APPENDIX B AMCC VECTOR RULES CHECKER (AMCCVRC) (809)

INTRODUCTION TO AMCCVRC

AMCCVRC (VECTOR RULES CHECKER) is an AMCC tool designed to screen functional and AC test vectors for violations of the rules stated in Volume II, Section 4, Vector Submission Rules and Guidelines. Not all rules are checked by the AMCCVRC software. Some screening of the optional parametric vector set may also be performed.

All sampled simulation output files for functional, AC test and parametric vectors (worst-case maximum simulations only) must be successfully processed through AMCCVRC prior to submission. [1]

All AMCCVRC-detected errors must be resolved before submission or a waiver obtained from AMCC for each remaining error and included with the submission.

The vector files will be rerun through AMCCVRC at AMCC as a further verification. Incorrect vectors will be returned to the designer for correction.

SCRATCH FILES

AMCCVRC currently creates three scratch files, ASFLSNL.DAT, ASFLTIME.DAT and ASFLTSTFMT.DAT. These will be deleted by the AMCC shell scripts on the workstations after the AMCCVRC execution is complete. If the shell scripts are not used, these files may be manually deleted after the AMCCVRC execution is complete.

[1] Parametric tests are not complete as of this release. Refer to Section 4 for checking that needs to be done for AC tests and for parametric vector sets.

CHECKS

There are seven checks the AMCCVRC program performs:

- vector race conditions,
- simultaneously switching outputs in a vector,
- unknown signals,
- required signals,
- non-toggled signals,
- differential input,
- vector length check.

The checks can be run all at once or separately. Submit a report that runs all checks on a file in one pass.

CHECKS BY SIMULATION TYPE: *

	RACE	SSO	UKN	TOGGL	DIFF	LENGTH
FUNCTIONAL, CLOCKED	X	X	X	X	X	X
FUNCTIONAL, NON-CLOCKED	-	X	X	X	X	X
AC TEST, CLOCKED	X	X	X	**	X	X
AC TEST, NON-CLOCKED	-	X	X	**	X	X
PARAMETRIC, EITHER	-	X	X	***	X	X

* SAMPLED, MAXIMUM WORST-CASE AMCCSIMFMT OUTPUT FILES ARE THE ONLY SIMULATION FILES PROCESSED BY AMCCVRC

** ONLY SIGNALS INVOLVED IN THE AC TEST WILL CHANGE AND THEN ONLY IN ONE DIRECTION, THEREFORE NO TOGGLE TEST IS REQUIRED.

*** EACH INPUT MUST TOGGLE;
ONE OUTPUT PER VECTOR MUST TOGGLE (THE GATE OUTPUT) FOR THE GATE TREE VERSION; REFER TO SECTION 4.

CALLING AMCCVRC

The signal analysis file is generated by the user and is required when the vector race checks are done (i.e., for any clocked circuit). Create the signal analysis file prior to calling AMCCVRC.

AMCCVRC is called differently on each EWS.

On DAISY, move to the top of the design tree and type:
AMCCVRC

On MENTOR, move to the top of the design tree and type:
/USER/AMCC/COM/AMCCVRC

On VALID, move to the top of the design tree and type:
/uX/amcc/com/run_amccvrc

On a VAX/VMS, consult the systems operator.
(LASAR 6 users - TBS)

On the AMCC VAX/VMS, set up the log-in file to allow:
AMCCVRC
(Consult with the assigned Implementations Engineer to do this.)

Refer to Volume II, Section 7 for the specific instructions for the system to be used.

On all EWS, AMCCVRC invocation exists as a menu option in the run_amcc shell script.

INPUT FILES

Input files to the AMCCVRC program are:

- The AMCCSIMFMT output file
  (only an AMCCSMIFMT file can be processed)
- CIRCUIT.SDI (AMCC AGIF netlist)
- the signal analysis file (clocked circuits only),

Thermal diodes, AC monitors and VBxx macros must not appear in the output file format of the simulation files.

AMCCVRC.LST

AMCC AMCCVRC software generates an output file called AMCCVRC.LST. The output file contains a listing of any errors for each of the checks performed on the vectors.

