'

.Xdefaults Default settings for applications

.csglogo Background for screen

If these files already exist in your directory, you may want to make a backup so that you can merge the commands from the shipped files with your existing ones.

3. Make sure that the directory xfontMIT exists below the directory you have selected for ECS_ROOT. For servers on non-Sun machines follow the steps in (A) for making a font directory, if needed.

(C) If your window manager is OpenLook (olwm) and you are using the X11/NeWS server:

1. Add a 'setenv' of:

```
setenv LD_LIBRARY_PATH ....openwin/lib
```

This should point to the directory where the olwm libraries are stored. You may need to setup the the path to the OpenWindows binaries in your path if you have not already done so. If your OpenWindows software is not installed in /usr/openwin you will need the environment variable OPENWINHOME set to point to the directory where it is located. Refer to your OpenLook documentation for details.

2. Copy the file in 'dots/openwin' into your home directory:

.xinitrc startup file for 'X', starts up olwm

.Xdefaults Default settings

.ccglogo Background for screen

Note that if these files do not exist, the start-up script (openwin in \$OPENWINHOME/bin) will create default copies in your home directory, so ECS will start-up without problems except for differences in color and window positions.

If you already have existing files by this name, back them up or merge in the new commands.

3. Make sure that the xfontXNW directory exists below the ECS_ROOT. If you have problems with fonts and need to build them you may use the script INSTALL.XNW under the xfonts directory or do so at the command line with the programs "convertfont" and "bdlfamily" and rename the directory as xfontXNW.

(D) If you have a different window manager / server, please contact the CAD/CAM Group for more information on the setup requirements. We will attempt to help you although as mentioned above, we are primarily supporting the OSF MOTIF window manager.

After you have set up the above configuration, you are ready to start the X-windows version of ECS:

1. Remove the sunview call in your .login file if it is present.
2. Start X11 with the line 'xinit'
3. Start ECS with a call to 'ecsshell' on the command line.

For customizing colors, text fonts, default window sizes etc., you will need to modify the ".Xdefaults" file. The appropriate resource names are listed in the ECS User's Manual. However, ECS works with default values even if no resources are specified in the .Xdefaults file. For customizing the auxiliary shell window, set the resources for xterm in the .Xdefaults file instead of the ECS resources.

ECS Installation Procedure for IBM PC

Follow the procedure below to install the version 2.4 of ECS software on the PC.
Read the complete installation procedure before beginning the installation.

1. Backup your designs from previous versions of ECS software. It is always good engineering practice to back up your designs before installing new software.
2. Backup any *.ini* files that you currently use. You may wish to save your old *ecs.ini* file as it contains any special attributes that you created.

If you install this version of the software over an existing version, the *ecs.ini* and *simulator.ini* files provided with this copy of *ecs* may overwrite any earlier *.ini* files. You should compare your old *ECS.INI* file with the new file to ensure that any new switches or attributes from the latest *ecs.ini* file are merged into your old *ecs.ini* file.

You are given a chance to create a backup data directory before your existing data directory. If you choose to overwrite the existing data directory, any *.ini* files found in the data directory are copied and given file names with the extension *%%%* during the installation.

3. Insert ECS Program disk #1 in your floppy drive.
4. To start install, select the *RUN* command from the Program Manager's File Menu. Enter the following as the command line when the dialog box comes up:

A:SETUP (replace A: with the floppy drive you are using)

5. The installation program checks to see that you have enough disk space and prompts you when needed. Keep an eye on the Information window for additional comments and tips. A program group named *ECS Release 2.4* and a program item for ECS are created.
6. You need to modify your search path and create an environment variable for ECS. You can either let the installation program add these changes to your *autoexec.bat* file automatically or follow the directions below.

After Installation is complete:

After the installation is complete, follow the procedure below to ensure proper operation of the Engineering Capture System.

1. If a protection key was supplied with your software, insert the protection key into the parallel port of the computer. Note that this will not affect the operation of the printer in any way.
2. If you have not allowed the installation program to modify your *AUTOEXEC.BAT* file, you must do so yourself to include the following.

- A Set your path to include the directory containing the ECS executables. The path is typically modified in the autoexec.bat file so that each time you reboot, you have the correct path.
- B Set the path to include the windows 3.0 directory. The windows directory should follow the ECS directory in the search path.
- C Set the environment variable ECS_ROOT to point to the directory containing the ECS executable files. For example, if the directory containing the executable files is C:\ECS24, then add the following line to your autoexec.bat file

set ECS_ROOT=C:\ECS24

This statement **MUST** come before any batch file calls in your AUTOEXEC.BAT file or it will not work.

