

ChipVault
2002.11

User Definable Flag Options:

Perl Source Line Number: 734

```
[user_vars_start]
#####
User INI File Settings:
    $user_ini_enable = 1; # Turn Off to speed up start/quit

    $user_ini_file = ".USER_NAME_cv.ini"; # Hidden User Config
    $user_ini_file = "USER_NAME_cv.ini"; # Visible User Config

    ( $user_ini_file =~ s/USER_NAME/$my_name/go ); # Search+Replace Tag
    $notify_on_hlist_change = 1; # Tell user when hlist.txt changes
#####
Advanced Feature Enables:
$disp_tool_opts = 0; # Display the Tool Options in the Tool Menu
$disp_tool_bindings = 1; # Display the Key bindings in the Tool Menu
$disp_file_in_main = 1; # Display the Select Filename in Main Window
$disp_file_xterm_title = 1; # Display the Select Filename in Xterm Title
$enable_split_display = 1; # Split Display in 2 for simultan port views
$split_display_size = 1; # Default Split Display Sizing
$split_display_mode = 0; # Default Split Display Mode (Port vs Log)
$enable_archives = 1; # Make 0 if you don't want TARs made
$enable_history = 1; # Make 0 if you don't want History Logs
$enable_vt100_color = 0; # Make 0 if you don't want UNIX VT100 Color
$enable_web_access = 1; # 0 to turn off Web Access Version Checking
$vers_check_after = 50; # How many invokes before web-vers-check
$proxy_en = 0; # 1 if you need a Proxy Server for Web Access
$proxy_authentication = 0; # 1 if username and password reqd
$proxy_ip = "proxy103a"; # IP name for your Proxy Server
$proxy_port = "8080"; # Port (ie 80 or 8080)
$encryption_type = "cheezy"; # cheezy or blowfish
$default_crypt_key = "128bitOpenAccess"; # Default Encryption Key
$encryption_suffix = "cv_world"; # ie foo.v => foo.v.cv_world
$newbie = 0; # Turns on annoying hints and stuff
$enable_confirm_checkin = 1; # Make 0 if you don't confirmation prompting
$enforce_checkouts = 1; # Prohibits running editor on mainline files
$enforce_checkouts_truly = 0; # Uses CVs RTL Viewer instead of vi -R
$enforce_warn_time = 2; # How many seconds to display warning dialog
$dynamic_resize_checking = 0; # Allow for dynamic window resizing
# Turn on for really fast systems
# Then you don't need to press <r>esize key.
$stty_pipe_doesnt_work = 0; # Set 1 to not call stty for screen size
$default_noise_filter = ""; # Null is default as it forces Prompt Popup
$license_banner = 1; # Set to 1 to make popup last 1 second
$license_banner_quit = 1; # Set to 1 to make popup last 1 second
$draw_main2wip_line_en = 1; # ASCII Line connecting Mainline file to WIP
$win32_mouse_enable = 1; # Enables Mouse Support on win32
$vim_lines = 80; # Defines VIM Screen Line Size under Win32
$disp_select = 0; # Set 0 for module, 1 for file, 2 for refdes
$gates_per_cell_area = 40; # For report_analysys, this tries to
# correlate Synopsys Cell Area with Gate cnt
# This will change for every tech library.
$enable_clock = 1; # Display the clock on screen or not
$gzip_branched_files = "?"; # Gzip the branched files or leave in clear
$unix_tar_en = 1; # Use UNIX tar instead of internal TAR
$unix_gzip_en = 1; # Use UNIX gzip to compress TAR files

[user_vars_stop]

$style = 0;
@style_flag_expanded [$style] = " - "; # Classic
@style_flag_collapsed[$style] = " + ";
@style_flag_nokids [$style] = " ";
$style++;
@style_flag_expanded [$style] = "[-] "; # KMH Favorite
@style_flag_collapsed[$style] = "[+] ";
@style_flag_nokids [$style] = " ";
```

```

$style++;
@style_flag_expanded [$style] = "[-] "; # Minor Variation
@style_flag_collapsed[$style] = "[+] ";
@style_flag_nokids   [$style] = "\ \ ";
$style++;
@style_flag_expanded [$style] = "\(-\) "; #
@style_flag_collapsed[$style] = "\(+\) ";
@style_flag_nokids   [$style] = " ";
$style++;
@style_flag_expanded [$style] = "<-> "; #
@style_flag_collapsed[$style] = "<+> ";
@style_flag_nokids   [$style] = " ";
$style++;
@style_array[$style] = "end";

$style = 1; # Specify the Default Style
update_style($style);

Win32 users under Active-Perl can pick some different stock color schemes
@win32_color_array[0] = ($FG_BLACK | $BG_WHITE );
@win32_color_array[1] = ($FG_WHITE | $BG_BLACK );
@win32_color_array[2] = ($FG_WHITE | $BG_BLUE );
@win32_color_array[3] = ($FG_GRAY | $BG_BLUE );
@win32_color_array[4] = ($FG_GRAY | $BG_BLACK );
@win32_color_array[5] = ($FG_GREEN | $BG_BLACK );
$win32_color_select = 0;
$win32_color_scheme = @win32_color_array[$win32_color_select];

VT100 users can pick some different stock color schemes
@vt100_color_array[0] = " ";
@vt100_color_array[1] = "black white";
@vt100_color_array[2] = "white black";
@vt100_color_array[3] = "white blue";
@vt100_color_array[4] = "green black";
@vt100_color_array[5] = "blue black";
$win32_color_select = 0;
$vt100_color_scheme = @vt100_color_array[$win32_color_select];

if ( $os ne "unix" )
{
    $unix_tar_en = 0; # Use UNIX tar instead of internal TAR
    $unix_gzip_en = 0; # Use UNIX gzip to compress TAR files
}

Ignore this stuff. Just author specific prefs
if ( $my_name eq "khubbard" )
{
    $disp_file_in_main = 1; # Display the Select Filename in Main Window
    $disp_file_xterm_title = 1; # Display the Select Filename in Xterm Title
    $license_banner = 1; # Set to 1 to make popup last 1 seconds only
    $license_banner_quit = 1; # Set to 1 to make popup last 1 seconds only
    $proxy_en = 0; # 0 if no Web Proxy Server
    $proxy_authentication = 0; # 1 if username and password reqd
    $proxy_ip = "proxyl03a"; # IP name for your Proxy Server
    $proxy_port = "8080"; # Port (ie 80 or 8080)
}

[user_vars_start]
# Set this to NULL if you want no web access for checking versions.
$web_page_url = "http://chipvault.sourceforge.net";
$change_list_url = $web_page_url . "/change_list.txt";

$cv_include_var = "CV_INCLUDE"; # Environment variable for libs
$cv_include_file = ".cv_include"; # Home Hidden file for libs

$issue_file = "issues.txt"; # Issue Tracking DataBase
$print_file = "print.txt"; # Text Printout of Blocks
$pdf_file = "print.pdf"; # PDF Printout of Blocks
$branches_subdir = "branches"; # Subdir to put branches in

