set_scan_configuration

NAME

set_scan_configuration

Specifies the scan chain design.

SYNTAX

```
int set_scan_configuration
    [-chain_count chain_count
     | -max_length max_chain_length
     | -exact_length chain_length
     | -count_per_domain chain_count]
    [-add_lockup true | false]
    [-clock_mixing no_mix | mix_edges | mix_clocks | mix_clocks_not_edges]
    [-add_test_retiming_flops begin_and_end | begin_only | end_only | none]
    [-create_dedicated_scan_out_ports true | false]
    [-internal_clocks single | none | multi]
    [-insert_terminal_lockup true | false]
    [-lockup_type latch | flip_flop]
    [-mix_internal_clock_driver true | false]
    [-preserve_multibit_segment false | true]
    [-style multiplexed_flip_flop | clocked_scan | lssd | combinational | scan_enabled_lssd | none]
    [-shared_scan_in pin_count]
    [-exclude_elements exclude_list]
    [-voltage_mixing true | false]
    [-power_domain_mixing true | false]
    [-test_mode mode_name]
    [-domain_based_scan_enable true | false]
    [-reuse_mv_cells true | false]
    [-pipeline_scan_enable true | false]
    [-pipeline_fanout_limit max_scan_cells]
    [-create_test_clocks_by_system_clock_domain true | false]
    [-replace true | false]
    [-hierarchical_isolation true | false]
    [-static_x_chain_isolation X_chains_isolation]
```

Data Types

chain_count	integer
max_chain_length	integer
chain_length	integer
pin_count	integer
exclude_list	list
max_scan_cells	integer

ARGUMENTS

-chain_count *chain_count*

Specifies the number of scan chains to build. The tool attempts to meet this constraint. If unable to do so, the tool issues a warning. See the DESCRIPTION section for more information.

-max_length ***max_chain_length***

Specifies the maximum allowed length of the scan chains. The tool attempts to meet this constraint. If unable to do so, the tool issues a warning. See the DESCRIPTION section for more information.

-exact_length ***chain_length***

Specifies a positive integer that indicates the exact chain length. **insert_dft** builds, as much as possible, scan chains with the specified length. See the DESCRIPTION section for more information.

-count_per_domain ***chain_count***

Specifies a positive integer for the number of chains that **insert_dft** is to build per clock domain. See the DESCRIPTION section for more information.

-add_lockup true | false

When set to its default of **true**, inserts lockup latches (synchronization element) between clock domain boundaries on scan chains. To disable synchronization element insertion, set this option to **false**. If the scan specification does not mix clocks on chains, **insert_dft** ignores this option.

-clock_mixing no_mix | mix_edges | mix_clocks | mix_clocks_not_edges

Specifies whether the **insert_dft** command can include cells from different clock domains in the same scan chain. The following allowed values control the clocking of cells in scan chains to be inserted by the **insert_dft** command:

- **no_mix** (the default)
  Cells must be clocked by the same edge of the same clock.
- **mix_edges**
  Cells must be clocked by the same clock, but the clock edges can be different.
- **mix_clocks_not_edges**
  Cells must be clocked by the same clock edge, but the clocks can be different.
- **mix_clocks**
  Cells can be clocked by different clocks and different clock edges.

-add_test_retiming_flops begin_and_end | begin_only | end_only | none

Specifies whether the **insert_dft** command can insert leading edge retiming lockup flip-flops to scan chains clocked with trailing edge scan elements. If a scan chain contains only leading edge scan elements, a retiming lockup flip-flop is not inserted. The following allowed values control the lockup flip-flop insertion by the **insert_dft** command:

- **begin_and_end**
  Retiming lockup flip-flops are added to the scan chain beginning and end.
- **begin_only**
  Retiming lockup flip-flop is added only to the scan chain beginning.
- **end_only**
  Retiming lockup flip-flop is added only to the scan chain end.
- **none** (the default)
  No retiming flip-flop is inserted.

-create_dedicated_scan_out_ports true | false

When set to **true**, instructs **insert_dft** to implement dedicated scan-out signal ports on the current design. When set to **false** (the default), **insert_dft** uses mission-mode ports as scan-out ports whenever possible.

-internal_clocks single | none | multi

This option applies only to the multiplexed flip-flop scan style; it is ignored for other scan styles.

The **-internal_clocks** option can be set to the following values:

- **single**
  The **insert_dft** command treats internal clock network regions driven by any combinational gate, including buffers and inverters, as separate clocks for the purposes of scan chain architecture.
- **none**
  The **insert_dft** command does not perform internal clock analysis. This is the default.
- **multi**
  The **insert_dft** command treats internal clock network regions driven by multiple-input gates, such as MUX cells, as separate clocks for the purposes of scan chain architecture. Buffers and inverters are transparent for the analysis.