- D Set the environment variable TEMP to point to a scratch directory. This is normally created during a proper Windows installation and if it already exists, continue to step 3.

If you have a scratch directory called C:\TEMP, add the following line to your autoexec.bat file.

set TEMP=C:\TEMP

3. REBOOT YOUR MACHINE TO INVOKE THE NEW AUTOEXEC.BAT FILE.

Note: If you get the message "Out of environment space" while booting your system, you can extend the environment space by inserting the line into your CONFIG.SYS file:

shell c:command.com /p/e:512

Installation is now complete.

Running ECS

To start the ECS program, use the following procedure.

1. Start windows by typing win at the DOS prompt.
2. If you have installed the DCT, a Program item called ECS 2.4 is created and double clicking on this icon starts the ECS Shell.

Alternatively, you can type ECS from the RUN menu of the program manager. This starts the ECS Shell from which all the ECS programs can be started.

New Features in Release 2.4

The following features have been added since the 2.21 release. The new features are described briefly here and are grouped according to the following classifications:

ECS Shell	New changes to the ECS Shell and security
Hierarchy Navigator	New changes to the Hierarchy Navigator only
Schematic Editor	New changes to the Schematic Editor only
Symbol Editor	New changes to the Symbol Editor only
Waveform Tool	New changes to the Waveform Tool only
ecs.ini Editor	New changes to the ecs.ini Editor only
Interfaces	New changes to the various netlisters and other interfaces

ECS Shell

- The ecs.cfg file on the PC is now an ASCII file. This allows problems with the license file to be fixed using a fax rather than having to send a file on a floppy disk.
- A new licensing mechanism is used on UNIX workstations allowing more flexible use for MOTIF users. It requires that a daemon be started.
- Running the program ecsshell in Sunview and Motif automatically starts as a background job. You no longer need the following ampersand, "&", to run ecsshell in the background.
- Command line options have been added on UNIX workstations allowing the ecsshell to be started and the Symbol Editor, Schematic Editor or Hierarchy Navigator spawned immediately with a particular design. The syntax is shown below:

```
ecsshell -nav design_name
ecsshell -sch design_name
ecsshell -sym design_name
```

Hierarchy Navigator

- Global constants can now be edited in the Navigator and passed to the netlisters. Use the **Misc->Edit Constants** command in the Navigator to change values of existing global constants. The changes to the global constants are temporary and only remain valid until you quit from the Navigator. The changed values are not stored anywhere.
- The simulator control menu has been ported from the Sun to the PC. The simulator parameter in the control section of the ecs.ini file causes ECS to look for a file called *simulator.ini*. ECS on the PC looks first in the local directory for

this file and if none exists, it then looks in the data directory. This file configures the simulation menu and the operation of the simulator and Waveform Tool.

- The Navigator can be rebuilt on all platforms from inside the Navigator using the **File->Restart** command. This allows you to modify a symbol or schematic inside a design while the Navigator is open and then rebuild the design and see the changes reflected in the Navigator in real time. The rebuilding is automatic on the Sun and PC.
- The **Misc->Query** command now supports iterated instances. You can query individual instances of an iterated instance and also pins of iterated instances.
- The **Misc->Query** command now supports reference designators. You can search for a symbol having a particular reference designator by typing in the reference designator at the query prompt.
- The **Misc->Attributes** command in the Navigator now allows you to edit attributes on individual elements of a bus.
- A new command called **Tools->View Report** has been added to the Design Analysis Tools package. This command displays a dialog box allowing you to choose a file. The contents of this file are displayed in a list box. Error messages from netlisters and other PIK programs now use this as the standard interface to display error messages in the Navigator. Third party programs can also make use of this interface.