```

```

$log_file_tag      = "USER_NAME_log.txt"; # Standard Output Log File
$log_file         = $log_file_tag;
( $log_file      =~ s/USER_NAME/$my_name/go ); # Search+Replace Tag
$slog_file_tag    = "USER_NAME_slog.txt"; # Standard Output Small Log File
$slog_file       = $slog_file_tag;
$strace_file      = "cv_trace.txt";      # Software Debug Trace File
( $slog_file     =~ s/USER_NAME/$my_name/go ); # Search+Replace Tag

$sys_file_tag    = "USER_NAME_syslog.sh"; # Batch Log File
$sys_file       = $sys_file_tag ;
( $sys_file     =~ s/USER_NAME/$my_name/go ); # Search+Replace Tag

$rel_path_to_src = "../src/";           # Relative Path from Xilinx to Src

#####
Admin Setup : Admin users can change file Write Permissions from ChipVault.
$admin_hash {"khubbard"} = 1;
$admin_hash {"hunt"}     = 1;
$company_name = "Acme ASIC Design";

#####
Email Setup : This allows for automatic Email notification to entire
              group on file Checkins.
$email_domain   = ""; # ie "\@foo.com" Can be kept null oftentimes.
@email_list     = ("khubbard","khubbard"); # This is for checkins.
$send_email_on_check_in = 0; # Annoying. Some people like this.
$send_email_allowed = 1; # Security feature Email enabling.

if ( $os eq "unix" ) # Configure Browser for Webbing to ChipVault WebSite
{ $browser          = "netscape"; }
else
{ $browser          = "iexplore.exe"; }

#####
Shared Library Component Modules : These are like hlist.txt files but they
                                  contain common building block components
                                  for designers to browse and use in their
                                  designs.
@lib_list        = ("/home/asic/group/examples/vhdl/library.txt",
                  "/home/asic/group/examples/verilog/library.txt" );

#####
File Name Formats:
$wip_header      = "wip."; # Defines Name for CheckedOut Files.
                  # ie "wip.foo.vhd"
$star_header     = ".archive."; # I like this hidden.
$star_list_header = ".archive_list."; # Remove 1st dot to make these visible
$star_list_footer = ".txt"; #

$hist_log_header = ".history_log."; # I like this hidden
$hist_log_footer = ".txt"; #

#####
Misc Options:
$bell           = chr(7); # Warning Bell. I find them annoying. Set to "" instead.
$x_scale        = 2; # Defines the number of spaces between hierarchy levels
                  # for shallow hierarchy designs, you might want this to
                  # be 4 to give more spacing. For really deep designs,
                  # you'll want this to be 1 so all levels fit on the screen
$activate_tool_once_selected = 1 ; # Once you pick a new tool, use it now

$horiz_scroll_bar_option = 1 ; # Enables drawing horizontal scroll bar.
[user_vars_stop]

#####

```

Perl Source Line Number: 946

KeyCodes: These work for my machine, which means 1% chance they'll work on
yours. Envoke cv.pl DEBUG to see your key codes.

```
$k_ctrl_d = 4 ;  
$k_ctrl_u = 21 ;
```

```
if ( $os eq "unix" )
```

```
{  
  $k_up   = 65 + 256; # Note: 256 is my offset out of ASCII range for 3-char key  
  $k_dn   = 66 + 256;  
  $k_lf   = 68 + 256;  
  $k_rt   = 67 + 256;  
  $k_cr   = 10;  
  $k_tab  = 9;  
  $k_esc  = 27;  
}
```

```
else
```

```
{  
  $k_cr = 13;  
  $k_up = 72; # Win32 ScanCodes  
  $k_dn = 80;  
  $k_lf = 75;  
  $k_rt = 77;  
  $k_tab = 9;  
  $k_esc = 27;  
}
```

```
$k_sp = 32;  
$k_p  = 112;  
$k_q  = 113;  
$k_a  = ord("a");  
$k_e  = ord("e");  
$k_h  = ord("h");  
$k_z  = ord("z");  
$k_1  = ord("1");  
$k_2  = ord("2");  
$k_3  = ord("3");  
$k_4  = ord("4");  
$k_5  = ord("5");  
$k_6  = ord("6");  
$k_7  = ord("7");  
$k_8  = ord("8");  
$k_9  = ord("9");
```

```
$k_mouse_double_click_togl = -1;  
$k_mouse_double_click_edit = -2;  
$k_mouse_port_view         = -3;
```

Now Load the User Preferences

```
  read_user_ini(); # Yuck. Call this twice, 1st time for Editor Selection  
                  # but the HLIST hasn't been read in yet, so checksum fails
```

User Definable Editor and Tool Assignment:

Perl Source Line Number: 1003

#####

\$j=1;

Note : enforce_checkouts will only check for tool#2 being used against a non WIP file, so the 1st tool assigned after this "Edit Tools" label will be the only one checked.

-----Display----- # -- UNIX Command --

#####

```
@my_tool[$j++] = "- Edit Tools ----- #";
@my_tool[$j++] = "- Editors ----- #";
```

Assign your editor and Tools Here

```
if (
  ( ( $my_editor_unix_bin eq "" ) && ( $os eq "unix" ) ) ||
  ( ( $my_editor_win32_bin eq "" ) && ( $os eq "win32" ) )
)
{
  # $def_editor = "emacs";
  if ( $os eq "unix" ) { $def_editor = "vi"; }
  else { $def_editor = "c:\\vim\\gvim.exe -c \"set lines=$vim_lines\""; }

  print STDOUT "Please select your default editor\n";
  print STDOUT "(ie 'vi<enter>' or 'emacs<enter>'\n";
  print STDOUT "NULL<enter> will select $def_editor.\n";
  $rts = my_get_line();
  chomp $rts;

  if ( $os eq "unix" )
  {
    if ( $rts eq "" ) { $my_editor_unix_bin = $def_editor; }
    else { $my_editor_unix_bin = $rts ; }
  }
  else
  {
    if ( $rts eq "" ) { $my_editor_win32_bin = $def_editor; }
    else { $my_editor_win32_bin = $rts ; }
  }

  print STDOUT "Thank You. Your default editor is now ";
  print STDOUT "$my_editor_bin and will be saved in your";
  print STDOUT "$user_ini_file file.";
}# if ( $my_editor eq "" )

if ( $os eq "unix" )
{
  if ( $os_title eq "linux" )
  {
    # Assume VIM
    # $my_editor = " " . $my_editor_unix_bin . " -c \"set noeb\" -c \"set vb t_vb=\" ";
    $my_editor = " " . $my_editor_unix_bin . " "; # Assume Vanilla Vi or Emacs
    @my_tool[$j++] = " Edit <e> # $my_editor +LINE_NUM FILE_NAME";
  }
  else
  {
    $my_editor = " " . $my_editor_unix_bin . " "; # Assume Vanilla Vi or Emacs
    @my_tool[$j++] = " Edit <e> # $my_editor +LINE_NUM FILE_NAME";
  }
}

if ( $my_editor_unix_bin eq "vi" )
{
  @my_tool[$j++] = " Edit ReadOnly # $my_editor +LINE_NUM -R FILE_NAME";
  $read_only_viewer = $j-1;
}
else
{
```

```

    @my_tool[$j++] = " Edit ReadOnly          # less +LINE_NUM FILE_NAME";
    $read_only_viewer = $j-1;
}

$macro_hash { ord( "e" ) } = $j-1; # Assign a Macro Key to Tool
}
else
{
    $my_editor = " " . $my_editor_win32_bin . " ";
    @my_tool[$j++] = " Edit                  <e> # $my_editor FILE_NAME";
    @my_tool[$j++] = " Edit ReadOnly        # $my_editor FILE_NAME";
    $read_only_viewer = $j-1;
}

$macro=0;
@tool_macro[$macro++] = $j-1 ;          # PreLoad Macro with Editor

@my_tool[$j++]=" cv_viewer              (int)    <v>    # cv_viewer ";
@tool_macro[$macro++] = $j-1 ;          # PreLoad Macro
@my_tool[$j++]=" cv_rtl_viewer          (int)    # cv_rtl_viewer ";
@tool_macro[$macro++] = $j-1 ;          # PreLoad Macro
@my_tool[$j++]=" Edit_fork_xterm+vi     # xterm -T ChipVault-FILE_NAME -e vi FILE_
@my_tool[$j++]=" Edit_Vi-Text           # vi                FILE_NAME";
@my_tool[$j++]=" Edit_Emacs-Text        # emacs -nw         FILE_NAME";
@my_tool[$j++]=" Edit_Emacs-GUI         # emacs              FILE_NAME";
@my_tool[$j++]=" Edit_DOS_Vim           # c:\\vim\\gvim.exe FILE_NAME";
@my_tool[$j++]=" Edit_log_file           # $my_editor $log_file ";
@tool_macro[$macro++] = $j-1 ;          # PreLoad Macro

```

Note: This Number of leading - dashes designates the hierarchy of the folders. The 1st space after the last - dash is important!

