# Innovus/Tempus Non-functional ECO

**tempus**

-eco

Needed to enables the Tempus ECO feature of the software and for running the Tempus ECO flow using the eco_opt_design command

tempus #> eco_opt_design -setup

**ERROR: (IMPESO-322): eco_opt_design has run once in the session. Cannot run eco_opt_design again in the same session.

To run again, restart the session.

*info: Ending eco_opt_design: (totcpu=0:00:00.0, real=0:00:00.0, mem=3872.3M)

tempus #> get_eco_opt_mode -allow_multiple_incremental

-allow_multiple_incremental false # bool, default=false

**false**

tempus #> set_eco_opt_mode -allow_multiple_incremental true

**ERROR: (IMPESO-304): set_eco_opt_mode -allow_multiple_incremental cannot be changed once incremental mode has started.

setEcoMode

refinePlace

extractRC

ecoRoute

The DRV/Setup/Hold optimizations can be called in any order, since timing is incrementally updated. But it is recommended to run DRV fixing first because it can impact Setup and Hold timing. Then, it is usually better to continue with Setup optimization as it can help Hold optimization by creating extra Setup timing margin.

*Timing Signoff Optimization Using Tempus and Innovus Rapid Adoption Kit (RAK)*

set_eco_opt_mode -allow_multiple_incremental true

set_eco_opt_mode -eco_file_prefix DRV

eco_opt_design -drv

set_eco_opt_mode -eco_file_prefix SETUP

eco_opt_design -setup

set_eco_opt_mode -eco_file_prefix HOLD

eco_opt_design –hold

tempus:

**set_eco_opt_mode**

[-help]

[-reset]

[-add_inst {true|false}]

[-add_load {true|false}]

[-allow_multiple_incremental {true|false}]

this option allows the user to run **several** eco_opt_design commands in a row using the same initial ECO Timing DB. In the same session, the user can fix Setup, Hold and DRV violations. An **ECO file** is generated **per eco_opt_design** call.

[-allow_skewing {true|false}]

[-along_route_buffering {true|false}]

[-buffer_cell_list cell_list]

[-check_drv_from_hold_views]

[-check_type {early | late | both}]

[-clock_cell_list {}]

[-clock_max_level <INT>]

[-delete_inst {true | false}]

[-drv_margin <FLOAT>]

[-disable_geometry_checks {true|false}]

[-eco_file_prefix name]

[-fix_clock_drv {true|false}]

[-fix_data_drv {true|false}]

[-fix_glitch {true|false}]

[-fix_hold_allow_setup_optimization {true|false}]

[-fix_hold_allow_setup_tns_degrade {true|false}]

[-fix_hold_with_margin <FLOAT>]

[-fix_ir_drop {true|false}]

[-fix_max_cap {true|false}]

[-fix_max_tran {true|false}]

[-fix_si_slew {true|false}]

[-fix_xtalk {true|false}]

[-hold_target_slack number]

[-hold_xtalk_delta_threshold <DOUBLE>]

[-hold_xtalk_slack_threshold <DOUBLE>]

[-ignore_drv_checks {true|false}]

[-keep_temp_files {true|false}]

[-legal_only {true|false}]

[-load_cell_list <cell_list>]

[-load_eco_opt_db name]

Specifies the path of the directory where previously generated ECO Timing DB files are kept. **If not specified, the ECO Timing DB files will be generated**.

Default=""

Note: If set, this parameter must be applied before eco_opt_design.

[-load_irdrop_db name]

[-max_cap_margin <float>]

[-max_slack <FLOAT>]

[-max_paths <paths>]

[-max_run_time <minutes>]

[-max_tran_margin <float>]

[-nworst <INT>]

[-optimize_core_only {true|false}]

[-optimize_sequential_cells {true|false}]

[-optimize_replicated_modules {true|false}]

[-partition_list_file <string>]

[-pba_effort {medium|high}]

[-post_sta_tcl file ]

[-power_aware {true|false}]

[-power_opt_focus {total|leakage|dynamic}]

[-pre_sta_tcl file ]

[-prefix_name string ]

[-pruned_block_name string ]

[-resize_inst {true|false}]

[-retime {none| aocv | path_slew_propagation | aocv_path_slew_propagation}]

[-routing_congestion_aware {true|false}]

[-save_eco_opt_db <dir_name>]

Path of directory where generated database files are kept.

Default: ecoTimingDB

Note: If set, this parameter must be applied before write_eco_opt_db.

