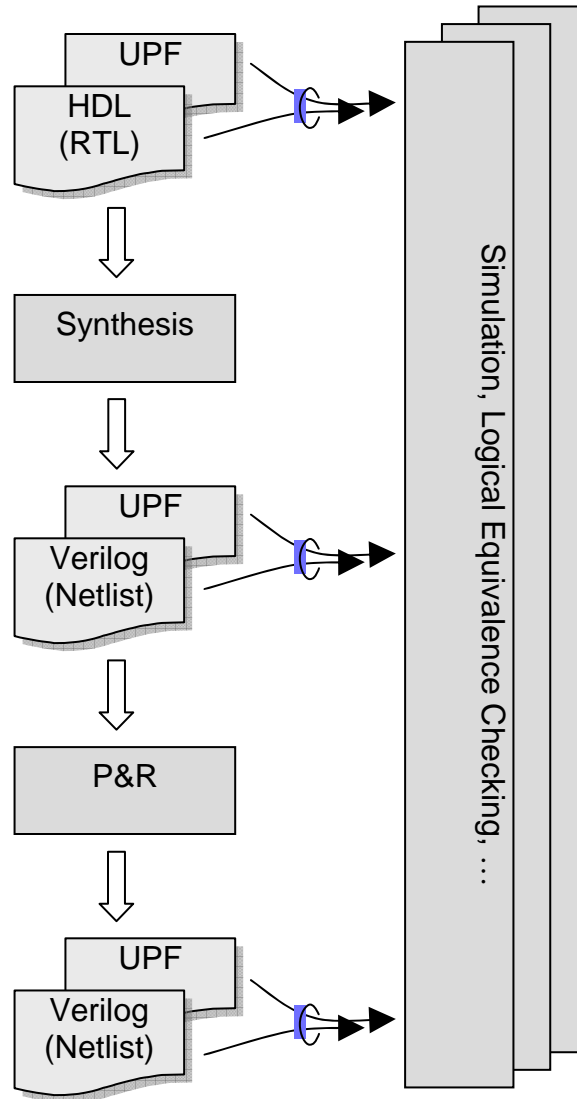


UPF 1.0 Quick Reference Guide



Excerpts from the "Unified Power Format(UPF) Standard, Version 1.0" reproduced by permission of Accellera. Copyright(c) 2006-2007 by Accellera. Accellera does not warrant or represent the accuracy or content of the excerpted material, and expressly disclaims any express or implied warranty. Accellera Standards excerpts are supplied "AS IS." The full standards document in its entirety can be found under www.accellera.org.

Command Syntax	Description and Example
add_domain_elements <i>domain_name</i> -elements <i>list</i>	Add design elements to a power domain add_domain_elements U1/PD1 -elements {U1/U2/foo1 U1/U2/foo2}
add_port_state <i>port_name</i> {-state { <i>name</i> < <i>nom</i> < <i>min nom max</i> > off>}}*	Add state to a port add_port_state VN1 -state {active_state 0.88 0.90 0.92} -state {off_state off}
add_pst_state <i>state_name</i> -pst <i>table_name</i> -state <i>supply_states</i>	Define the states of each of the supply nets for one possible state of the design create_pst pt -supplies { PN1 PN2 SOC/OTC/PN3 } add_pst_state s1 -pst pt -state { s08 s08 s08 } add_pst_state s2 -pst pt -state { s08 s08 off } add_pst_state s3 -pst pt -state { s08 s09 off }
bind_checker <i>instance_name</i> -module <i>checker_name</i> -elements <i>list</i> [-ports {{ <i>port_name net_name</i> }}*]	Inserts checker modules and binds them to design elements bind_checker chk p clks -module assert_partial_clk -elements {U1/U2 U1/U3} -ports {{prt1 clknet2} {port3 net4}}
connect_supply_net <i>net_name</i> [-ports <i>list</i>] [-pins <i>list</i>] [<-cells <i>list</i> / -domain <i>domain_name</i> >] [<-rail_connection <i>rail_type</i> -pg_type <i>pg_type</i> >]* [-vct <i>vct_name</i>]	Connect a supply net to supply ports and/or pins connect_supply_net v09 -ports {VDD U18/v9 U21/v9} connect_supply_net pd1_vdd -ports pll inst/vdd -vct upf2vlog_vdd
create_hdl2upf_vct <i>vct_name</i> -hdl_type {<vhdl vlog SV> [<i>typename</i>]} -table {{ <i>from_value to_value</i> }}*	Define value conversion table that can be used in converting HDL logic values into net_state_type values create_hdl2upf_vct vlog2upf_vss -hdl_type vlog -table {{X OFF} {0 ON} {1 OFF} {Z PARTIAL ON}}
create_power_domain <i>domain_name</i> [-elements <i>list</i>] [-include_scope] [-scope <i>instance_name</i>]	Define a power supply distribution network for a set of design elements create_power_domain PD1 -elements {top/U1} set_scope /top/U1 create_power_domain PD2
create_power_switch <i>switch_name</i> -domain <i>domain_name</i> -output_supply_port { <i>port_name</i> <i>supply_net_name</i> } {-input_supply_port { <i>port_name</i> <i>supply_net_name</i> }}* {-control_port { <i>port_name net_name</i> }}* {-on_state { <i>state_name input_supply_port</i> { <i>boolean_function</i> }}}* [-on_partial_state { <i>state_name</i> <i>input_supply_port</i> { <i>boolean_function</i> }}]* [-ack_port { <i>port_name net_name</i> { <i>boolean_function</i> }}]* [-ack_delay { <i>port_name delay</i> }}* [-off_state { <i>state_name</i> { <i>boolean_function</i> }}]* [-error_state { <i>state_name</i> { <i>boolean_function</i> }}]*	Define a switch in the power domain create_power_switch sw1 -domain PD_SODIUM -output_supply_port {vout VN3} -input_supply_port {vin1 VN1} -input_supply_port {vin2 VN2} -control_port {ctrl_small ON1} -control_port {ctrl_large ON2} -control_port {ss SUPPLY_SELECT} -on_state {partial_s1 vin1 {ctrl_small & !ctrl_large & ss}} -on_state {full_s1 vin1 {ctrl_small & ctrl_large & ss}} -on_state {partial_s2 vin2 {ctrl_small & !ctrl_large & !ss}} -on_state {full_s2 vin2 {ctrl_small & ctrl_large & !ss}} -error_state {no_small {!ctrl_small & ctrl_large}}
create_pst <i>table_name</i> -supplies <i>list</i>	Create power state table with a specific ordering of supply nets create_pst MyPowerStateTable -supplies {PN1 PN2 SOC/OTC/PN3}
create_supply_net <i>net_name</i> -domain <i>domain_name</i> [-reuse] [-resolve <unresolved one_hot parallel>]	Create a power or ground supply net create_supply_net v09 -domain PD1
create_supply_port <i>port_name</i> [-domain <i>domain_name</i>] [-direction <in out>]	Create a port on a power domain create_supply_port VN1 -domain PD1