#####

Note: A word about key bindings. A tool key binding is like <a> which binds the "a" key to the specified tool. If you want to use a Tk button, but not have a key binding, declare a non-key value for the hash table to lookup. My example is <0x01> for check_out and <0x02> for check_in. These can actually be any non-key values as long as they are unique

#####

```
$tee = " | tee $slog_file >> $log_file "; # Generate Standard Logs using tee
```

```

@my_tool[$j++]="-- RTL Tools ----- # ";
@my_tool[$j++]="-- Library Tools ----- # ";
@my_tool[$j++]=" check_out (int) <0x01> # check_out";
@tool_macro[$macro++] = $j-1 ;
@my_tool[$j++]=" check_in (int) <0x02> # check_in";
@tool_macro[$macro++] = $j-1 ;
@my_tool[$j++]=" check_abandon (int) # check_abandon";
@my_tool[$j++]=" check_restore (int) # check_restore";
@my_tool[$j++]=" check_in_out (int) <tab> # check_in_out";
@my_tool[$j++]=" Diff_COD_file (results) # diff FILE_NAME " .
    $wip_header . "FILE_NAME $tee";
@my_tool[$j++]="-- Issue Tracking ----- # ";
@my_tool[$j++]=" Issue_List_Add (int) # Issue_List_Add";
@my_tool[$j++]=" Issue_List_View (int) <0x04> # Issue_List_View";
@my_tool[$j++]=" Issue_List_Modify # $my_editor $issue_file ";
@my_tool[$j++]="-- RTL Generation ----- # ";
@my_tool[$j++]=" generate_VHDL (int) <0x05> # generate_VHDL";
@my_tool[$j++]=" generate_Verilog (int) <0x06> # generate_Verilog";
@my_tool[$j++]="-- Label Release Tools -- # ";
@my_tool[$j++]=" label_release (int) # label_release";
@my_tool[$j++]=" TAR_it_up (hier_exp) # tar -rvf LABEL_RELEASE.tar FILE_LIST >>
@my_tool[$j++]=" create_branch (int) # create_branch";
@my_tool[$j++]="-- CV_World Tools -- # ";
@my_tool[$j++]=" encrypt_file (int) # encrypt_file";
@my_tool[$j++]=" import_web_files (int) # ";
@my_tool[$j++]="-- Port Viewing ---- # ";
@my_tool[$j++]=" port_view_parent (int) <p> # port_view_parent ";
@my_tool[$j++]=" port_view_kids (int) <b> # port_view_kids ";
#####

```

```

@my_tool[$j++]="-- EDA Tools ----- # ";
@my_tool[$j++]="-- ModelSim ----- # ";
Note: The following are ModelSim specific Configurations. Change accordingly
$mti_work_path="../modelsim/". $my_name ./work"; # Relative Path to MTI

if ( $os eq "unix" )
{
@my_tool[$j++] = " Vcom (results)(hier) <c> # vcom -explicit FILE_NAME -work $mti_wor
@my_tool[$j++] = " Build MTI Work Directory (int) # build_vlib";

@my_tool[$j++] = " Vsim_Simulate # vsim MOD_NAME -lib $mti_work_path $tee

}
else
{
@my_tool[$j++] = " Compile (hier) # d:\\mti5.4e\\win32\\vcom.exe FILE_NAME -work $m
@my_tool[$j++] = " Simulate # d:\\mti5.4e\\win32\\vsim.exe MOD_NAME -lib $mti_

}

@my_tool[$j++]="-- Cadence ----- # ";
@my_tool[$j++]=" TBD (results)(hier) # foo FILE_NAME $tee";
@my_tool[$j++]="-- Synopsys ----- # ";
@my_tool[$j++] = " Synthesize (fork)(hier)(results) # synth.sh MOD_NAME $tee";
@my_tool[$j++]="-- Xilinx ----- # ";
@my_tool[$j++]=" Generate_Xilinx_Project (int) # xilinx_proj ";
@my_tool[$j++]="-- Altera ----- # ";
@my_tool[$j++]=" Generate_Altera_Project (int) # altera_proj ";
@my_tool[$j++]="-- Synplicity ----- # ";
@my_tool[$j++]=" Generate_Synplify_Project (int) # synplify_proj ";
@my_tool[$j++]="-- GCC ----- # ";
@my_tool[$j++]=" C-Compile (results)(hier) # gcc -c FILE_NAME $tee";
@my_tool[$j++]=" C-Linker (results) # link.sh $tee";
@my_tool[$j++]="-- Misc ----- # ";
@my_tool[$j++]=" block_print (hier)(int) # block_print";

#####
@my_tool[$j++] = "-- UNIX ----- # ";
@my_tool[$j++] = " Execute_Script (results) <x> # FILE_NAME $tee";
@my_tool[$j++] = " UNIX_Shell (fork) # xterm -T UNIX_Shell &";
@my_tool[$j++] = " UNIX_Shell (fork) # gnome-terminal -sdf-name UNIX_Shell >> $log_file

@my_tool[$j++] = " Make_TarBall (hier_exp)(results) # tar -rvf PARAM1.tar FILE_LIST $tee";
$my_tool_param{($j-1)} = "#PARAM1 = Tar File Name (no ext, ie foo)#:";

@my_tool[$j++] = " View_TarBall (results) # tar -tf PARAM1.tar $tee";
$my_tool_param{($j-1)} = "#PARAM1 = Tar File Name (no ext, ie foo)#:";

@my_tool[$j++] = " File_List (hier)(results) # ls -l FILE_NAME $tee";

#####
@my_tool[$j++] = " Grep (hier)(results) # grep PARAM1 FILE_NAME $tee";
$my_tool_param{($j-1)} = "#PARAM1 = Search Pattern :".
"";

#####
if ( $admin_hash{$my_name} == 1 )
{
@my_tool[$j++] = "-- Admin Tools ----- # ";
@my_tool[$j++] = " chmod_ReadOnly (hier) # chmod a-w FILE_NAME >>$log_file";
@my_tool[$j++] = " chmod_ReadWrite (hier) # chmod a+w FILE_NAME >>$log_file";

} # Note: These tools only show up for 'admin' users

#####
... Insert more tools here:
@my_tool[$j++] = "-- User Defined ----- # ";
@my_tool[$j++]=" a_user_tool # a_user_tool FILE_NAME";
@my_tool[$j++]="-- Example Ext Tools ---- # ";
@my_tool[$j++]=" InsertNet in VHDL(hier) # insert_net.pl FILE_NAME FILE_NAME PARAM1
This Describes the params that the above tool needs:
$my_tool_param{($j-1)} = "#PARAM1 = Net Name (ie foo) :".

