DC Synthesizer Command Reminder

Here is a generic version of the dc_shell-t *design_name*.sct script which has been provided with many of the labs. Use builtin DC help for more:

```
# dc_shell TSMC TcL startup script for XG mode (jmw 2005-12-18):
# (Many back-end setup commands not shown:  See generic.sct on disc)
# DC variables:
set search_path ". [getenv SYNOPSYS]/../TSMCLibes/tcbn90ghpSYN"
set target_library tcbn90ghptc.db
set link_library   tcbn90ghptc.db
#
set symbol_library tcbn90ghp.sdb
#
# User-defined synthesis output location and compilation commands:
define_design_lib TopModuleName -path ./TopModuleNameSynth
# analyze -work TopModuleName -format verilog SubModuleName.v
analyze   -work TopModuleName -format verilog TopModuleName.v
elaborate -work TopModuleName TopModuleName
#
# chip-related technology:
set_operating_conditions NCCOM
set_wire_load_model -name "TSMC8K_Lowk_Conservative" [all_designs]
#
# Typical technology-related design rules:
set_drive      10.0 [all_inputs]   # drive x load = R x C --> delay time
set_load       30.0 [all_outputs]
set_max_fanout 20   [all_inputs]
set_max_fanout 20   [all_designs]
#
# Typical design-specific constraints:
#create_clock -period 10 ClockPort
#set_max_area   250
#set_max_delay   10 -to [all_outputs]
#set_input_delay  1 -clock ClockName {in1 in2 ...}   # or, [all_inputs]
#set_output_delay 1 -clock ClockName {out1 out2 ...} # or, [all_outputs]
#
# Drop into interactive mode for compile & optimize:
# compile
# compile -map_effort high
# compile -incremental_mapping
# compile -area_effort high
# ungroup -all -flatten
# report_area
# report_timing
# write -hierarchy -format verilog -output verilogNetlist.v
# write -hierarchy -format ddc  #  Don't write .db in XG mode!
```

In the verilog, comment directives can have file scope; in a module, they have module scope:

```
 . . .
//synopsys dc_tcl_script_begin
// set_dont_touch instance_name
// set_max_delay n [all_outputs]
// ...
//synopsys dc_tcl_script_end
 . . .
```