1. How did you compile spi\_test\_lib\_pkg? Packages must be compiled before they can be imported.
2. For cadence, incdir is prefixed by “-“ , for vcs & modelsim it’s prefixed by “+”
3. Interface is compiled separately instead of adding it to a package since it’s not allowed to be a part of a package or a module
4. Always include “incdir” before each and every directory, both in cadence and in vcs,questasim
5. Find out the stuff to be randomized and include it in seq\_item under uvm\_object\_utils\_begin/end
6. Mistake made :: instead of using uvm\_sequence\_item, used uvm\_object !!!!
7. Forgot how to create random items under uvm\_object\_utils\_begin/end.
   1. Order of creation of sequence\_item is :
      1. Declare the class as txn extends “uvm\_sequence\_item”
      2. Declare the variables under the class before registering them
      3. Register the variables under “uvm\_object\_utils\_begin/end”
      4. Use “uvm\_field\_int” for bits as well as integers. Others max utilized are:
         1. “uvm\_field\_enum”
         2. “uvm\_field\_real”
         3. Uvm\_field\_string
8. Uvm\_sequence\_item doesn’t involve usage of build\_phase or connect\_phase as it is purely a “data” package only
9. Missed out on “uvm\_field\_array\_int” and used “uvm\_field\_int” for usage of dynamic array which is to be randomized for data packet generation
10. Commented out the rest of the packages and instead currently using only Yapp\_pkg & common\_pkg alone for first round of data\_generation
11. Only “new” function is non-virtual, rest of the functions seem to be virtual
12. Mistake made : - instead of build\_phase , added new in its place
13. It is illegal to import a package inside a class. It’s possible to import a package only inside a module
14. Can’t include an “interface” within a package as “Interface” is a design element
15. Task & functions don’t need “begin” statement when they’re used in SV unlike Verilog
16. Warning : “Treating stand-alone use of function 'randomize' as an implicit VOID cast.”
    1. Ans : Randomize function returns a value of ‘1’ if successful and ‘0’ if unsuccessful. You should always check the return value.

Tool based debugs:

Questasim:

1. Didn’t create vlib work first under sim directory
2. Didn’t map the work directory using “vmap” command, which copies modelsim.ini from elsewhere(Tool’s home) to the work directory
3. Didn’t know to use “-l <file\_name>” for compile log
4. Error: ../agents/yapp/yapp\_seq\_item.sv(2): near "uvm\_sequence\_item": syntax error, unexpected IDENTIFIER
   1. Your pakmx\_transaction class should be inside a package that imports uvm\_pkg.
   2. Ans : Imported uvm\_pkg::\* ; under the package to avoid this error
5. \*\* Error: ../agents/yapp/yapp\_seq\_item.sv(9): (vlog-2163) Macro `uvm\_object\_utils\_begin is undefined.
   1. Ans : Imported the `include “uvm\_macros.svh” to avoid this error.
6. \*\* Error: ../agents/yapp/yapp\_seq\_item.sv(21): Unresolved reference to 'build\_phase'
   1. Ans : uvm\_sequence\_item doesn’t involve any build\_phase or connect\_phase
7. \*\* at ../agents/yapp/yapp\_seq\_item.sv(2): Typedef 'yapp\_seq\_item' multiply defined.
   1. Ans : uvm\_sequence\_item declared both in package and in yapp\_env.sv
8. tools/mentor/qsim10.3b/questasim/bin/../linux/vish: error while loading shared libraries: libXft.so.2: cannot open shared object file: No such file or directory
   1. Ans : Forgot to include –L ${PRJ\_HOME}/sim/work as library before doing simulation
   2. Forgot to add work.top as the module to be simulated and instead only used “vsim –L work top” as the vsim command