```



```
"#PARAM2 = Net Direction (ie inout) :".
"#PARAM3 = Net Width      (ie 32)   :".
"";
```

```
#####
@my_tool[$j++]="- CV_Options      --      #      ";
@my_tool[$j++]="-- Display Options ---  #      ";
@my_tool[$j++]=" rotate_view      (int)    <0x07> # rotate_view      ";
@my_tool[$j++]=" rotate_style     (int)    # rotate_style     ";
@my_tool[$j++]=" rotate_x_scale   (int)    # rotate_x_scale   ";
@my_tool[$j++]=" resize_display    (int)    <r> # resize_display   ";
@my_tool[$j++]=" split_display_size_adj (int) <S> # split_display_size_adj ";
@my_tool[$j++]=" split_display_mode_adj (int) <s> # split_display_mode_adj ";
@my_tool[$j++]=" change_color      (int)    # change_color     ";
@my_tool[$j++]="-- Configuration Options  --      #      ";
@my_tool[$j++]=" List User Variables (int)    # list_user_vars   ";
@my_tool[$j++]=" View User INI File   # $my_editor $user_ini_file";
@my_tool[$j++]="-- CV Debug+Develop  --      #      ";
@my_tool[$j++]=" Enable Trace Log    (int)    # enable_trace_log";
@my_tool[$j++]=" View Trace Log      # $my_editor $trace_file";
@my_tool[$j++]=" Expand_CV_Source    (int)    # expand_cv_source ";
@my_tool[$j++]="-- Contact ChipVault      #      ";
@my_tool[$j++]=" Send_Author_Email  (int)    # Send_Author_Email ";
@my_tool[$j++]=" change_list        (int)    # change_list      ";
@my_tool[$j++]=" Web to chipvault.sourceforge.net # $browser chipvault.sourceforge.net ";
@my_tool[$j++]="--"; # This is a Pragma to set the level
@my_tool[$j++]=" about              (int)    # about            ";
#####
@my_tool[$j++]="--"; # This is a Pragma to set the level back to root
@my_tool[$j++]=" View Long Log File  # $my_editor $log_file";
@my_tool[$j++]=" View Short Log File # $my_editor $slog_file";
@my_tool[$j++]=" tool_menu          (int)    <t> # tool_menu       ";
@my_tool[$j++]=" help               (int)    <h> # help            ";
@my_tool[$j++]=" quit              (int)    <q> # quit           ";
```

User Definable Report File Associations:

Perl Source Line Number: 1522

```
#####
Define Reports to View in right columns
#####
$j=2; # Note. 0&l taken already for Source
#####
$report_filename_title [ $j ] = "File Modification Time Stamp"; # Title
$report_function       [ $j ] = "time_stamp";
$report_filename_rule  [ $j++ ] = "FILE_NAME";
#####
$report_filename_title [ $j ] = "HistoryLog"; # Title
$report_filename_rule  [ $j++ ] = $hist_log_header."FILE_NAME".$hist_log_footer;
#####
$report_filename_title [ $j ] = "TAR-Ball Archive List"; # Title
$report_filename_rule  [ $j++ ] = $star_list_header ."FILE_NAME".$star_list_footer;
#####
$report_filename_title [ $j ] = "Synopsys Netlist"; # Title
$report_filename_rule  [ $j++ ] = "../synopsys/map/vhdl/MOD_NAME.vhd";
#####
$report_filename_title [ $j ] = "Synopsys Area Report"; # Title
$report_filename_rule  [ $j++ ] = "../synopsys/report/MOD_NAME.area.report";
#####
$report_filename_title [ $j ] = "Gate Estimate Analysis"; # Title
$report_function       [ $j ] = "gate_estimate";
$report_filename_rule  [ $j++ ] = "../synopsys/report/MOD_NAME.area.report";
#####
$report_filename_title [ $j ] = "Flip-Flop Count Analysis"; # Title
$report_function       [ $j ] = "flop_count";
$report_filename_rule  [ $j++ ] = "../synopsys/map/vhdl/MOD_NAME.vhd";
#####
$report_filename_title [ $j ] = "Worst Slack Analysis"; # Title
$report_function       [ $j ] = "worst_slack";
$report_filename_rule  [ $j++ ] = "../synopsys/report/MOD_NAME.verbosetiming.report";
#####
$report_filename_title [ $j ] = "Synopsys Timing Report"; # Title
$report_filename_rule  [ $j++ ] = "../synopsys/report/MOD_NAME.verbosetiming.report";
#####
$report_filename_title [ $j ] = "Synopsys Timing Report"; # Title
$report_filename_rule  [ $j++ ] = "../synopsys/report/MOD_NAME.registertiming.report";
#####
```

User Definable VHDL Template:

Perl Source Line Number: 1569

```
#####
This is the VHDL Template. Go ahead and change everything but the #T's
#####
-- *****
-- (C) Copyright 2002 #COMPANY_NAME#
-- All rights reserved.
--
-- Source file: #FILENAME#
-- Date:         #DATE#
-- Author:       #AUTHOR#
-- Description:  #DESCRIPTION#
--
-- THIS SOURCE FILE IS PROVIDED "AS IS" AND WITHOUT ANY
-- EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT
-- LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
-- FITNESS FOR A PARTICULAR PURPOSE.
--
-- In no event will the Author, Kevin M. Hubbard be liable for direct,
-- indirect, special, incidental, or consequential damages resulting from the
-- use of this RTL template, even if advised of the possibility of such damages.
--
-- Use of this RTL template in the design or control of machinery involved in
-- 'HIGH-RISK' activities, i.e. activities where failure of this software
-- could reasonably be expected to cause DEATH, INJURY or the RELEASE OF
-- HAZARDOUS MATERIALS, IS NOT PERMITTED.
--
-- Revision History:
-- Ver#  When      Who      What
-- ----  -
-- 0.1    #AUTHOR# Created.
-- *****
LIBRARY ieee ;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
-- USE ieee.std_logic_textio.all;
-- LIBRARY std ;
-- USE std.textio.all;

entity #MOD_NAME# is
port
(
#LOCATION_PORT_DECLARATION_LOCATION#
#SIG_NAME# : #SIG_DIR# #SIG_WIDTH# ;
ETC ETC ETC.....
);

end #MOD_NAME#;

architecture #ARCH_TYPE# of #MOD_NAME# is

--#CHILD_LINE#component #CHILD_NAME#
--#CHILD_LINE#port
--#CHILD_LINE# (
--#CHILD_LINE# #LOCATION_CHILDREN_COMPONENT_DECLARATION_LOCATION#
--#CHILD_LINE# );
--#CHILD_LINE#end component ; -- #CHILD_NAME#

type state_type is
(
s0_idle,
s1_busy
);

type ram_type is array (511 downto 0 ) of std_logic_vector ( 7 downto 0 );
attribute state_vector : string;
attribute state_vector of #ARCH_TYPE# : architecture is "current_state" ;
```

```

signal    current_state, next_state    : state_type ;

#LOCATION_SIGNAL_DECLARATION_LOCATION#
-- signal #SIG_NAME# : #SIG_WIDTH# ;
signal clk                : std_logic;
signal reset              : std_logic;
signal foo                : std_logic_vector ( 7 downto 0 ) ;
signal ram                : ram_type ;
signal read_addr         : std_logic_vector ( 8 downto 0 ) ;

begin

-----
-- Process
-----

foo_proc : process ( clk )
begin
  if ( clk'event and clk = '1' ) then
    if ( reset = '1' ) then
      foo <= ( others => '0' );
    else
      foo <= foo ( 7 downto 0 );
    end if;-- if ( reset = '1' ) then
  end if;-- if ( clk'event and clk = '1' ) then
end process foo_proc;

clocked_fsm : process ( clk )
begin
  if ( clk'event and clk = '1' ) then
    if ( reset = '1' ) then
      current_state <= s0_idle;
    else
      current_state <= next_state;
    end if;-- reset
  end if;-- clk
end process clocked_fsm;

-----
-- nextstate assignment based on inputs and current_state
-----

nextstate : process ( current_state , reset )
begin
  case current_state is
    when s0_idle =>
      if ( reset = '1' ) then
        next_state <= s1_busy;
      else
        next_state <= s0_idle;
      end if;
    when others =>
      next_state <= s0_idle;
  end case;
end process nextstate;

-----
-- Infer a RAM for the FPGA guys
-----

ram_proc : process ( clk )
begin
  if ( clk'event and clk = '1' ) then
    if ( we = '1' ) then
      ram(conv_integer(addr) ) <= di ( 7 downto 0 );
    end if;
    read_addr <= addr ( 8 downto 0 );
  end if;
end process ram_proc;
do <= ram ( conv_integer( read_addr ) );

#LOCATION_INSTANTIATE_CHILDREN_LOCATION#

end #ARCH_TYPE#;