AMCCVRC REPORT SUMMARY

AMCCVRC produces a report summary at the end of AMCCVRC.LST if the program is exited via the menu. The summary is not generated on an abort, although the individual check reports will be available in the file. Warnings are not reported.

SIGNAL ANALYSIS FILE FORMAT

The signal analysis file is used as input by the race check option of AMCCVRC. It describes which signals are to be checked for race conditions in a vector set and provides a list of clock signals to be checked.

Associated with each clock entry is the clock transition against which the signals are to be checked: rising edge, 0 to 1; falling edge, 1 to 0; or both; and a list of data signals to be checked against that edge for that clock.

The file is composed of one or more lines. Each line may be either a clock definition line or a continuation of a clock definition.

```
<clock_name> <clk_transition> <data_name>
       [<data_transition>] [<data_name>
              [<data_transiton>]]...
```

A clock definition line starts with a clock name (in upper case letters).

The next field is the signal transition for the clock:

(1) "+" = check signal for race condition any time the signal goes from a 0 to 1 in the simulation.

```
EXTCLKA + EXCEPT!
```

(2) "-" = check signal for race condition any time the signal goes from a 1 to 0 in the simulation.

```
EXTCLKB - EXCEPT!
```

(3) "*" or " " = check signal for race condition any time the signal changes.

```
EXTCLKC * EXCEPT!
EXTCLKD   EXCEPT!
EXTCLKE   DAT1  DAT2
```

The rest of the line defines which data signals are to be checked against the clock signal transition(s). Each data signal may also have a signal transition. If no transition is specified for the data signal, then both transitions of the data signal are checked against the specified clock signal transition(s).

The data definition has two forms;

- a simple list of data signals to be checked,

```
EXTCLK + DAT0 DAT1 DAT2 DAT3 DAT4 DAT5 DAT6
&        DAT7 DAT8 DAT9 DAT10 DAT11 DAT12
&        DAT13 DAT14 DAT15 SELCT0 SELCT1 +
&        SELCT2 + SELCT3 + EXTRST -
```

- or the keyword "EXCEPT!" and a list of data signals to be excluded from the check.

```
EXTCLK + EXCEPT! EXTRST -
```

- the EXCEPT! may be followed by nothing to indicate that there are no exceptions.

```
EXTCLK + EXCEPT!
```

There is no punctuation between fields, the space is the delimiter.

Comment lines are not currently permitted.

Continuation lines begin with an "&" in card column 1. The continuation line contains only data signal names.

A data signal name may reference a bus. A bus reference is of the form: "<bus_name>(<start>,<stop>)". This program assumes that the name of the signals in the bus are of the form: <bus_name>0, <bus_name>1, <bus_name>2, ... <bus_name>n. Note there are no leading zeros in the number appended to the bus name. i.e.: ADR(0,15)

Note:
(1) Bus definitions MUST be the only data signal definition on a line.
(2) There is a limit of 32 signals in a bus at this time.
(3) The clock signal is automatically excepted when the "EXCEPT!" clause is used.
(4) Transition specifications may be applied to the "EXCEPT!" clause. This will cause the transition to be set on all the signals that are not excepted by this specification.

Defintions:
(1) Clock - The clock signal to check.
(2) Data signal - signal to check for a change when the clock goes through the specified transition.
(3) Bus - A group of signals that have the same base name with only a number suffix varying.

SIGNAL ANALYSIS FILE EXAMPLE 1:

```
CLOCK + DATA(0,15) +
&       LRESET STOP LBRESET
&       ADDR(0,15)
CLOCK - DATA(16,31)
CLOCK - EXCEPT! + LRESETN BRESET
```

In this example,

(a) DATA0 through DATA15 are checked for any switching anytime CLOCK goes from a 0 to a 1 (rising edge) and DATA0 through DATA15 themselves change from 0 to 1 (rising edge). A falling edge on DATA0 through DATA15 can occur when the CLOCK signal is rising.
(b) LRESET, STOP, LBRESET are checked for any switching anytime CLOCK goes from a 0 to a 1.
(c) ADDR0 through ADDR15 are checked for any switching anytime CLOCK goes from a 0 to a 1.
(d) DATA16 through DATA31 are checked for any switching anytime CLOCK goes from a 1 to a 0 (falling edge).
(e) All data signal transitions from 0 to 1, except LRESTN and BRESET, are checked for switching anytime CLOCK goes from a 1 to a 0 (falling edge). The CLOCK signal is not checked against itself.