Integrated clock-gating cells are transparent for the determination of internal clocks.

The **-internal_clocks** setting is common to all test modes. As a result, this option can be specified only if the **-test_mode** option value is set to 'all'. Otherwise, the tool issues a warning message, ignores the **-internal_clocks** option, and processes the remaining **set_scan_configuration** options.

-insert_terminal_lockup true | false

When set to **true**, inserts synchronization element at the end of scan chains. The default is **false**. This option applies only to the multiplexed flip-flop scan style; it is ignored for other scan styles.

-lockup_type latch | flip_flop

Selects the type of synchronization element used in the scan chain. The default lock-up type is a level-sensitive latch. If you choose **flip_flop** as the lock-up type, an edge-triggered flip-flop is used as the synchronization element. You can use this option in conjunction with the **-add_lockup** option or **-insert_terminal_lockup** options.

-mix_internal_clock_driver true | false

When set to **true**, **insert_dft** allows mixing in a scan chain of any internal clocks that are derived from a single external clock pin. This capability can only be activated when the clock mixing scan configuration is either **no_mix** or **mix_edges**.

-preserve_multibit_segment false | true

Specifies whether to treat sequential multibit components as scan segments. A multibit component is a group of cells with identical functionality, inferred from RTL code or created by the **create_multibit** command. Depending on tool configuration and design constraints, synthesis implements a multibit component using multibit cells or single-bit cells.

When this option is set to the default of **false**, the **insert_dft** command treats the cells inside a multibit component as discrete sequential cells that can be reordered, split up, and rebalanced across scan chains as needed. Note that this does not affect the functional behavior of a multibit component as used by synthesis.

When this option is set to **true**, the **insert_dft** command treats each sequential multibit component as a scan segment. As a result, the cells inside a multibit component cannot be separated for length-balancing purposes. You can report the multibit scan segments that will be used by using the **preview_dft -show {segments}** command.

To report the multibit components identified by synthesis, use the **report_multibit** command.

-style multiplexed_flip_flop | clocked_scan | lssd | combinational | scan_enabled_lssd | none

Identifies the scan style. Select **none** if you have not selected a scan style for the design. By default, **insert_dft** uses the scan style value specified by environment variable **test_default_scan_style**.

-shared_scan_in *pin_count*

Specifies the number of scan-in pins to be shared between chains. The pins will be shared between chains which do not have a scan-in pin assigned to them. If there are existing scan-in pins in the design that are not already assigned to any chain, they will be used. If there are no existing pins that can be used, new pins are created and shared between the chains.

-exclude_elements *exclude_list*

Specifies the list or collection of design objects to exclude from scan chain insertion. You can specify leaf cells, hierarchical cells, designs, or scan segment names. If hierarchical cells or designs are specified, the specification applies to all sequential cells inside those cells or designs.

Specification of scan segment names results in spurious UID-95 warnings, which can be ignored.

This option causes the specified cells to be excluded from scan stitching by the **insert_dft** command. It does not prevent the specified cells from being scan-replaced by the **compile -scan**, **compile_ultra -scan**, or **insert_dft** commands, and it does not cause excluded test-ready cells to be unscanned by the **insert_dft** command. To prevent cells from being scan-replaced, use the **set_scan_element false** command.

This option is cumulative with other previously specified **-exclude_elements** lists. Refer to the example at the end of this man page. Wildcards and collections are supported.

This option does not affect DFT connections to clock-gating cell test pins; use the **set_dft_clock_gating_configuration -exclude_elements** command instead.

-voltage_mixing true | false

When set to **true**, allows **insert_dft** to insert scan elements from different voltage domains into the same scan chain. This requires you to insert level shifters after scan insertion is complete. The default is **false**.

-power_domain_mixing true | false

Specifies that **insert_dft** can mix scan elements from different power domains into the same scan chain, when set to **true**. The default is **false**.

-test_mode *mode_name*

Indicates which test mode the scan configuration applies to. The default is the last test mode specified with the **current_test_mode** command, or the last test mode created by **define_test_mode** if no current test mode has been set.

-domain_based_scan_enable true | false

When set to **true**, **insert_dft** creates one scan-enable signal per clock domain. This process is independent of the value of the **-clock_mixing** option. The default is **false**.

This option is intended to be used to create a core that can use pipelined scan-enable signals at a higher integration level.

To make domain-specific scan-enable connections for a subset of clocks instead of all clocks, use the **set_dft_signal -type ScanEnable -connect_to <clock>** command instead.