```

#####

User Definable Verilog Template:

Perl Source Line Number: 1716

```
#####
This is the Verilog Template. Go ahead and change everything but the #V's
#####
// *****
// (C) Copyright 2002 #YOUR_COMPANY_NAME#
// All rights reserved.
//
// Source file: #FILENAME#
// Date: #DATE#
// Author: #AUTHOR#
// Description: #DESCRIPTION#
//
// THIS SOURCE FILE IS PROVIDED "AS IS" AND WITHOUT ANY
// EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT
// LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
// FITNESS FOR A PARTICULAR PURPOSE.
//
// In no event will the Author, Kevin M. Hubbard be liable for direct,
// indirect, special, incidental, or consequential damages resulting from the
// use of this RTL template, even if advised of the possibility of such damages.
//
// Use of this RTL template in the design or control of machinery involved in
// 'HIGH-RISK' activities, i.e. activities where failure of this software
// could reasonably be expected to cause DEATH, INJURY or the RELEASE OF
// HAZARDOUS MATERIALS, IS NOT PERMITTED.
//
// Revision History:
// Ver# When Who What
// ---- -
// 0.1 #AUTHOR# Created.
//
module #MOD_NAME#(
    #SIG_NAME#,
    #LOCATION_PORT_DECLARATION_LOCATION#
    ETC ETC ETC.....
);

#SIG_DIR# #SIG_WIDTH# #SIG_NAME# ;
#LOCATION_SIGNAL_DECLARATION_LOCATION#

input foo_clk ;
output foo_out ;
input [31:0] foo_bus ;
reg foo ;
wire bar ;

always @(posedge clk)
begin
foo <= bar ;
end

foo u_foo
(
    .clk( clk ),
    .reset( reset )
);

#LOCATION_INSTANTIATE_CHILDREN_LOCATION#

endmodule
```

On-Line Help:

THIS IS THE EMBEDDED HELP #####

ChipVault Help: (Ctrl-D to page down)

Disclaimer and Terms of Use:

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the enclosed GNU General Public License for more details.

In no event will the Author, Kevin M. Hubbard be liable for direct, indirect, special, incidental, or consequential damages resulting from the use of this software, even if advised of the possibility of such damages.

Use of this software in the design or control of machinery involved in 'HIGH-RISK' activities, i.e. activities where failure of this software could reasonably be expected to cause DEATH, INJURY or the RELEASE OF HAZARDOUS MATERIALS, IS NOT PERMITTED.

Web-Site: <http://chipvault.sourceforge.net>

Please email the author at khubbard@users.sourceforge.net to let him know you have tried ChipVault and what you like/dislike about it. The author is always looking for improvement ideas.

Table of Contents:

- i) Key Codes
- ii) Vi+Emacs Help
- iii) FAQ
- iv) Author Info
- v) GPL License

i) Key Codes:

Note: This is for using Vi Non-Friendly Cursors

```
[ Navigation ]
<UP> or <i> : Navigate Up Hierarchy
<CTRL-U> : Navigate Page Up Hierarchy
<DOWN> or <k> : Navigate Down Hierarchy
<CTRL-D> : Navigate Page Down Hierarchy

<LEFT> or <j> : Navigate Left
<RIGHT> or <l> : Navigate Right

</> : Search forward on Module Name
<?> : Search backward on Module Name
```

Note: This is for using Vi Friendly Cursors

```
#H <UP> or <k> : Navigate Up Hierarchy
#H <CTRL-U> : Navigate Page Up Hierarchy
#H <DOWN> or <j> : Navigate Down Hierarchy
#H <CTRL-D> : Navigate Page Down Hierarchy
#H
#H <LEFT> or <h> : Navigate Left
#H <RIGHT> or <l> : Navigate Right
```

[Hierarchy Expanding/Collapsing]

```
<SPACE> : Expand and Collapse hierarchy Views.
<l> to <9> : Collapse all below level < >.
```

[Schematic'ish Block Viewing]

```
<p> : View Ports into each Block.
Inputs on the left an
```

outputs and bidis on the right.
 : View Children Blocks. Same as <p> but shows all the children instantiated by the current block.

[Tool Control]
<ENTER> : Perform Tool Bar Operation on File (Edit, CheckIn, etc)
<t> : Tool Popup Scroll Menu. Select tool and press <ENTER>
<!> : Bang History. Recall prior used tools via <1>..<9>
<@> : Macro Tools. Call pre-assigned tools via <1>..<9>

[Misc]
<r> : Resize Screen and Redraw.
<s> : Toggle Split Display Modes.
<d> : Toggle Debug Mode. Useful for displaying key codes.
<q> : Quit

[Macro Assigned Keys]
<c> : Compile
<e> : Edit
<v> : View Log File
<tab> : Checkfile In/Out

ii) Vi + Emacs Help
Vi Help (Brief):

[Saving and Exiting]
":w" : Write file to disk
":q" : Quit Vi. (use :q! to abandon edits)
":w foo" : Write file foo to disk
":r foo" : Read and Insert file foo from disk
":e foo" : Edit file foo

[Switching between Commands and Edit modes]
"<ESC>": Switch from text edit mode to command mode.
"i" : Start inserting text at current cursor position
"R" : Start overwriting text at current cursor position
"A" : Start overwriting text at end of current line

[Cut and Paste]
"2yy" : Yank-Yank 2 Lines (Copy) at current cursor position
"3dd" : Delete-Delete 3 Lines (Cut) at current cursor position
"p" : Paste Yanked or Deleted line(s) to current cursor.
"dw" : Delete Word right of the cursor.
"5d<sp>": Delete 5 characters right of the cursor.

[Scrolling]
":1" : Go to Line 1 (Top of File)
":n" : Go to Line n
":\$" : Go to Last Line (End of File)
":set number" : Display Line Numbers

[Search]
"/foo" : Search forward in file for "foo"
"?foo" : Search backward in file for "foo"

[Search and Replace]
":.,+8s/foo/bar/g" : Replace "foo" with "bar" on the next 8 lines.

[Block Move]
(Yes a mouse would be nice here.)
1) Position Cursor at top of block, leftmost column.
2) "ma~".
3) Position Cursor at bottom of block, leftmost column.
4) "ms\".
5) "`sd`a" (can sub "d" for "y" for Yanking instead of Deleting).
6) Paste ("p") at desired location.