A special syntax allows the the **Tools->View Report** command to jump to and highlight specified nets, instances, pins or symbol types in a design. If you click on a line in the list box, the line is parsed looking for a keyword followed by a valid identifier. The keyword and identifier can be embedded inside a comment. The keyword and identifier are separated by: any number of spaces (including 0) followed by an equal sign (=) followed by any number of spaces (including 0). You can use uppercase, lowercase or mixed case for the keyword and identifier. The keywords are:

I or Inst or Instance	represents an instance, identifier is instance
N or Net	represents a net, identifier is net name
P or Pin	represents a pin, identifier is pin name
T or Type or Symbol	represents a symbol type, identifier is type

Examples of lines in the list box are:

I=adf.pdiff Bad spice attribute on this line

Unconnected pin = adf.pdiff-input1

The following symbol=NAND2 has a connectivity problem

Clicking on the first example line causes the Navigator to jump to schematic *adf* which contains instance *pdiff*. Instance *pdiff* is highlighted. The text of the error

message is ignored. This free format allows the keywords in the error message to be relatively unobtrusive.

- The above command, **Tools->View Report**, has been extended to allow display of critical paths. If the format of the file is as shown in the example below:

```
[ path 1]
first net=.input
first inst=.adder.nand3
second net=.adder.nand3.N_23
...
[path 2]
first net=.cin
first inst=.adder.mux
second net=.adder.mux.control
...
```

where header information is between square brackets and items in each sublist below the corresponding header are treated as described above for the **Tools->View Report**. When the command is first invoked, a list box is displayed containing the lines from each header. Clicking on a header line in this list box displays a second list box containing the individual entries for the selected header.

Clicking on an entry in the second list box causes the Hierarchy Navigator to jump to the instance or net contained in the selected line. The system of keywords and identifiers as described above is valid here also.

Schematic Editor

- New commands have been added to support data tables in a schematic. You can access data stored in the table using derived attributes. The command names are **Add->Table** and **Add->Table Data**.
- A menu driven method of specifying net attribute values for the simulation interface has been added. The particular net attribute to be added is specified in the *simulator.ini* file as *NetAttribute* under the *[ModelBuilder]* section. The net attribute is referenced by attribute number.

The values that this net attribute can assume are also specified in the *simulator.ini* file. A new section is defined in the *simulator.ini* file called *[net attribute]*. This has entries of:

```
button1_label = value1
button2_label = value2
...
```

where *button1_label* is the label you want displayed on a dialog box and *value1* gets sent to the simulator when the button labelled *button1_value* is pressed.

- Net attributes are logged so nets can be moved and retain the net attributes. This allows you to move wires and maintain the verilog wire types.
- Pins can now use derived attributes. Nets can be referenced from within a derived attribute. The new additions to the derived attribute syntax are described below.

If the instance name is terminated by a minus sign, then a pin name must follow. Pin names are terminated by a space, equal sign (=) number sign (#) or dollar sign (\$). If the pin name is empty it refers to the current pin (@-) or (@.-). If the pin name is followed by an equal sign (@=) or (-pinname=), the net connected to the pin is used instead of the pin.

- If you try to open a schematic and the symbol search paths are not set correctly, the schematic editor may not be able to locate all of the symbols. The Schematic Editor now opens the schematic so you can see blanks on the schematic where the missing symbols should be placed. This allows you to see which symbols are missing. The schematic opens with the name *untitled* so if you accidentally save, you do not destroy the references to the original symbols which are missing.
- The **Misc->Query** command in the Schematic Editor or Navigator displays the full path name of symbols or schematics not located in the current directory.
- You can add tools to the **Tools** menu of the Schematic Editor. Command line options are indicated using minus signs. This allows UNIX type commands to be added to the or **Tools** menu. You add tools menu entries from the **Tools->Schematic Tools** section of the ecs.ini editor (Preferences Editor on Macintosh).
- Underscores are now legal characters inside instance names except the first and last characters cannot be underscores.
- **Misc->Query** now supports buses.
- A table function has been added to the schematic editor. This functionality is accessed using the **Add->Table** command. Various elements in the schematic can reference values in these tables by using derived attributes.
- You can override the location of an attribute window on an instance-specific basis. This is useful for making a schematic look neater. Use the **Add->Show Attribute** command to do this.
- Schematic sheets can hold 16 times as much data as in the 2.21 release.
- You can now edit the names of nets using the the **Add->Net Name** command. Use the following procedure to do this:
 1. Select the **Add->Net Name** command.
 2. Hold down the shift key while clicking on the net whose name you like to edit. The name of the selected net is placed on the prompt line.