[-select_hold_endpoints string]

[-select_drv_net_file <FILE>]

[-select_setup_endpoints string]

[-setup_recovery {true|false}]

[-setup_target_slack number]

[-setup_xtalk_delta_threshold <DOUBLE>]

[-setup_xtalk_slack_threshold <DOUBLE>]

[-skip_drv_net_file <FILE>]

[-specify_hold_endpoints_margin string]

[-specify_setup_endpoints_margin string]

[-swap_inst {true | false}]

[-verbose {true | false}]

Sets the global parameters for ECO. Parameters that you specify with set_eco_opt_mode are then used automatically when you run ECO optimization by running the eco_opt_design command.

**eco_opt_design**

**set_eco_mode**

[-help]
[-reset]
[-batch_mode {true | false}]
[-honor_dont_touch {true | false}]
[-honor_dont_use {true | false}]
[-honor_fixed_net_wire {true|false}]
[-honor_fixed_status {true | false}]
[-honor_power_intent {true|false}]
[-leq_check {true | false}]
[-prefix_name prefix]
[-refine_place {true|false}]
[-si_effort {low | medium}]
[-update_timing {true | false}]

Controls the timing updates during ECO changes, checks for dont touch and dont use on cells, and allows the option of swapping of non-equivalent cells.

**write_eco_opt_db**

[-help]

Saves an ECO Timing DB after performing timing analysis(explicitly report_timing, report_constraint, report_analysis_coverage is NOT necessary), in a non-distributed mode or in a distributed-MMMC session. In case several views are active at the same time, ECO Timing DB for each of them is saved. The data is saved in the directory pointed by the set_eco_opt_mode -save_eco_opt_db parameter.

Note: This command requires a 64-bit executable.

Examples

The following command saves an ECO Timing DB in the mydb directory.

tempus> set_eco_opt_mode -save_eco_opt_db mydb
tempus> write_eco_opt_db

**ECO db consistent**

set_delay_cal_mode siAware should be consistent between “write_eco_opt_db” and “tempus -eco”

**ERROR: (IMPESO-530): Tempus session has "set_delay_cal_mode siAware true" forced by the user, but the loaded eco_db were generated with "set_delay_cal_mode siAware false".

To avoid this error, make sure that this option is set to the same value in Tempus session and eco_db generation sessions.

*info: Ending eco_opt_design: (totcpu=0:00:21.2, real=0:00:21.0, mem=3758.3M)

| generate db | eco opt design |
|---|---|
| source ../scripts/spef.tcl<br>set_delay_cal_mode **-SIAware true**<br>write_eco_opt_db | source ../scripts/spef.tcl<br>set_delay_cal_mode **-SIAware true**<br>set_eco_opt_mode -load_eco_opt_db ecoTimingDB<br>eco_opt_design -hold |

innovus(legacy)
**setEcoMode**

## Speeding up run time of interactive ECO commands by disabling timing updates

problem

I am executing several interactive ECO commands (ecoAddRepeater, ecoChangeCell, ecoDeleteRepeater). After each command, the tool updates the timing, which involves placement legalization (refinePlace), extraction (extractRC), and timing analysis. How can I prevent it from doing this until all the ECOs are completed?

Solution

To improve the run time, you can execute the commands in batch mode and disable timing analysis until the end.

Here is an example script.

Legacy UI:

```
setEcoMode –refinePlace false -updateTiming false -batchMode true
source ecoAddRepeater.tcl ;#Script with large number of ecoAddRepeater commands
setEcoMode –refinePlace true -updateTiming true –batchMode false
refinePlace
report_timing/timeDesign ; # Updating timer is optional if your script is already
                           # doing a timer update later in the flow
```

Common UI:

```
set_db eco_refine_place 0
set_db eco_update_timing 0
set_db eco_batch_mode 1

source ecoAddRepeater.tcl ;#Script with large number of ecoAddRepeater commands

set_db eco_refine_place 1
set_db eco_update_timing 1
set_db -eco_batch_mode 0

place_detail/time_design ; # Updating timer is optional if your script is already
                           # doing a timer update later in the flow
```

Notes:

Make sure that -batchMode is set to false before saving the design. If it is set to true, any other setEcoMode commands will not be observed. setEcoMode commands such as -refinePlace and -updateTiming should always be set prior to, or at the same time as, setting -batchMode true.

Make sure that you exit batch mode (setEcoMode -batchMode false) prior to running timing analysis.

Otherwise, report_timing will report the following:

No constrained timing paths found.

Paths may be unconstrained (try '-unconstrained' option).

Refer to Innovus Command Reference > setEcoMode.

Refer to Innovus Stylus Common UI Text Reference Manual > eco Category Attributes

**defOut**

- **-floorplan**
- **-placement**
- **-netlist**
- **-routing**
- **-ioRow**