(End VI Help)

Emacs Help (Brief):

[Saving and Exiting]
"Ctrl-X Ctrl-S" : Save
"Ctrl-X Ctrl-C" : Quit

[Navigating]
"Ctrl-A" : Start of Line
"Ctrl-E" : End of Line

[Cut and Paste]
"Ctrl-Space" : Set Mark
"Ctrl-W" : Cut Mark to Cursor
"Ctrl-Y" : Paste
"Ctrl-X U" : Undo
"Ctrl-K" : Delete Cursor to End of Line

(End Emacs Help)

On-Line FAQ:

iii) Frequently Asked Questions (FAQ)

1) What is ChipVault?

ChipVault is an interactive Perl program for controlling hierarchy of a chip design. It can handle both VHDL and Verilog, or any clear text design effort that may be sorted by hierarchy.

2) What are the main features?

- a) Hierarchy Organization of Source Files.
- b) Simple Revision Control for Group Projects.
- c) Instant Port Views for VHDL subblocks.
- d) Issue Tracking Database.
- e) Auto Instantiate VHDL and Verilog blocks.
- f) Hardware and OS Flexible.
- g) Free (Open-Source GPL'd).

a) Hierarchy Organization of Source Files:

ChipVault organizes your source files via hierarchy and allows you to rapidly navigate thru hundreds of source files deep into a design. CV can be used to launch your favorite editor like Vi or Emacs. ChipVault is fully customizable allowing you to launch other tools such as compilers, Synthesis, Perl scripts, Lints, etc.

Since ChipVault knows the hierarchy of your design, any tool you configure for launching from CV may be launched for bottom up hierarchy execution.

b) Simple Revision Control for Group Projects.

ChipVault provides a Check_Out and Check_In feature for source files. Checking a file out generates a copy of the file for you to edit. The original file remains untouched until you check in your changes, allowing a group of designers to share the same source tree.

WhatThisIsnt : This isnt RCS, CVS, etc. . IMHO those tools are great for big SW project, but overkill for HW designs and they slow down the design effort with an extra layer of complexity. Revision Control in ChipVault is tailored for HW designers. Only one designer may check out a file at a time.

HowThisWorks : When you check out "foo.vhd", ChipVault creates a copy called "wip.foo.vhd". So long as "wip.foo.vhd" exists, no one else may check out foo.vhd. The mainline foo.vhd still exists and will exist until you check "wip.foo.vhd" back in. When you check the file back in, you are prompted to describe the changes made which is then automatically added to a ChangeLog file. When file archiving is enabled (default), every time a user checks a file in or out, the file is automatically added to a hidden gzip'd tarball file. RTL is highly compressible, so it is practical to archive everything.

c) Instant Port Views for VHDL subblocks.

One of ChipVaults built-in tools is a VHDL reader for reading entity declarations and deciphering port declarations. ChipVault uses this information to draw a block diagram of all signals going in and out of a block.

d) Issue Tracking Database.

ChipVault will generate a data base for you for tracking issues that come up during your design phase. Issues can be sorted by ID, Reporter, AssignTo, Title, etc.

Everything is stored in a readable text file issues.txt.

- e) Auto Instantiate VHDL and Verilog Blocks.
ChipVault has the ability to read in your existing RTL blocks and generate new template files on the fly which instantiate your existing block.
- f) Hardware and OS Flexible.
ChipVault runs on UNIX, Linux and Windows. The UserInterface scales from a dumb terminal 80x24 Telnet display to 1280x1024 (or 160x128) making it an ideal tool for large designs as well as dialup access.
- g) Free (Open-Source GPL'd).
ChipVault is Open-Source GPL'd, meaning all are free to download ChipVault from <http://chipvault.sourceforge.net> and execute ChipVault without having to pay any license fees. The author maintains copyrights to the ChipVault source, preventing 3rd parties from modifying the source and then selling a compiled commercialized application. Please read the full enclosed GNU General Public License for full details on license agreement.

3) Which version should I use?

For serious work, the UNIX/Linux port of course.
The author uses the basic POSIX console version on a day to day basis.

A Curses version existed once which is slightly faster than POSIX. It was removed as installing Curses libs for Perl wasnt trivial and the benefits (like mouse support) were minor compared to the overhead of supporting.

Win32 users now only have the Console version for ActivePerl.

but Caveot Emptor, different versions of Win32 OSs out of Redmond tend to have different Console IO behaviors running the same version of ActivePerl. The author runs on Win2K but the Console has to be setup for BufferSize equal to ScreenSize. By default, the Console window has a buffer larger than the screen which then maps the cursor keys for scrolling the buffer (and ChipVault never hears the key presses). The author generates a shortcut which has the Buffer set to 25 lines.

```
ie: ShortCut Properties which work for win2K
The Tab Buttons are represented via [] brackets.
[Shortcut]
  Target      : perl.exe cv_win32.pl
  Start in   : Your_Design_Path_Containing_hlist.txt
[Layout]
  ScreenBufferWidth  : 80
  ScreenBufferHeight : 25
  Window Size Width  : 80
  Window Size Height : 25
```

Once this shortcut is created, you can double-click to start ChipVault on your desired design with the Console set properly.

- 4) I hate all GUIs. What good is ChipVault to me?
Use ChipVault in batch mode for doing bottom-up stuff like Compiling and Synthesis.
ie: %cv.pl hlist.txt -c "cp FILE_NAME FILE_NAME.foobak"
will make a foobak copy of every file in your design.
Insert a Pound inside your hlist.txt for files to skip.
- 5) I don't want to use the ChipVault GUI but I need to participate on a group project using ChipVaults simple Revision Control System. How do I check files in and out via the UNIX command line?

Thats easy.

To see who has what already checked out:

```
"ls -l wip.*.vhd"
```

To check out a file:

```
"cp foo.vhd wip.foo.vhd"
```

To check a file back in:

```
"cp wip.foo.vhd foo.vhd"
```

Note: Auto-archiving will not be performed using this method. You'll need to manually add the files to existing TAR-balls if you need an archive.

- 6) How do I assign my own tools for the tool bar?
Grep on `my_tool` assignments and copy examples.
Tags like `FILE_NAME` get replaced by current selected file.
- 7) What is/is_not supported in the Win32 port?
The author tried in earnest to make ChipVault portable over to a Win32 environment.
Unfortunately POSIX terminal emulation is spotty for Win32 ports of Perl, so the native Win32 console interface instead. To run on a Win32 system you need to install ActivePerl from www.activestate.com. This is a port of standard Perl5 with some hooks to Win32 routines.
By definition, Win32 is lacking in features that come standard in UNIX environments. ChipVault on Win32 does Not Supported:
 - o GZIPping.
 - o File ownership information. (checked out files will be listed as owned by "USER")
 - o Email notifications on CheckIns, New Issues.
 - o etc,etc,etc.

On a plus side, the Win32 port does have some limited mouse support. Double-Click on "+" and "-" to expand and collapse views. Double-Click on Module Names to activate tool. Single Click to change cursor position. For some reason, when system calls are made and fail (ie compiling or something), the mouse seems to break.