-reuse_mv_cells true | false

Specifies whether **insert_dft** should reuse existing level shifters, isolation cells, and enabled level shifters if they are on the scan path. When set to **true** (the default), **insert_dft** will attempt to reuse these cells to avoid creating dedicated test ports and additional multivoltage cells. When set to **false**, **insert_dft** will always create new multivoltage cells, creating new test ports as needed.

-pipeline_scan_enable true | false

When set to **true**, **insert_dft** creates a scan-enable pipelining stage for each scan-enable signal. The default is **false**.

-pipeline_fanout_limit *max_scan_cells*

When set to a positive integer value, **insert_dft** creates each scan-enable signal so that the number of flip-flops driven by the same scan-enable signal does not exceed the value. This option only takes effect if the **-pipeline_scan_enable** or **-domain_based_scan_enable** option (or both) is set to **true**. The default is to not apply a limit.

-create_test_clocks_by_system_clock_domain true | false

When **true**, **insert_dft** creates dedicated test clocks (for example, test_scan_clock_a) according to different system clocks. Thus, scan cells that have different system clocks have different test clocks. When **false** (the default), **insert_dft** creates test clocks without considering system clocks.

-replace true | false

When **true** (the default), **insert_dft** replaces sequential cells with scan cells, if the sequential cells are not violated by scan design rule checking. To disable scan replacement, set this option to **false**.

-hierarchical_isolation true | false

When **true**, **insert_dft** builds hierarchical isolation logic, so that dedicated subdesign scan-out signals are gated by the design scan-enable signal. This prevents long top-level scanout nets from toggling during functional operation. When **false** (the default), **insert_dft** does not add hierarchical isolation logic.

When this option is set to **true**, you cannot disable the use of shared subdesign scan-out signals by setting the **test_dedicated_subdesign_scan_outs** variable to **true**.

DESCRIPTION

There are four related options to control the number of scan chains. Only one option can take effect at a time. They are, in order of highest precedence first:

- **-max_length**
- **-chain_count** or **-count_per_domain**
- **-exact_length**

You can use these options to directly or indirectly specify the scan chain configuration for your design. If none of these options are specified, the tool builds the minimum number of scan chains consistent with the clock mixing constraints.

The following option controls clock domain identification:

```
-internal_clocks single | none | multi
```

Clocks driven by separate top-level input ports are always considered to be separate clocks for scan chain routing. The **-internal_clocks** option controls whether regions of the same clock signal, driven by multi-input combinational gates, should also be treated as separate clock domains. For designs where the entire clock tree is balanced through these multi-input combinational gates, you can specify a value of **none** to treat the entire clock port fanout as a single clock domain. For designs where clock latencies might differ in the fanout of each multi-input gate, it is recommended that you specify a value of **multi** to treat these regions as separate clock domains.

The following options control how separate clock domains can be mixed during scan chain routing:

```
-clock_mixing no_mix | mix_edges | mix_clocks | mix_clocks_not_edges
-mix_internal_clock_driver true | false
```

The **-clock_mixing option** allows you to control two independent aspects of mixing: whether separate clock domains can be mixed together on the same scan chain, and whether scan cells clocked by different clock edge polarities can be mixed together on the same scan chain. Four possible values allow all four combinations of these two aspects to be specified. When separate clock domain mixing is disabled by specifying a **-clock_mixing** value of **no_mix** or **mix_edges**, the **-mix_internal_clock_driver** option can be used to allow only separate clock domains driven by the same input port (such as those created with the

-internal_clocks option) to be mixed.

For all possible clock domain mixing configurations, lockup latches are inserted between the separate clock domains when they are mixed on the same scan chain.

EXAMPLES

The following example allows DFT Compiler to establish the maximum sequential length of scan chains (30 in this example):

```
prompt> set_scan_configuration -max_length 30
```

The following example demonstrates how a mix of object types can be provided to the **set_scan_configuration** command. In this example, the **-exclude_elements** option takes **s2/ff1** which is a cell name, **s1/f***, which is a wild card representation, **s0/1** which is a segment name, **$foo** which is a collection, and **U5** which is a design instance name.

```
prompt> set foo [get_cell s4/*]

prompt> set_scan_config -exclude_elements \
               [list s2/ff1 s1/f* s0/1 $foo U5]
```

The following example allows DFT Compiler to mix scan cells in the same scan chain when a clock passes through multi-input combinational cells. Lockup latches will be inserted between scan cells driven by different multi-input combinational cells. Scan cells driven by separate clock input ports will not be mixed.

```
prompt> set_scan_configuration -internal_clocks multi

prompt> set_scan_configuration -clock_mixing no_mix

prompt> set_scan_configuration -mix_internal_clock_driver true
```

SEE ALSO

current_design(2)
insert_dft(2)
preview_dft(2)