3. You can now edit the name using arrow and delete keys. Type *Return* when your edits are complete.
 4. The edited name can be placed on the schematic in the usual ways.
- Entering a bus name for a net has been improved. You no longer need to enter an equal sign preceding or following the bus name. To get individual elements of a bus stripped off and attached to the cursor for naming bus taps, use the new **Add->Expand Bus** command. This command is only active when you use the **Add->Net Name** command and you have a bus name attached to the cursor. The command is activated through the menu, by typing the escape key, or pressing the middle mouse button on 3 button mouse systems, or by pressing the right button on 2 button mice.
 - File lockout has been extended to include operation between PCs, UNIX workstations and Macintosh computers connected by a network. This means that if a person is editing a schematic or symbol file on a UNIX platform, and a second person tries to access the same schematic or symbol file over the network from another UNIX platform, a PC, or a Macintosh, the second person is told the file is in use and asked if anyone on the network is using the file. The same protection is provided if a PC or a Macintosh open a file over the network and another machine tries to access the same schematic or symbol file. File lockout is integrated with crash recovery as described in the next point.
 - Crash recovery is now automatic on the Sun. Previously, you had to rename a file ending in *_SC* to the same file ending in *_sc*. This process has been automated so you are guided through the crash recovery procedure. An additional benefit is that possible conflicts with others using the network are avoided.

Crash recovery works in the following way. A log file is created when the Schematic Editor is first started. The name of the log file is *schematic_name._sc*. The schematic editor checks for the existence of this file before opening an existing schematic. If this file exists, it means

- either someone else on the network has opened this particular schematic, or,
- the last time the schematic was open, the Schematic Editor did not exit gracefully (i.e., it crashed).

If someone else on the network is editing the file, you should not edit it or you will overwrite each other's changes. In order to prevent this from occurring, ECS creates a warning message that asks you to verify that no one else on the network is editing the target schematic file.

If you answer that no one else on the network is using the file, then it assumes that the log file was left from the Schematic Editor crashing. The Schematic Editor prompts you asking if you want to recover the file. Responding with a yes causes the Schematic Editor to automatically recover the file.

A side effect of this is that automatic crash recovery is not possible on new schematics that have not yet been saved. It is recommended that you perform a save shortly after starting a new schematic.

- The **Misc->Error Check** command now saves a list of errors to the file *schematic_name.err*.
- A PCB packager has been added to the Design Analysis Tools module. The packager can automatically assign reference designators and pin numbers to symbol instances in the schematic.

Symbol Editor

- You can add tools to the **Tools** menu of the Symbol Editor. Command line options are indicated using minus signs. This allows UNIX type commands to be added to the Symbol Editor or **Tools** menu. You add tools menu entries from the **Tools->Schematic Tools** section of the ecs.ini editor (Preferences Editor on Macintosh).
- Pin name offsets can be increased to 31 fine grid units. This has been increased from the previous limit of 15 fine grid units.
- Bus pins can be created on *component* type symbols. When a bus pin is created on a component type symbol, the numbers of the physical pins must be specified as part of the symbol definition.

These numbers are a list of pins assigned to the pin attributes, *BusPin1* through *BusPin8*. These pin attributes are numbered 90 through 97 in the ecs.ini file. The normal *PinNumber* pin attribute must be left unused. The list can be spread sequentially through the 8 attributes. Each individual attribute can hold about 200 characters. The list can be comma or blank delimited and can contain sequences of pins by using parentheses ()'s or square brackets []'s to delineate the numeric range.

Examples are:

BusPin1 = 1,3,5,(7:10)	represents 7 pins - 1, 3, 5, 7, 8, 9, 10
BusPin2 = A1 B[2:4] C1	represents 5 pins - A1, B2, B3, B4, C1

- Text justification is now indicated by a horizontal bar above and below the justification point. Figure 1 shows left and center justification.

normal left
Center

Figure 1 Left and Center Justification

- You can change the *type* of a symbol in the Symbol Editor by using the **Misc->Change Symbol Type** command.
- Title blocks can make use of standard symbol attributes allowing you to create your own attributes such as *designer* or *project* for use on title symbols. Symbol attributes numbered from 100 to 199 can be used for this purpose.
- Default reference designators can now be placed on symbols by using the **Add->Symbol Attributes**. This is described in the packaging section of this release note.
- The **Misc->Error Check** command now saves a list of errors to the file *symbol_name.err*.