Command Syntax	Description and Example
create_upf2hdl <i>vct_name</i> -hdl_type {<vhdl vlog SV> [<i>typename</i>]} -table {{ <i>from_value to_value</i> }*}	Define value conversion table that can be used in converting UPF supply_net_type.state values into HDL logic values <pre>create_upf2hdl vct upf2vlog_vdd -hdl_type {vlog} -table {{OFF X} {ON 1} {PARTIAL ON 0}}</pre>
get_supply_net <i>net_name</i> [-domain <i>domain_name</i>] [-scope <i>scope_name</i>]	Return logical net name for this supply net in the given scope <pre>get_supply_net VDD -domain mySoC_PD -scope /camera_i</pre>
load_upf <i>upf_file_name</i> [-scope <i>instance_name</i>] [-version <i>string</i>]	Set the scope to the specified instance and execute the specified UPF commands <pre>load upf my.upf -scope U1</pre>
map_isolation_cell <i>isolation_name</i> -domain <i>domain_name</i> [-elements <i>list</i>] [-lib_cells <i>list</i>] [-lib_cell_type <i>lib_cell_type</i>] [-lib_model_name <i>lib_model_name</i> {-port { <i>port_name net_name</i> }*}	Map a particular isolation strategy to a library cell or range of library cells. <pre>map_isolation_cell test_PD1 -domain PD1 -lib_cell_type jims_iso_fast</pre>
map_level_shifter_cell <i>level_shifter_name</i> -domain <i>domain_name</i> -lib_cells <i>list</i> [-elements <i>list</i>]	Map a particular level shifter strategy to a library cell or range of library cells <pre>map_level_shifter_cell shift_up -domain PwrDomZ -lib_cells {/lib2/LS LH /lib2/LS HL}</pre>
map_power_switch <i>switch_name</i> -domain <i>domain_name</i> -lib_cells <i>list</i>	Specify which power switch cell is to be used for the corresponding switch instance <pre>map_power_switch switch_1A -domain myPowerDomain -lib_cells /lib2A/switch2A</pre>
map_retention_cell <i>retention_name</i> -domain <i>domain_name</i> [-elements <i>list</i>] [-lib_cells <i>list</i>] [-lib_cell_type <i>lib_cell_type</i>] [-lib_model_name <i>lib_cell_name</i> {-port <i>port_name net_name</i> }*}	Specify which retention cell is to be used for the retention registers in a power domain <pre>map_retention_cell test_PDA -domain {PowerDomainA} -elements {foo/U1 foo/U2} -lib model name RETFF7 -port U1 N1</pre>
merge_power_domains <i>new_domain_name</i> -power_domains <i>list</i> [-scope <i>instance_name</i>] [-all_equivalent]	Merge two or more existing power domains into a single, new power domain <pre>merge_power_domains PD9 -power_domains {PD9A PD9B PD9C}</pre>
name_format [-isolation_prefix <i>string</i>] [-isolation_suffix <i>string</i>] [-level_shift_prefix <i>string</i>] [-level_shift_suffix <i>string</i>]	Allow user guidance in constructing port and signal names related to isolation or level shifter cells <pre>name_format -isolation_prefix "MY_ISO_" -isolation_suffix ""</pre>
save_upf <i>upf_file_name</i> [-scope <i>instance_name</i>] [-version <i>string</i>]	Create a UPF file relative to the specified scope <pre>save upf my_saved_file Jan14</pre>
set_design_top <i>instance</i>	Specify the top-level design instance <pre>set design_top ALU07</pre>
set_domain_supply_net <i>domain_name</i> -primary_power_net <i>supply_net_name</i> -primary_ground_net <i>supply_net_name</i>	Set the default power and ground supply nets for a power domain <pre>set_domain_supply_net PD1 -primary_power_net PG1 -primary_ground_net PG0</pre>
set_isolation <i>isolation_name</i> -domain <i>domain_name</i> <-isolation_power_net <i>net_name</i> -isolation_ground_net <i>net_name</i> -isolation_power_net <i>net_name</i> -isolation_ground_net <i>net_name</i> -no_isolation> [-elements <i>list</i>] [-clamp_value <0 1 latch Z>] [-applies_to <inputs outputs both>]	Specify the elements in the domain to isolate using the specified strategy <pre>set_isolation outputs_only -domain PD1 -isolation_power_net VDDbackup -clamp_value 1 -applies_to outputs</pre>