SIGNAL ANALYSIS FILE EXAMPLE 2:

```
EXTCLK + EXCEPT! EXTCLKN
```

where EXTCLKN is the differential pair for the clock input EXTCLK. The file indicates that there is one clock (EXTCLK), it is differential so its pair is excluded from the check, and all signals are checked against the rising edge to be certain that they do not vary in any direction on the active edge of the clock.

SIGNAL ANALYSIS FILE EXAMPLE 3:

```
EXTCLK + EXCEPT!
```

In this example, all signals are checked against the rising edge of the clock EXTCLK.

SIGNAL ANALYSIS FILE EXAMPLE 4:

```
EXTCLK + EXCEPT! D1 D2 D3 D4
EXTCLK - EXCEPT! D4 D5 D6 D7
```

All signals except D1, D2, D3, D4 are checked against the rising edge of the clock EXTCLK while all signals except D4, D5, D6, D7 are checked against the falling edge of the clock EXTCLK.

ERROR MESSAGES CONCERNING THE SIGNAL ANALYSIS FILE

ERROR936: SIGNAL IN SIGNAL ANALYSIS FILE NOT IN SDI NETLIST

There is a mismatch between the CIRCUIT.SDI file and the signal analysis file, probably due to a misspelling in the signal analysis file.

ERROR937: ILLEGAL SIGNAL TRANSITION IN SIGNAL ANALYSIS FILE

A transition other than 1-0 or 0-1 has occurred.

ERROR938: CLOCK REFERENCED IN SIGNAL ANALYSIS FILE NOT FOUND

Check the spelling in the signal analysis file and check that the correct CIRCUIT.SDI file has been supplied.

ERROR939: ILLEGAL CLOCK TRANSITION IN SIGNAL ANALYSIS FILE

Improper definition of clock in signal analysis file.

When the program is invoked it will begin prompting for files and checks to perform as follows:

```
Enter the SDI file name (input) >
```

This requests the filename of the SDI netlist. The netlist file is called CIRCUIT.SDI and located in the /ERC subdirectory. If this file is renamed, be careful to list the correct filename.

```
Enter the AMCCSIM file name (input) >
```

This requests as input to the AMCCVRC program the user-defined name of the AMCCSIMFMT output file to be processed. Be certain that the file is a sampled file.

```
What type of check do you want to do?
   0) Quit
   1) Race conditions
   2) Simultaneously Switching Outputs
   3) Unknown signals
   4) Signal Toggle check
   5) Required Signal Check
   6) Differential Input Check
   7) Vector Length Check
   9) All but Race (Tests 2 to 7)
  10) All the above (Tests 1 to 7)
  11) Change error limit
 Enter choice (0..11) >
```

This is the AMCCVRC main menu. AMCCVRC will always return to this menu after a selection is completed.

Selecting "0" will generate the AMCCVRC.LST file and the program will exit.

Selecting any of "1" - "7", will cause AMCCVRC to perform one of the seven checks on the vectors before returning to the main menu. If option "10" is selected, all seven checks will be done. Use "10" for clocked circuits. If option "9" is selected, all checks except the race check will be done. Use "9" for non-clocked circuits.

Selecting "11" is used to change the default error limit of 100.

```
Enter the signal analysis file (input)>
```

This requests the filename of the signal analysis file. The program prompts for this only when option "1" or "10" has been selected from the menu. This file is only required when vector race checking is performed. Vector race checking is required for any clocked circuit. The file must exist before AMCCVRC is invoked.

DESCRIPTION OF THE VECTOR CHECKS

RACE CONDITION CHECKER (menu option 1)

CLOCKED CIRCUITS

A signal analysis file must be created before running the race conditon checker. The format for the signal analysis file has been described.

The race condition checker verifes that no external races exist in the functional vectors, provided that the signal analysis file has been accurately specified before running the race condition checker.