Waveform Tool

- Printing on the PC has been rewritten to include start time, end time, scale, etc.
- The zoom factors on the PC now allow the full data range to be viewed.
- The Waveform Tool on the PC has been renamed to *WAVES*. It has the following command line options:
 - nav connects to navigator
 - sim specifies that program is running dynamically with simulator
 - name* the parameter *name* specifies the root name for the display save file (.wav) and the timewave history (.his) if specified
 - time reads waveform data from timewave history file
 - haz opens a hazard file having name *design.haz* where *design* is the name of your current design.
- A new command, **Misc->View Report**, has been added to display hazard information and/or any other information. Selecting this command pops up a dialog box allowing you to choose the file name to open. The default file name is *design_name.haz*.

After opening a hazard file you can select a particular hazard from a list box and the Waveform Tool jumps to display the hazard condition. The Navigator, if connected, jumps to highlight the problem node. The Waveform Tool must have the command line option *-haz* specified in order for the command to be added to the Waveform Tool menu.

The format for hazards in the hazard file is:

```
hazard_time1 net_name1    Comment about hazard1
hazard_time2 net_name2    Comment about hazard2
```

where *hazard_time* is a time in integer format and *net_name* is an ECS hierarchical net name and the comment field is any text information. The three fields are separated by one or more spaces.

Hazard conditions following the current query cursor time are displayed in the list box by the **Misc->View Report** command. Hazard conditions prior to the the current query cursor time are pruned from the list box display.

- The Waveform Tool can now print and view up to 256 waveforms.
- Waveforms can be added to the display by searching through a hierarchical list box containing all the monitored nets. This is activated by clicking on the alternate mouse button on UNIX workstations or the PC.
- Waveforms can now be stored in a binary format by using the **File->Save** command and then viewed at a later time. This allows you to save many simulations from a single design and then compare them at a later time.

ecs.ini Editor

- You can store default configurations in the ecs.ini file for several graphics options. The following options can be manually added to the [controls] section of the ecs.ini file.

DefaultTextSize	sets the default text size. choices are 0, 1, 2
DefaultTextJust	sets the default text justification
DefaultTextRot	sets the default text rotation
FullCursor	defines whether full cursor is the default
DefaultLineStyle	sets the default line type (only available on UNIX workstations)
DefaultGrid	sets the default grid spacing
ShowGrid	if set =yes, the grid is displayed by default on schematics
NetNameCaps	if set =yes, net names are forced to uppercase. This is default if this parameter is not specified. If set = no, net names retain the case you typed them in with.
AttributeCaps	if set =yes, attribute values are coerced to be uppercase. If set =no, attribute values retain the case you type them in with.
Editor	sets the text editor that ECS calls for displaying textual information. Examples are notepad.exe, emacs.exe.

- The printer setup on the PC can now be changed from within ECS rather than having to go to the Windows control panel. The command is **File->Printer Setup**.
- Multiple master symbols can now be specified in the ecs.ini Editor. They should be separated with spaces in the entry field in the **Controls->Sheet Values** section of the ecs.ini Editor (Preferences Editor on Macintosh).

Interfaces

- ECS now supports an archive utility that gathers all of the schematics and symbols required in a design and places them all in a single directory. It is run as a tool from the **Tools** menu of the Hierarchy Navigator. There are currently two options for the command as described below:

archive creates a list of all files needed for the current design. This list is written to file *design_name.lst*.

archive -save=path saves all symbol and schematic files necessary to build the design. These files are all saved in the directory referenced by *path*.

The disk drive name and any folders referenced on the *path* must not contain spaces on the Macintosh.

The archive feature does not currently save graphic symbols or master symbols. Graphic and master symbols do not affect the connectivity of a design so archive is sufficient for electrical purposes.

- The VHDL netlister can now write ports of type *buffer*. To make a buffer port, the I/O marker on the schematic must be a BIDIR and the matching pin on the symbol representing the schematic must also be bidirectional. The symbol pin must also have attribute number 35, *VHDL_PinUse* set to *buffer*. With this combination of attributes, the port netlists as a BUFFER.
- The following netlisters have been consolidated into a single executable file called *lister*:

<i>netorder</i>	net list by net
<i>pinorder</i>	net list by instance and pin
<i>listmark</i>	list only marked nets and instances
<i>listinst</i>	list type, location, parent and instance names
<i>listpart</i>	list block symbols, count of primitive symbols

Each of these netlist options is created by using one of the following command line options:

-netorder
-pinorder
-listmark
-listinst
-listpart

- The following options are also available on all the netlisters:
-nohead suppress printing of header with time and date

