* **Redundancy Cell** - When a flip flop is driving some critical function then user uses more than one of the flip flop to drive them and made them redundancy cell. Their function is same for all of them.
* **Block root –** Used in case of WB, BB and lib cells. We can restrict that we need to see the structure of leaf cell or not. flattening is not done for the lib cell in our present view for lib cell.

**Running Different Spyglass Version**

* Load spyglass spyglass/2018.09-SP2 – module load

**To generate prd file (out.flat.v)**

* set\_option sflat3789 on

**How to run testpoints setup**

remote/vgatr\_dft\_spyglass/dft\_tmp/atpg\_setup/tp\_qor\_exp

su -l dft-spyglass

4h\*RCGG7

**How to debug license**

spg -LICENSEDEBUG

**IN DFTRules.TOC**

We checkout IF at setup and EF at specified view by T=

**For using FTP –** see file reference/p4\_ref

**To decript a file –**echo <key> | gpg --passphrase-fd 0 --batch --no-tty –yes file.gz.gpg

**For taking machines –**

<https://vgdhelp/article/details/?id=276>

**To run Classic batch**

setenv SPG\_ALLOW\_CLASSIC\_MODE 1

run below cmd

spyglass test.v -policies dft\_dsm -sgdc test.sgdc -rules "Info\_soft\_error\_propagation Info\_diagnostic\_coverage" -gateslib MACROS.lib -batch -enable\_gateslib\_autocompile -diagnostic\_coverage\_report\_probability=on -f sources.f -use\_vcs\_compile -top top

val valgrind.out

**dbus-launch not working with su -l**

xhost+ and then use su -l and then take machine and run console

**to run vcs flow with debug build**

setenv SVI\_DEBUG\_REGRESSION 1

set\_option compat\_opts old\_enc\_supp

**while running vhdl encrypted test case run in 2 pass flow 1- generate then run test.prj**

spg -project dw\_ufe\_step1.prj -batch -designread

**For encrypting file**

setenv PATH /global/apps/syn\_2018.06/linux64/syn/bin:$PATH  
module load syn/2019.03-SP3  
synenc test.v

**Setting up new clients –**

vgp4init -r TD

vgp4init -r VCS2019.06

vgp4sync spyglass

vgp4link <source-dir> <link-dir>

vgbuild -mode64 [--debug-full] –dir=spyglass-src

incremental (spyglass-src/spyglass/dft/dev)

vgbuild -mode64 --debug-full

incremental:

p4sync (alias)

supporting

**To Create source code client**: vgp4init -r TD

**To Sync:**

|  |  |  |
| --- | --- | --- |
| Product | Sync command | Build Command |
| Spyglass | vgp4sync spyglass | vgbuild –mode64 –dir=spyglass-src |
| Vcs+spyglass | vgp4sync vtg spyglass | vgbuild –mode64 |
| vcstatic | vgp4sync vcstatic | vgbuild –mode64 –engineer  –dir=monet |
| Vcstatic+spyglass | vgp4sync vcstatic  spyglass | vgbuild –mode64 –dir=spyglass-src  vgbuild –mode64 –dir=monet |

**To Update existing VCS clients, please run the following command to get the template updated.**cd <existing client root>p4 client -t TD\_template

**To Create regression client:** vgp4init -r TD --regr

**Nightly Safe Build paths of spyglass:**

Mode64 Opt: /remote/vtgimages/SAFE/linux\_RH6\_EM64T\_TD\_mode64/spyglass-src/sgcommon/kernel/source/SPYGLASS\_HOME  
Mode64 Debug: /remote/vtgimages/SAFE/linux\_RH6\_EM64T\_TD\_mode64\_debug\_Engineer/spyglass-src/sgcommon/kernel/source/SPYGLASS\_HOME

**To shelve and unshelve**To shelve the changes(go to your TD.SPYGLASS client where you have changes): p4 shelve To unshelve the changes at TD client: p4 unshelve -s <sl\_no> -b TD

**Nightly build time:**Nightly builds are based on time stamp 20:00:00 PM PDT. Builds start at 20.30 PM PDT with code synced to 20:00 PM PDT