The signal analysis file specifies which data signals should not change on the active edge of each clock, and AMCCVRC will check the vectors and report (in AMCCVRC.LST) any specified data signals that do change on the active edge of a clock. Multiple clock systems may have data related to one clock switching when a different, unrelated clock switches. The user must carefully describe these conditions to the software via the signal analysis file.

Sometimes it is difficult to specify a race condition with the pad signals available in the functional vectors. Occassionally, complex clocking or complex data logic (i.e. muxs) can make it hard to determine if simultaneous clock and data changes on the pads are actually causing a race.

One approach to this problem is to check the pads that might cause a race and verify in your vectors that the errors flagged in AMCCVRC.LST are actually not races. If the number of false errors is too large, races can be checked for by specifying internal signals.

A separate simulation can be run using the same input stimulus as your functional vectors but having internal signals listed for easier checking of externally caused vector races. AMCCVRC will allow internal signals in the vector and signal analysis files but will generate an error message for each internal signal:

ERROR905: INTERNAL SIGNAL REFERENCED IN SIMULATION
Signal name = <signal>

If an internal signal having a default name is specified, then the SDI netlist-defined name for the signal must be used. This can be found in the AMCCXREF.LST file that was produced during the AGIF-ERC execution phase.

Note: AMCC does not want simulations containing internal signals (other than the bidirectional and 3-state enable controls) to be submitted. They are for your own use during debug. The required signal checker will flag these internal signals. If internal signals are used in vectors for race checking, they MUST BE REMOVED before the vectors are submitted.

Vector races are reported in AMCCVRC.LST as follows:

Clock <clock signal> had <nnn> Errors

ERROR930: RACE CONDITION ON RISING EDGE AT TIME <ttt>
signals switching = <data signal list>

A data signal has been detected as switching when the clock switches 0-1. Verify that the signal analysis file definition is correct. If it is correct, alter the vector set to delete the race.

ERROR931: RACE CONDITION ON FALLING EDGE AT TIME <ttt>
signals switching = <data signal list>

A data signal has been detected as switching when the clock switches 1-0. Verify that the signal analysis file definition is correct. If it is correct, alter the vector set to delete the race.

ERROR932: RACE CONDITION ON EITHER EDGE AT TIME <ttt>
signals switching = <data signal list>

A data signal has been detected as switching when the clock switches 0-1 or 1-0. Verify that the signal analysis file definition is correct. If it is correct, alter the vector set to delete the race.

ERROR933: BAD BUS SPECIFICATIONS - ILLEGAL START VALUE
start parameter for data signal .... on line <nn>

Non-parsable data supplied as the start value.

ERROR934: BAD BUS SPECIFICATIONS - ILLEGAL END VALUE
end parameter for data signal .... on line <nn>

Non-parsable data supplied as the end value.

ERROR935: BAD BUS SPECIFICATIONS - RANGE (END-STOP) > 32
max range is 32 on line <nn>

There is a limit of 32 on the bus size range for any single data line entry.

SIMULTANEOUSLY SWITCHING OUTPUT CHECKER
Simultaneously Switching Vector Outputs
(menu option 2)

The number of outputs that are simultaneously changing state in a single vector may not exceed the limit for the I/O mode of the array, regardless of the simulation type. Outputs switching in a vector cause noise for the tester. (This limit is independent of the added power and grounds for simultaneously switching outputs in an array quadrant.)

All heavy (more than 8 or 16 outputs) switching vectors must be placed at the end (last third) of the vector set. Whenever possible, simultaneously switching outputs in a vector should be kept at or below 8 or 16 (array and I/O mode dependent). Refer to the table below.

NUMBER OF OUTPUTS THAT MAY SWITCH IN A VECTOR:

I/O MODE:	# TTL	# ECL
100% TTL CIRCUIT	16	0
100% ECL CIRCUIT	0	16
MIXED ECL/TTL	8	8
MIXED +5V ECL/TTL	8	8

If the simultaneous switching limit is exceeded on a vector step within the last 1/3rd of the vector file, a WARNING will be reported in AMCCVRC.LST.

Circuits submitted with heavy switching in excess of the stated limits will have those vectors sets truncated during wafer-sort, reducing the level of confidance that a part of good below the fault-grade score.

AMCC strongly recommends that the vectors be written to remove these warnings from the SSO check.

In a +5V REF ECL/TTL or STD-REF ECL/TTL circuit, when either ECL or TTL are not switching, the limit on the other type (TTL or ECL) may NOT be raised from 8 to 16. In a 100% TTL or 100% ECL circuit, the ground bus within the I/O ring is double-wide compared to the mixed type circuit, allowing more outputs to switch.

The other consideration is the tester, and the limits used in the SSO check are tester-driven.

3-STATE ENABLE AND BIDIRECTIONAL ENABLE

AMCC recommends that the enable structure be designed to allow no more than 8 or 16 enables to change state during any one test vector. (Follow the limits in the above table.) This is a testing and not an operational limit. This degating will allow full functional testing to be run for 3-state and bidirectional circuits for wafer sort and packaged parts.

RESET, SET

Document clearly which SSO error and warning messages which relate to a master RESET or SET. RESET and SET by their definition will occur throughout the vector set.

ERROR915: TOO MANY SIMULTANEOUSLY SWITCHING OUTPUTS AT TIME <tt>
<nn> SIGNAL SWITCHING = <list>

If the simultaneous switching limit is exceeded on a vector step within the first 2/3rds of the vector file, an ERROR will be reported in AMCCVRC.LST.

WARN916: TOO MANY SIMULTANEOUSLY SWITCHING OUTPUTS AT TIME <tt>
<nn> SIGNAL SWITCHING = <list>

If the simultaneous switching limit is exceeded on a vector step within the last 1/3rd of the vector file, a WARNING will be reported in AMCCVRC.LST.

Rearrange the vector set to place heavy switching at the end of the vector set or break up the vectors so heavy switching does not occur. The latter is the recommended approach

UNKNOWN SIGNAL STATE CHECKER (menu option 3)

Unknown signal states are restricted in the vector set. AMCC prefers that initialization take 25 steps or less. [1] This check verifies inputs and outputs follow vector submission rules for unknown (X) signal states.

o Inputs are flagged if they are unknown at any time in the vector set, including bidirectional macros in the input mode.

o Outputs are flagged if they are unknown after 25 vector steps, including bidirectional macros in the output mode.

If an input or output is flagged, it will be reported in AMCCVRC.LST as follows:

ERROR920: UNKNOWN SIGNAL STATE AT TIME <tt>
signals = <list>

Inputs must always be defined as 0 or 1. Change the vectors to properly initialize the inputs. Change the vectors to allow a faster initialization of the outputs if possible. This error, when for an output past the first 25 vectors, can be waived by AMCC.

[1] A later release will raise this to 100 vectors for the large arrays.

SIGNAL TOGGLE CHECKER (menu option 4)

All inputs and all outputs must toggle from 1 to 0 and from 0 to 1 least once in the vector set (ERROR945, ERROR946, ERROR947, ERROR948). Internal 3-state control signals must toggle in both directions in the vectors (ERROR950). Bidirectional signals should change mode (ERROR949).

When parametric vectors using a gate tree are supplied with a circuit, the parametric vectors themselves "cover" the gate tree for functional testing. The parametric vector set may be concatenated to the functional set for fault grading. Vectors may need to be added to the functional set to "toggle" the gate tree output and suppress any AMCCVRC toggle test error message.

When a signal does not toggle in both directions, it will be reported (depending on type of signal) in AMCCVRC.LST as follows:

ERROR945: INPUT SIGNAL DOES NOT TOGGLE

A primary input has not been toggled 0-1 and 1-0 within the vector set. Add a vector or vectors to the set to eliminate the error.

ERROR946: OUTPUT SIGNAL DOES NOT TOGGLE

A primary output has not been toggled 0-1 and 1-0 within the vector set. Add a vector or vectors to the set to eliminate the error.

ERROR947: BIDIRECTIONAL SIGNAL DOES NOT TOGGLE
IN INPUT MODE

A bidirectional macro has not been toggled 0-1 and 1-0 when in the input mode within the vector set. Add a vector or vectors to the set to eliminate the error.

ERROR948: BIDIRECTIONAL SIGNAL DOES NOT TOGGLE
IN OUTPUT MODE

A bidirectional macro has not been toggled 0-1 and 1-0 when in the output mode within the vector set. Add a vector or vectors to the set to eliminate the error.

ERROR949: BIDIRECTIONAL SIGNAL DOES NOT CHANGE MODE

A bidirectional macro has not changed mode input-output and output-input within the vector set. Add a vector or vectors to the set to eliminate the error.

ERROR950: INTERNAL SIGNAL DOES NOT TOGGLE

An internal 3-state control signal (3-state or bidirectional enable) has not been toggled 0-1 and 1-0 within the vector set. Add a vector or vectors to the set to eliminate the error.

All error messages identify the component, the macro, the signal and the schematic page where the signal can be found.

REQUIRED SIGNAL CHECKER (menu option 5)

All required signals must be present in each vector. This includes all primary inputs, all primary outputs, all primary bidirectionals and all 3-state and bidirectional enable signals (these are internal signals) in that order. No other internal signals are allowed. Exclude thermal diodes, AC monitors and VBxx macro signals.

If a required signal is found to be missing or if an extraneous signal is found to be present, it will be reported to AMCCVRC.LST as follows:

ERROR900: INTERNAL SIGNAL REFERENCED IN SIMULATION NOT IN NETLIST

The wrong netlist has been supplied or the wrong simulation file has been supplied.

ERROR960: EXTRANEOUS SIGNAL IN THE SIMULATION

Internal signals that are not 3-state enable or bidirectional enable signals have been listed in the simulation format. These internal signals must be deleted prior to submission. Correct the format and re-execute the simulation.

ERROR961: AN INPUT SIGNAL IS NOT IN THE SIMULATION

One of the primary input signals has been left out of the simulation. Correct the format and re-execute the simulation.

ERROR962: AN OUTPUT SIGNAL IS NOT IN THE SIMULATION

One of the primary output signals has been left out of the simulation. Correct the format and re-execute the simulation.

ERROR963: A BIDIRECTIONAL SIGNAL IS NOT IN THE SIMULATION

One of the primary bidirectional signals has been left out of the simulation. Correct the format and re-execute the simulation.

ERROR964: A TRISTATE/BIDIR CONTROL SIGNAL IS NOT IN THE SIMULATION

The enable signal for a 3-state TTL output or for a bidirectional macro (TTL or ECL) has been left out of the simulation. Correct the format and re-execute the simulation.

All error messages identify the driving component, the macro, the pin, the signal and the schematic page.

DIFFERENTIAL INPUT CHECKER (menu option 6)

This check verifies that differential inputs are properly driven by the simulation test vectors (are always in opposite states).

ERROR970: THE FOLLOWING SIGNAL PAIRS ARE NOT DIFFERENTIAL AT TIME <ttt>
<list>

For information, the module will list the differential pairs found:

The following signals are differential:
<list>

VECTOR LENGTH CHECK (menu option 7) [2]

This check will check that the array has 120 I/O pads or less and if so will then check for vector length over 4K. The program will compute the number of vectors that must be added to accommodate the SENTRY TESTER.

The number of vectors added to the simulation file is based on:

1) add 2 if a bidirectional I/O changes from an input to an output
2) add 1 if an output or a bidirectional I/O signal changes from unknow to known or from known to unknown.
3) add 1 if an output or a bidirectional I/O signal changes from Hi-Z to a known state or from a known state to a Hi-Z state.

The length check is made on the test program length, not to exceed 4096.

ERROR975: TOO MANY VECTORS IN SIMULATION

For information, the module will list the number of vectors in the submitted simulation output file, the number of vectors that will be in the test program.

[2] This check will be expanded in the next release.

```
AMCCVRC VERSION  3.20
```

• Note: The versions in individual reports may vary from the final release numbers on your reports since this run was made on an early version of the new AMCCVRC software.

```
 Enter the SDI file name (input) >  CIRCUIT.SDI
 Enter the AMCCSIM file name (input) > FUNCTION32.VEC


   What type of check do you want to do?
     0) Quit
     1) Race conditions
     2) Simultaneously Switching Outputs
     3) Unknown signals
     4) Signal Toggle check
     5) Required Signal check
     6) Differential Input check
     7) Vector Length Check
     9) All but Race (Tests 2 to 7)
    10) All the above (Tests 1 to 7)
    11) Change error limit
   Enter choice (0..11) >10
 Enter the signal analysis file (input)>SIGNAL.DAT
```

• CIRCUIT.SDI is the default name for the AGIF netlist.

• FUNCTION32.VEC is the user-defined name for the sampled functional vector simulation results file formatted by AMCCSIMFMT.

• The choice of "10" is typical for a clocked circuit; use "9" for a non-clocked circuit.

• SIGNAL.DAT is the user-defined name of the signal analysis file for this circuit.

The program produces the following trace of its execution. (PARTIAL)

```
>>> Creating working files

---------- Race CHECK ------------

>>> phase 1 - phase 2 - phase 3 .........

---------- SSO CHECK ------------

>>> phase 1 - phase 2 - phase 3 - phase 4

---------- Unknown signal CHECK ------------

>>> phase 1 - phase 2 - phase 3 - phase 4
```

• • •

```
  What type of check do you want to do?
    0) Quit
    1) Race conditions
    2) Simultaneously Switching Outputs
    3) Unknown signals
    4) Signal Toggle check
    5) Required Signal check
    6) Differential Input check
    7) Vector Length Check
    9) All but Race (Tests 2 to 7)
   10) All the above (Tests 1 to 7)
   11) Change error limit
  Enter choice (0..11) >0
```

If all tests are finished then select "0", else select the tests to be performed.

The AMCCVRC.LST file is formatted as follows:

```
*******************************************************
*          Race Condition Checker                     *
*          VERSION  2.50                              *
*******************************************************


   Path Name /USER/CLASS/Q3500REG32
   Product Name Q5REG32
   Circuit family Q5000
   Circuit technology M
   Date 16 OCT 1988
   Time 12:35
   Product Grade MIL


   No errors found.
```

This indicates that, for the conditions described in the signal analysis file, no data signal changes on the active edge of its clock.

```
*******************************************************
*          Simultaneously Switching Output Checker    *
*          VERSION  2.50                              *
*******************************************************


   Path Name /USER/CLASS/Q3500REG32
   Product Name Q5REG32
   Circuit family Q5000
   Circuit technology M
   Date 16 OCT 1988
   Time 12:35
   Product Grade MIL
```

```
ERROR915: TOO MANY SIMULTANEOUSLY SWITCHING OUTPUTS AT TIME 19999
     64 switching outputs= DO00 DO01 DO02 DO03 DO04
                           DO05 DO06 DO07 DO08 DO09
                           DO10 DO11 DO12 DO13 DO14
                           DO15 DO16 DO17 DO18 DO19
                           DO20 DO21 DO22 DO23 DO24
                           DO25 DO26 DO27 DO28 DO29
                           DO30 DO31 TDO00 TDO01 TDO02
                           TDO03 TDO04 TDO05 TDO06 TDO07
                           TDO08 TDO09 TDO10 TDO11 TDO12
                           TDO13 TDO14 TDO15 TDO16 TDO17
                           TDO18 TDO19 TDO20 TDO21 TDO22
                           TDO23 TDO24 TDO25 TDO26 TDO27
                           TDO28 TDO29 TDO30 TDO31
```

This error refers to a RESET execution in the first 2/3rds of the vector set. Document what this error is when it occurs. If this error is generated and it is not due to a SET or RESET or to a 3-state enable switch, it is an unacceptable error. Consult AMCC.

The number of outputs that may switch in a vector is not the same restriction as the number of outputs that may switch in the quadrant of an array and the two limits should not be confused. This test is for vector behavior on the tester. For 100% TTL or 100% ECL circuits, 16 outputs may switch in a vector independent and regardless of the number of added power and ground pads. In a mixed ECL/TTL circuit, up to 8 TTL and up to 8 ECL may switch in one vector. These limits are independent and may not be combined.

```
WARNING916: TOO MANY SIMULTANEOUSLY SWITCHING OUTPUTS AT TIME 469999
      32 switching outputs= TDO00  TDO01  TDO02  TDO03  TDO04
                            TDO05  TDO06  TDO07  TDO08  TDO09
                            TDO10  TDO11  TDO12  TDO13  TDO14
                            TDO15  TDO16  TDO17  TDO18  TDO19
                            TDO20  TDO21  TDO22  TDO23  TDO24
                            TDO25  TDO26  TDO27  TDO28  TDO29
                            TDO30  TDO31
WARNING916: TOO MANY SIMULTANEOUSLY SWITCHING OUTPUTS AT TIME 489999
      32 switching outputs= TDO00  TDO01  TDO02  TDO03  TDO04
                            TDO05  TDO06  TDO07  TDO08  TDO09
                            TDO10  TDO11  TDO12  TDO13  TDO14
                            TDO15  TDO16  TDO17  TDO18  TDO19
                            TDO20  TDO21  TDO22  TDO23  TDO24
                            TDO25  TDO26  TDO27  TDO28  TDO29
                            TDO30  TDO31

   Number of errors found = 1
```

These warnings are due to 3-state enable switching during the last 1/3rd of the vector set. Document this warning. If the warning is received and is not due to a SET, RESET or 3-state enable switch, review the vector set and eliminate the warning. Heavy switching should be kept in the last 1/3rd of the vectors. It does cause problems with the testers and should only be present when mandatory for the test.

```
*****************************************************
*          Unknown Signal State Checker             *
*          VERSION  2.50                            *
*****************************************************


   Path Name /USER/CLASS/Q3500REG32
   Product Name Q5REG32
   Circuit family Q5000
   Circuit technology M
   Date 16 OCT 1988
   Time 12:35
   Product Grade MIL


   No errors found.
```

```
**************************************************
*          Signal Toggle Check                   *
*          VERSION  2.50                         *
**************************************************

   Path Name /USER/CLASS/Q3500REG32
   Product Name Q5REG32
   Circuit family Q5000
   Circuit technology M
   Date 16 OCT 1988
   Time 12:35
   Product Grade MIL

   No errors found.

**************************************************
*          REQUIRED SIGNAL CHECK                 *
*          VERSION  2.50                         *
**************************************************

   Path Name /USER/CLASS/Q3500REG32
   Product Name Q5REG32
   Circuit family Q5000
   Circuit technology M
   Date 16 OCT 1988
   Time 12:35
   Product Grade MIL

   No errors found.
```

```
*******************************************************
*          Differential Input Checker                 *
*          VERSION  2.50                              *
*******************************************************


   Path Name /USER/CLASS/Q3500REG32
   Product Name Q5REG32
   Circuit family Q5000
   Circuit technology M
   Date 16 OCT 1988
   Time 12:35
   Product Grade MIL


 The following signals are differential:
          EXTCAN  , EXTCKA


   No errors found.


*******************************************************
*          Vector Length Check                        *
*          VERSION  3.00                              *
*******************************************************


   Path Name /USER/CLASS/Q3500REG32
   Product Name Q5REG32
   Circuit family Q5000
   Circuit technology M
   Date 16 OCT 1988
   Time 12:35
   Product Grade MIL


Number of simulation vectors=104
Number of Test vectors=106

   No errors found.
```

• The messages on vector length and vector count are currently only printed if the number of pads available (not the number of pads used) is 120 or less. If greater than 120, these messages are suppressed.

```
**************************************************
*          VRC report summary                    *
*          VERSION  2.20                         *
**************************************************

   Path Name /USER/CLASS/Q3500REG32
   Product Name
   Circuit family
   Circuit technology
   Date 16 OCT 1988
   Time 12:35
   Product Grade


Test number  1  Had   0  Errors
Test number  2  Had   1  Errors
Test number  3  Had   0  Errors
Test number  4  Had   0  Errors
Test number  5  Had   0  Errors
Test number  6  Had   0  Errors
Test number  7  Had   0  Errors
```

This summary identifies problem areas. Except for test 2, which will report errors for SET, RESET and 3-state enables, all tests should show zero errors.

Note: If on exiting the AMCCVRC program a proper exit is not performed (by entering "0" (zero) when asked for a choice), the report summary will not be written.