- pcb* use reference designator and pin number instead of instance name and pin name
- view* run the currently specified editor to view the netlist file immediately after netlist creation
- ext=.abc* create a netlist file with the specified extension, i.e., *root_design.abc*

- The edifin program has improved text handling capabilities. When translating EDIF symbol libraries, the text positioning is closer to that on the original symbols.
- The schematic PIK can now handle derived attributes.
- The Interprocess communication on the PC is more robust. This improves the communication between the Waveform Tool and the Navigator and helps the simulation interface.
- The EDIF netlister now supports instance, pin and net attributes.
- The EDIF netlister has the following new options:
 - /N use notepad to view netlist immediately after it is written
 - /E use external statement instead of library statement on primitives
 - /S suppress whitespace in netlist, makes netlist smaller but less readable
- The verilog netlister has been broken out of the Navigator and is now much easier to support. The new executable is called vericode.
- The verilog netlister now has an option to suppress the implicit pin ordering for verilog primitives. For example, the verilog primitive XOR was previously coded using implicit pin ordering rather than explicitly referencing each pin to the corresponding net. The following option

vericode -noprimitives

 forces explicit pin connections to be made on verilog primitives.
- The verilog netlister now gathers individual indexed pins from a symbol and groups them to form a bus. For example, a symbol having individual pins DATA[0], DATA[1], and DATA[2] are grouped together in the verilog netlist to form DATA[0:2]. This allows a clean interface to symbols created by logic synthesis tools.
- The netlisters now print out the netlist creation date in the netlist, as well as a version number for the specific netlister.
- The *Spiceline2* attribute is now automatically appended to the end of subcircuit headers. Previously it was added to the SPICE netlist on the line following the subcircuit header, forcing you to add a plus sign at the beginning of the line.

- The *Spiceline* and *Spiceline2* attributes now break automatically if they are longer than 80 characters.
- The error log from CheckCkt and CheckPCB are now printed to a file having the same root name as the current design and an extension of *.err*. You can use the **Tools->View Report** command of the Hierarchy Navigator to view this file and dynamically locate all the errors specified in the file.

Changes to PCB Netlisters

- The PCB netlisters have been consolidated into one program called *pcbnet*. The different netlists (PADS, RINF, Cadnetix) are selected using the following command line options to the *pcbnet* executable.

<i>-pads</i>	write PADS netlist file
<i>-rinf</i>	write Recal Redac RINF format netlist file
<i>-cadnetix</i>	write Cadnetix netlist file

- An option has been added enabling the use of hidden power pins in the PCB netlisters. It is often desirable to hide the power pins in a schematic drawing. Symbol attributes 60 to 69 are used to represent hidden power connections to symbols.

The following command line option enables the use of hidden power pins in the PCB netlisters:

<i>-power</i>	use the hidden power pin option for PCB netlisting
---------------	--

Examples of their use in the processes section of the *ecs.ini* file are

PADS Netlist = *pcbnet -pads -power*

- The PADS netlister supports the partshape attribute and hidden POWER and GND pins.
- The net attribute *width* has been added to the PADS netlister to represent trace widths in PCB applications. This is netlisted with the proper syntax in the PADS netlist.
- The bill of materials program has been improved.
- A PCB packager has been added to the Design Analysis Tools module. The packager can automatically assign reference designators and pin numbers to symbol instances in the schematic.

Back Annotation Programs

- A new command, **File->Back Annotate**, for performing back annotation has been added to the Hierarchy Navigator. The *.atr* file is no longer read in

automatically when the navigator loads the design. You must use the **File->Back Annotate** command to load any *.atr* files.

- The back annotation programs *padsback* and *rinfbac* have been consolidated into a single executable program called *pcbback*. It has the following command line options to distinguish between PADS and RINF formats.

```
-pads      back annotate from PADS file with name  
           root_design_name.eco  
-rinf      back annotate from Recal Redac file named  
           root_design_name.irp
```

Notes For MOTIF Users

ECS uses its own set of fonts which are contained in the ECS_ROOT tree. The raw fonts are in the xfont directory. A directory also exists which contains the compiled fonts for various manufacturers, such as xfontDEC and xfontIBM. These directory names are automatically generated from the X ServerVendor string. The first 3 capitals in the string are appended to the fixed 'xfont' string. On Sun platforms where there is a choice of X server, the system shipped by ECS is the MIT server. It thus uses the directory named xfontMIT. If you run OpenLook on your server, the fonts will be obtained from the directory xfontXNW. If you do not have any machine of a particular manufacturer in your network, the corresponding directory may be deleted. If you have a custom server on your machine, you should be able to create another directory with the name built up as above and install the fonts using the bdf2snf and mkfontdir X processes, or the vendors equivalent procedures. If there is no change in font formats, link the directory name to an existing directory.