Command Syntax	Description and Example
set_isolation_control <i>isolation_name</i> -domain <i>domain_name</i> -isolation_signal <i>signal_name</i> [-isolation_sense <high low>] [-location <self parent sibling fanout automatic>]	Specify the control signals for a previously defined isolation strategy <pre>set_isolation outputs_only -domain PDI -isolation_power_net VDDbackup -clamp_value 1 -applies_to outputs set_isolation control_outputs_only -domain PDI -isolation_signal cpu_iso -isolation_sense low -location parent</pre>
set_level_shifter <i>level_shifter_name</i> -domain <i>domain_name</i> [-elements <i>list</i>] [-threshold <i>value</i>] [-applies_to <inputs outputs both>] [-rule <low_to_high high_to_low both>] [-location <self parent sibling fanout automatic>] [-no_shift]	Specify a level shifter strategy <pre>set_level_shifter shift_up -domain PowerDomainZ -applies_to outputs -threshold 0.02 -rule both</pre>
set_pin_related_supply <i>library_cell</i> -pins <i>list</i> -related_power_pin <i>supply_pin</i> -related_ground_pin <i>supply_pin</i>	Define the related power/ground pair for a library cell <pre>set_pin_related_supply library1/cell1 -pins {A B C} -related_power_pin VDDX -related_ground_pin VSSX</pre>
set_power_switch <i>switch_name</i> -output_supply_port { <i>port_name</i> <i>supply_net_name</i> } -input_supply_port { <i>port_name</i> <i>supply_net_name</i> }* -control_port { <i>port_name</i> <i>net_name</i> }* -on_state { <i>state_name</i> <i>input_supply_port</i> { <i>boolean_function</i> }}* -on_partial_state { <i>state_name</i> <i>input_supply_port</i> { <i>boolean_function</i> }}* -off_state { <i>state_name</i> { <i>boolean_function</i> }}* -error_state { <i>state_name</i> { <i>boolean_function</i> }}*	Extend a switch by adding the input supply port(s) and output supply port to the switch <pre>set_power_switch hmacro/sw1 -input_supply_port {i1 always_on_power} -output_supply_port {o1 switched_power} -control_port {on} -on_state {sw1_on on {on}} -off_state {sw1_off on {-on}}</pre>
set_retention <i>retention_name</i> -domain <i>domain_name</i> <-retention_power_net <i>net_name</i> -retention_ground_net <i>net_name</i> -retention_power_net <i>net_name</i> -retention_ground_net <i>net_name</i> > [-elements <i>list</i>]	Specify which registers in the domain need to be retention registers and set the save and restore signals for the retention functionality <pre>set_retention my_retention -domain PDA -retention_power_net volt_high</pre>
set_retention_control <i>retention_name</i> -domain <i>domain_name</i> -save_signal {{ <i>net_name</i> <high low posedge negedge>}} -restore_signal {{ <i>net_name</i> <high low posedge negedge>}} -assert_r_mutex {{ <i>net_name</i> <high low posedge negedge>}}* -assert_s_mutex {{ <i>net_name</i> <high low posedge negedge>}}* -assert_rs_mutex {{ <i>net_name</i> <high low posedge negedge>}}*	Specify the control signals and assertions for a previously defined retention strategy <pre>set_retention my_retention_strategy -domain PDA set_retention_control my_retention_strategy -domain PDA -save_signal {power_ctrl_inst/save_1 high} -restore_signal {power_ctrl_inst/restore_1 low} -assert_rs_mutex {clock_a posedge}</pre>
set_scope <i>instance</i>	Specify the current UPF scope <pre>set_scope foo/bar set_scope ..</pre>
upf_version [<i>string</i>]	Specify the version for the UPF file/syntax <pre>upf version 1.0</pre>