So, in-short, ChipVault was written for UNIX/Linux, but it seems to work on Win32 too, so here it is.
- 8) Why is my Win32 window so small?
ChipVault runs as a console app on Win32. Win32 defaults to 24 lines. Change your window properties for more lines after starting ChipVault then press <r>esize. Also see the generated `cv.lnk` file.
- 9) How do I assign my own keyboard macros to avoid the tool bar?
Grep on `macro_hash` to see examples of setting up macros.
- 10) How do I get Mouse support on Linux/UNIX?
Mouse support once existed under Curses but was removed as the author never used it and the overhead to support Curses wasn't worth the hassle. ChipVault is much like Vi, Pine, MightNightCommander - fast and efficient via rapid keystrokes.
- 11) How do I enable Email Notification on file CheckIns?
Email notification on file CheckIns is turned off by default. To turn this feature on, you need to set up the `email_list` array with all the people to Email to. Then set the variable `send_email_on_check_in` to 1.
- 12) How do I setup a component library for all designers to have access to?
Fill in the `lib_list` array to point to other `hlist.txt` files that you want read in. Or manually insert a line like `[include] foo.txt`

in your hlist.txt file. When ChipVault reads in hlist.txt it will spot the [include] tag and automatically read in foo.txt as well.

- 13) Why is ChipVault slow on screen refreshes on my machine? Believe it or not, this isn't because ChipVault is interpreted by Perl rather than compiled into machine code. On some machines, the User Interface may be slow as ChipVault uses POSIX compliant terminal IO commands (think VT100) for doing things like cursor placement, screen erases, etc. On some OS's this can be really slow. I've noticed that a SunUltra60 is sluggish, but my AMD AthlonXP-2000 Linux box just screams. If you are stuck with a slow OS, you can still speed things up by giving ChipVault less to draw on every keystroke.

Ways to speed up the UI are:

- o) Disable Split-Display Mode with:
\$enable_split_display = 0; (or press <s> until it goes away)
 - o) Turn off "file=" window with:
\$disp_file_in_main = 0;
\$disp_file_xterm_title = 0;
 - o) Shrink your window size (ie 25 or 50 lines instead of 120).
 - o) Switch from Solaris to Linux.
- 14) How do I import libraries for browsing?
Either explicitly type an [include] lib.txt into your hlist file, or set the CV_INCLUDE environment variable to lib.txt. ie:
setenv CV_INCLUDE /home/me/foo.txt;/home/you/bar.txt will automatically append the 2 libraries foo and bar to your CV hlist session.
- 15) What is split_display?
When your Xterm screen is sufficiently large in either the X or Y direction, ChipVault will enable split_display and simultaneously display your hierarchy view alongside a port view of the block the cursor is on.
- 16) What is CV_World?
CV_World (or ChipVault World) is a feature of ChipVault for handling global design projects via the Web, either unsecured (open-source cores), or 448bit encrypted.

Open-Source Example: A group of grad-students at 12 different universities are working on a PC-in-a-chip design. Each designer keeps his RTL posted on a university web server and sends the URLs (not the files) to the other 11 team members who then copy the URL into their CV hlist. On ChipVault invocation, ChipVault then copies the remote URL files to a local file(mirror).

HLIST.TXT Example:

```
foo foo.v http://www.site.com/foo.v
```

Proprietary-Example: Two companies are working on a chip and need to share IP on a daily basis. Designers use ChipVault with Blowfish 448bit encryption to secure their files before placing them on a public web-server. The 448bit keys are then inserted (along with URL) in the hlist of remote site for web access. ChipVault will then autocopy over any updates every time ChipVault is invoked.

HLIST.TXT Example:

```
foo foo.v http://www.site.com/foo.v.cv_world=EncryptionKey  
/ /  
Encrypted version of foo.v --- 8-56 ASCII key -  
generated by ChipVault for En/DeCryption
```

- 17) How can I Help the ChipVault project?
- o) Use ChipVault and spread the word. The author's marketing budget for last year was \$0.00, so word-of-mouth is key.
 - o) Send author Email feedback on problems and feature requests.
 - o) Fix bugs and send author patches to be included in future releases.

18) Can I get phone-support?

Try email support first. If we find you really need phone support, the author is available on Saturdays and Sundays. You'll need to set up a 800 dialin conference call thru ATT or something. Email support works quite well with response time generally less than 12 hours.

Tutorial Text:

iv) Tutorial Text:
ChipVault Tutorial 11-12-2002

This Tutorial is a quick 10 minute walk-thru of the major features in ChipVault. You will need to de-tar the example files from <http://chipvault.sourceforge.net/tutorial.tar.gz>

```
%gunzip tutorial.tar.gz
%tar -xvf tutorial.tar
```

Index:

- Step-1) File Setup
- Step-2) Reading in an existing Design and generating a hlist.txt file
- Step-3) Navigating the ChipVault Hierarchy Interface
- Step-4) Port Viewing
- Step-5) File Editing
- Step-6) Tool Bar
- Step-7) Revision Control. CheckIn/CheckOut
- Step-8) Issue List
- Step-9) VHDL Module Generation
- Step-10) Bottom Up Compiles
- Step-11) Schematic Block Printing
- Step-12) RTL Viewer
- Step-13) Admin Control of Checkins
- Step-14) Netlist Viewing

Step-1) File Setup

- o) Copy cv.pl into the tutorial/src directory
- o) %cd tutorial/src

Step-2) Reading in an existing Design and generating a hlist.txt file

- o) %perl cv.pl top.vhd<ENTER>
This will read in the top.vhd file and find all the children by scanning thru the *.vhd and *.v files in the same subdirectory.
- o) Press <ENTER> when prompted for the Filter Value.

If the VHDL and Verilog files were read in OK, you should now see a hierarchy tree of the example design.

- o) Quit ChipVault by pressing <q>. Look at the hlist.txt file that CV built and then restart ChipVault in the normal fast mode:
- o) %perl cv.pl<ENTER> (or just %cv.pl if perl is in path and chmod +x)

Step-3) Navigating the ChipVault Hierarchy Interface

- o) Use the cursors keys (or <i,j,k,l> to go up and down the hierachy. Note the "file=" box will display the actual file name you are on.
- o) Navigate to "german" and press <SPACE> bar. This should collapse the "german" module and all of its children modules should now be invisible.
Press <SPACE> bar again, and the child modules should re-appear.
- o) Press <3>. This will collapse everything below 3 levels deep.
- o) Press (4). This will collapse everything below 4 levels deep.

Step-4) Port Viewing

- o) Navigate to module "top" and press the <p> key.
This will display all the port I/O to this module. Press <SPACE>.

Step-5) File Editing

- o) Navigate to module "metrics" and press <ENTER> key.
This should launch VI and open metrics.vhd.
- o) Type <:> <q> to exit VI and go back to ChipVault.

Step-6) Tool Bar

- o) Press <t> to bring the Tool Bar up.

- o) Scroll thru the Tool Bar window using up and down cursor keys (or <i> and <k>) until "Edit (Emacs)" under "Editors" appears and press <ENTER>.

This should bring up "metrics.vhd" in a Emacs Edit window.
The <ENTER> key is now assigned to "Edit (Emacs)" as indicated by the top line of the ChipVault screen.

- o) Quit Emacs and Cursor down to "modelo" and press <ENTER> and another instance of Emaces will be launched, but with modelo.vhd.
- o) Quit Emacs and Cursor down to "german" and <t>ool to Edit (fork) This will launch Vi and send the process in the background, keeping ChipVault active.
- o) Press <!> to bring up Bang History. Press <2> to select the Emacs.
- o) Press <@> to bring up Macro Tool. Press <1> to select Edit.

Step-7) Revision Control. CheckIn/CheckOut

a) Check Out a File

- o) Cursor back to "metrics", hit <t> for tool and select "check_out" <ENTER>. (or just press <TAB> key over metrics)

You should have just checked out the file metrics.vhd .
If this worked, to the right of "metrics" you should see your user name in the "CheckedOut" column.

- o) Cursor over to your user name and you should see "wip.metrics.vhd" at the bottom-left of the screen instead of original "metrics.vhd".

This Work-In-Progress (wip) file is a copy of metrics.vhd for you to work on.

- o) Press <e> to make changes to wip.metrics.vhd using your default editor.

b) Diff your Checked Out file

- o) Cursor back to the left column over "metrics" and press <t> and select "Diff CO'd file".
You should get an error message from Diff. Cursor down to "log.txt" and select "Edit" from the tool bar to view the diff output.

c) Check In a File

- o) Cursor back up to your username under the CheckedOut column to the right of "metrics". Tool Bar select "check_in".
(or just press <TAB> key over username)

- o) When prompted, type a short sentence about the change you made. ChipVault will now copy "wip.metrics.vhd" to "metrics.vhd" In the process it will destroy the original "metrics.vhd" and create a tar archive of the new "metrics.vhd" you modified.

d) View Change Log

- o) Cursor to "metrics" then cursor right twice.
The top-left should show [2 HistoryLog] and your cursor should be on history_log.metrics.vhd.txt.

- o) Select "Edit" from your tool bar and view the history change file.

e) View Archive Log

- o) Cursor right again
The top-left should show [3 TAR-Ball Archive List]
View the archive_list.metrics.vhd.txt file to see what archives have been auto-generated by ChipVault.

Step-8) Issue List

- o) Select Issue_List_View from the Tool Bar.
Press <1> to sort the issues by issue number.
Press <2> to sort the issues by Reporter.
Press <3> to sort the issues by AssignTo.
Press <4> to sort the issues by Module name.

Scroll to a desired issue and press <ENTER> to view the full description.

- o) Select tool bar from the main screen and walk thru the issue generation process.

Step-9) VHDL/Verilog Module Generation

- o) Cursor over to "grolsch" and tool-bar select "generate_VHDL"
- o) Type in "austrian" <ENTER> <ENTER> <ENTER>
- o) Type <y> when prompted for instantiating "grolsch.vhd"
- o) Type "foo" <ENTER> 16 <ENTER> in <ENTER> for adding new foo(15:0).
- o) Type "bar" <ENTER> <ENTER> <ENTER> for new net bar(15:0).
- o) Type "bob" <ENTER> 1 <ENTER> out <ENTER> for new net bob.
- o) Type <ENTER> <ENTER> <ENTER> to exit this loop.
- o) Cursor to the bottom of the ChipVault screen and edit "austrian" module at the very bottom. This should be a new VHDL module with the nets you described and it should instantiate "grolsch.vhd".
- o) You've added a new module, now you need to place it in the design hierarchy. You could edit "beers", add "austrian" insantiations, delete the original hlist.txt file and start-over from step-1
OR
- o) cursor up to top of the screen to hlist.txt and edit.
Type a new line above "german" with "austrian" and "austrian.vhd"
The cursor position of the 1st "a" of austrian MUST be at the same spot as the "g" of "german". This is how ChipVault knows the design hierarchy. Yank and Pase the grolsch line from under "german" and place it under "austrian".
- o) Quit ChipVault (<q>) and restart chip vault with no params.
%perl cv.pl

You should see your new module in the correct place in hiearchy

Note: Module Generation allows you to instantiate Verilog from VHDL and vice-versa.

Step-10) Bottom Up Compiles

- o) Get ModelSim stuff setup properly (see example).
- o) Cursor to "german" and press <SPACE> to collapse.
- o) ToolBar select "Compile".
This will vcom german.vhd. Any errors will dump to log.txt
- o) Cursor to "german" and press <SPACE> to expand.
- o) ToolBar select "Compile".
This will vcom cap.v, light.vhd, dark.vhd, amstel, heini, becks and then finally german.

Step-11) Schematic Block Printing

- o) Place the cursor on the top of the architecture you want a printout of.
Select block_print from the ToolBar EDA Tools list.
- o) The current directory will now contain print.txt and print.pdf suitable for printing. Note: Adobe-Acrobat likes to cache pdf files so you often need to quit and restart Acrobat if you re-gen print.pdf.

Step-12) RTL Viewer

- o) Place cursor on component "top" and select tool cv_rtl_viewer.
- o) Cursor up,down left and right thru the top.vhd file.
- o) Page-Up and Page-Down using Ctrl-U and Ctrl-D.
- o) Position cursor on the line:
"v component metrics"
- o) Press <space> to expand the component declaration for metrics.
- o) Repeat the same for foo_proc, u_metrics, etc.

Step-13) Admin Control of Checkins

- o) Place cursor on the top of the hierarchy you want to make ReadOnly.
Toggle expand/collapse so that all files are visible which you wish

to chmod. Scroll to admin tools on the tool bar and select ReadOnly.

- o) To allow a user to checkin a file, place cursor on the mainbranch version of that file, collapse the view so that his children are not visible. Scroll to admin tools on the tool bar and select ReadWrite. The user may now checkin his file over the mainline file. After the checkin, you'll want to set the permission back to ReadOnly.

Step-14) Netlist Viewing

- o) Start ChipVault as before but in a directory with only your netlist.
/tutorial/netlist/% cv.pl top.vhd
- o) Select a decent filter value of say 10 or 20 so that you won't have a hierarchy view full of ANDs and Flops.
- o) You should now have a hierarchy view of all the large netlist blocks.

TheEnd. See the on-line Help FAQ for more info.

(end)

About the Author:

iv) Author Info:

The Author's name is Kevin Hubbard. He is an ASIC designer living in Issaquah, Washington, USA and may be contacted at khubbard@users.sourceforge.net

Please drop the author a brief Email after you've tried ChipVault and let him know if you intend to use it. If nobody uses ChipVault, maintenance will be stopped.

Email with contributions, suggestions, requests for new features, etc is also welcome. The author is also compiling a world map of ChipVault users. Please send an Email indicating the country and city of your location.

Contributions : Please support the open-source community by

- 1) Using OpenSource software. Linux/GNU software is more reliable and higher performance than all other OSs I've used. Try it. Even Microsoft has good things to say about Linux these days. <http://www.opensource.org/halloween/halloween7.php>
- 2) Contributing OpenSource software. Distributing OpenSource software is incredibly easy. Check out www.sourceforge.net for details.
- 3) Buying books of open-source developers so that they may see ~some~ financial rewards for their efforts.

Recommended Book Reading List:

Non-Fiction:

Just for Fun: The Story of an Accidental Revolutionary

by Linus Torvalds, David Diamond

ISBN: 0066620724

<http://www.amazon.com/exec/obidos/search-handle-form/103-0109863-7405479>

(how the Linux Revolution came to being)

The Cathedral and the Bazaar:

Musings on Linux and Open Source by an Accidental Revolutionary

by Eric S. Raymond, Bob Young

ISBN: 0596001088

<http://www.amazon.com/exec/obidos/search-handle-form/103-0109863-7405479>

(why quality of open-source software is superior to closed proprietary)

The Future of Ideas: The Fate of the Commons in a Connected World

by Lawrence Lessig

ISBN: 0375726446

http://www.amazon.com/exec/obidos/ASIN/0375726446/qid=1030395035/sr=2-2/ref=sr_2_2/103-0

(how large corporate titans have influenced Copyright and Patent laws to guarantee their survival while stifling outside innovation)

Fiction:

Everything by Philip K. Dick

Software License:

v) License:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law:

that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections

1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other

circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it

free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>
```

```
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