Further Explanation of Master Symbols

Master symbols are used on schematics for title blocks and other tables that are required to appear on every design. They are automatically placed at the time a new schematic is created. There is a parameter in the **Controls** section of the ecs.ini file that sets which master symbols are to be placed.

The origin of a master symbol snaps to one corner of the schematic drawing. The corner is determined by the relationship between the origin of the symbol and the symbol data. If the origin is to the lower right of the symbol data, the master symbol is placed in the lower right hand corner of the schematic. If the origin of the symbol is to the upper left of the symbol data, the master symbol is placed in the upper left corner of the schematic. Similarly for the other two corners.

The origin must be located at least 1 grid unit outside of a rectangular area enclosing the symbol data. For example, if you want the master symbol to appear in the upper right hand corner of the schematic, you must set the origin 1 grid unit above and 1 grid unit to the right of any of the symbol's data.

Bug Fixes in Release 2.4

- The surfacing of sheets and views now works correctly when using the Query command in the Navigator.
- If the complete contents of a sheet is deleted, it no longer crashes the system.
- The Create Symbol utility on the Macintosh is more robust and allows mistyped characters.
- Crash recovery no longer crashes if you had changed the sheet size before the original crash.
- The *D* from demorgan equivalent symbols no longer appears in PCB netlists.
- The verilog attribute *veristrength* (attribute #22) now comes before *veritimes* (attribute #21) in the verilog netlist.
- PCB netlisters now netlist gate symbols properly. The PCB netlisters used to add extra pins to the first gate section.
- Verilog wires are now supported by Schematic to ASCII.
- Undo now works on verilog wires.
- The waves program now exits gracefully when you quit a verilog simulation.
- The ECS now allows you to use up to 200 attributes on one symbol. The previous limit was 128 attributes.
- Pin attributes now work properly on all sheets of a design. The PC version sometimes failed when you added pin attributes on multiple sheet schematics.
- The ASCII in and ASCII out programs no longer lose the text rotation.
- You can now have a text editor defined on the Macintosh desktop level.
- Attribute windows are now displayed correctly on all sheets.
- The Zoom cursor now works on a black background
- If you use the **Edit->Drag** command in the Schematic Editor to move large sections of your drawing, it now limits the number of connections to 100. The command now ignores any wires after the first 100.

Reporting New Bugs

Please use the form below to report any new bugs that you find. It is important to be as specific as possible when describing a bug. Try to determine the minimum number of commands necessary to reproduce the bug. When describing a message on the screen please spell out the message **EXACTLY** as it appears on the screen.

If any special data is required to reproduce the problem, try to create the smallest test case necessary exhibiting the problem. Use the archive feature to gather all necessary symbols and schematics into a single directory. Include your ecs.ini file and include the data and ecs.ini file on a floppy disk and mail to:

Customer Support Group
CAD/CAM Group
20480-B Pacifica Drive
Cupertino, CA 95014
telephone 408 725-0204
fax 408 725-0207
email uunet!ecs!support

After we receive the written bug report, we will assign a tracking number to the bug and return your bug report. You can inquire about the bug by referring to the bug tracking number.

ECS Bug Report

CAD/CAM Group fax # 408 725-0207
email uunet!ecs!support

Reported by:

ECS bug tracking #:

Company:

Phone:

Date:

ECS Software Version:

ECS Program name:

Hardware Platform:

Software Configuration (Motif, SunOS, etc):

Is bug reproducible?

Is bug only present with specific data?

If so, did you include data?

Importance of bug to you. (10 means your work is stopped, 1 is minor)?

Reference to proper operation in manual.

Description of bug:

ECS Enhancement Request

CAD/CAM Group fax # 408 725-0207

email uunet!ecs!support

Reported by:

ECS enhancement tracking #:

Company:

Phone:

Date:

Importance of enhancement to you. (10 means your work is stopped, 1 is nice to have if time)

Purpose of enhancement (What problem are you trying to solve)

Description of proposed enhancement: