Foreword

As size of logic circuit design gets larger, efficient reuse of design property has become extremely important. The design property, which is also called IP (Intellectual Property), is essential technology to realize System-on-a-Chip.

The organization called VSI (Virtual Socket Interface) is now trying to commonize IP to enable us to structure a system by purchasing IP from multiple vendors that a large-scale system can easily be integrated into a single chip. Therefore, a circuit, which is larger than conventional ones, can be designed within the similar timeframe.

Software IP is described in RTL (Register Transfer Level) by using mainly HDL (Hardware Description Language). By having IP described in RTL, changes can easily be made to the HDL description. Also, in the parameterized descriptions, a circuit can easily be changed according to the object of an application.

These software IP are realized in gate level circuits at the end by using logic synthesis tools and by specifying design constraints and technology. The design constraint includes specification of timing condition, area, and operation environment etc. as the required circuit performance. If the constraint given at the time of synthesis is not appropriate, the circuit may be synthesized incorrectly and therefore it becomes unlikely to meet the target performance. If description is not appropriate, it will also become hard to meet the target performance.

This Design Style Guide defines the design style to proceed with HDL design based on the IP reuse method. Commonizing design style enables design methods such as description style, synthesis and verification, which differ in each designer, to be commonized. Therefore, designers other than the original author can modify descriptions or improve readability to facilitate understanding.

The Design Style Guide bases on the knowledge of syntax rules on Verilog-HDLsyntax regulations. Please refer to IEEE1364Verilog Hardware Description Language Reference Manual or other Verilog-HDL manuals for Verilog-HDL syntax regulations. Logical circuit design here subjects to the verification by simualtor supporting Verilog-HDL and the usage of logic synthesis tools. For these, please refer to the manual of each tool.

The languages used for logic circuit design includes Verilog-HDL and VHDL. The design style guide is based on Verilog-HDL. Please refer to the syle guide VHDL version for VHDL.

The Design Style Guide aims at facilitating IP propagation based on generalized rules of logic synthesis design with Verilog-HDL (or VHDL), which are applied in various Japanese semiconductor companies and electronic manufacturers. Semiconductor Technology Academic Research Center (STARC) compiled opinions of participating companies that IP Reuse Subcommittee agreed upon to document the Design Style Guide. I would like to thank the members of Semiconductor Technology Academic Research Center who made substantial contributions to the content of the Design Style Guide.

February 1, 2002
Hiroyasu Hasegawa
hd Lab, inc.

For the Design Style Guide 2002

The Design Style Guide has been updated once a year to respond to technological innovation. The Design Style Guide 2002 **substantially** includes contents, which were deferred in the first version, the Design Style Guide 2001. We have also enhanced the explanation part to deepen the readers ' understanding. **Consequently** it has about twice as many pages as the 2001 version. The main revised points are as follows.

[1] "3.3 Design for Test(DFT) "in Section3 includes the detailed regulations and explanations about what kind of considerations is necessary as logic circuit design for the" design for testability " of LSI foundry test. (9 clauses including 37 new items)

[2] As appendix, the usage methods and the know-hows of the logic synthesis tools, that are Design Compiler by Synopsys, Inc., and Build Gates by Cadence Design Systems, Inc., are added.

[3] We have added explanations on low power consumption design in 3.4 Low-power Consumption Design and the usage method of CVS, a version management tool in 3.5.7 - 3.5.9, in Section 3. (14 new items) .

[4] The levels of each items have been changed from " mandatory ", " prohibited ", " recommended ", " caution " and " reference " to " mandatory ", " recommended 1 ", " recommended 2 ", " recommended 3 " and " reference " to clarify each item.

[5] Each item has been reviewed in other than 3.3, 3.4 and 3.5.7 in Section 3 deleting 14 items and adding 27 items. 59 items among existing items have been modified. Explanation in each item was modified accrodingly.

[6] The explanation part regarding the verification method of the cause for the different result between simulators has been substantially modified and added.

[7] The explanation of simultaneous RAM ragarding clock tree synthesis has been substantially modified.

The Design Style Guide consists of the following sections.

“Section 1: Basic Design Constraints” introduces basic design constraints, which should be considered when starting design, that are naming conventions, design style, clock design, considerations for synchronous design and asynchronous design, and the way of thinking for hierarchy design.

“ Section 2: RTL Description Techniques ”introduces the basic RTL description style necessary to create RTL description by Verilog-HDL, including the description style of combinatorial circuits and sequential circuits. How to use *always construct*, *function statement*, *if statement* and *case statement* etc., is also explained.

“ Section 3: RTL Design Methodology ”on the assumption of circuit partitioned design, introduces the creation method of the function library, IP parameterization, design for test, low power consumption design and design data management. By following this design method, IP reusability will be improved.

“ Section 4: Verification Techniques ”introduces the techniques to execute simulation; parameterization to describe test bench, task usage and procedure of verification.

As an appendix,“ A-5 Logical synthesis by Design Compiler ”introduces the usage method and the know-hows of the logic synthesisi tools.

The remarks used in this document are categorized as follows.

Remarks	Contents
mandatory	Items, which should always be followed
recommend 1	Items, which should be followed, but will not cause problem if appropriate counter measures are taken.
recommend 2	Recommended design style itmes regarding issues on circuit quality, readability and the usage of various tools.
recommend 3	Recommended styles, but strict observation is not required. Cautions are needed, however.
reference	Reference items of design know-how, which are better to know

Table of Contents

3.3.1. Clocks and Resets for DFT

[1]	The clocks must be directly controllable from external input ports	mandatory
[2]	When there are two selectable clock systems, one clock system must be selected throughout testing	mandatory
[3]	The output of random logic should not be used as a clock	recommend 1
[4]	The reset for the FFs must be directly controllable from an external input port	mandatory

Explanation

When inserting scans, the most important consideration is to structure the circuitry so that the scan shift is performed safely during the scan test. A circuit structure wherein the scan shift is performed safely is a circuit structure wherein the clock pins of all FFs connected to the scan path at the time of the scan shift can be controlled directly from an external input port, and where none of the asynchronous set or reset signals for any of the FFs connected to the scan path will become active during the scan shift. If each circuit structure is one wherein an asynchronous set or reset signal becomes active during the scan shift, the data in the FF will be lost during the scan shift (i.e., the FF will be set or reset during the scan shift), meaning that the data will not be shifted as it should. Because of this, it is necessary to structure the circuit so that the asynchronous set and reset signals will be inactive during the scan shift.

* The clocks must be directly controllable from external input ports

In order to insert scans, it must be possible to control the clock pins of the FFs from an external input port during the scan test. If it is not possible to control directly the clock pins of the FFs from an external input port, then the scan insert tool will exclude the affected FFs from the scan. Note also that it will be difficult to detect faults using the ATPG tool for any parts for which scans have not been inserted.

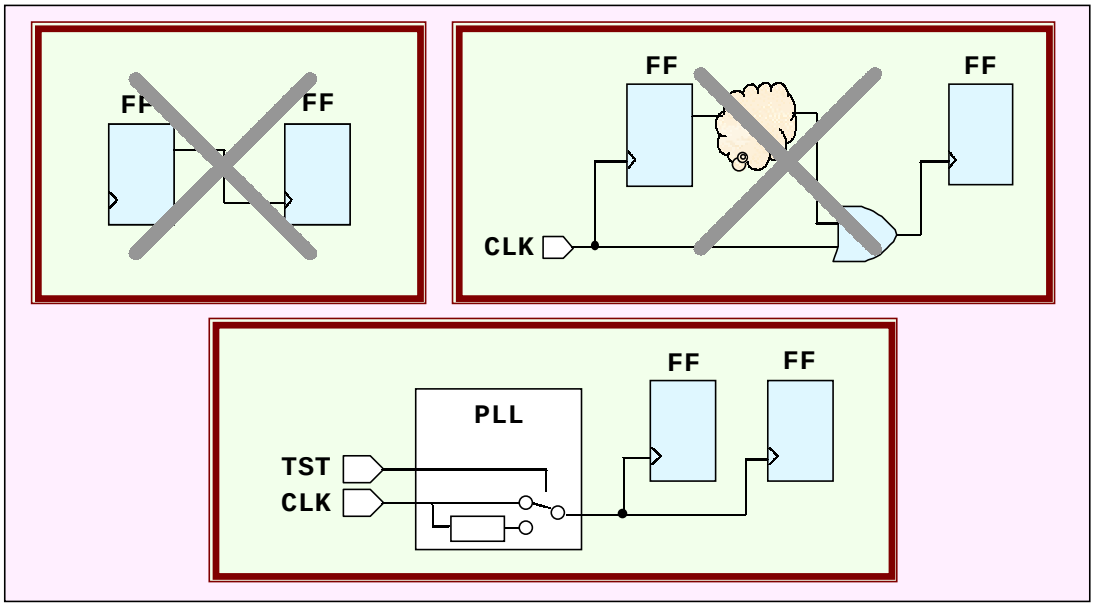

Figure 3-3 Examples of internally generated clocks, clocks that cannot be controlled directly, and PLLs

FFs wherein the clock lines are gated, and FFs where, as is shown at the top of Figure 3-3, the output of a FF is connected to a FF clock pin, will be excluded from the scan. In such cases, it is necessary, for example, to switch the clock lines during the scan test so that they can be controlled directly from an external input port. Note that it is also not possible to insert scans when the output of a hard macro within the LSI acts as a clock. PLLs and DLLs are used as clock generator macros in LSIs. These macros usually have test ports such as shown by the bottom figure of Figure 3-3 so that the reference clock that is input into the PLL when in test mode is output without modification to the clock lines. If PLLs or DLLs not equipped with this type of switching circuitry are used, the designer will have to insert circuitry to switch to an external clock.

When it comes to “the clock must be directly controllable from an external input port”, see “3.3.5.DFT in Clock Lines” for specific measures to be taken.

* When there are two selectable clock systems, one clock system must be selected throughout testing

Even if the clocks can be controlled from an external input port, if, as shown in Figure 3-4, it is possible to switch between two clocks, it will still not be possible to insert a scan. A test signal to control the select signal for the selector as shown in the figure must be used so that, during testing, the same clock will be selected throughout the entire process.

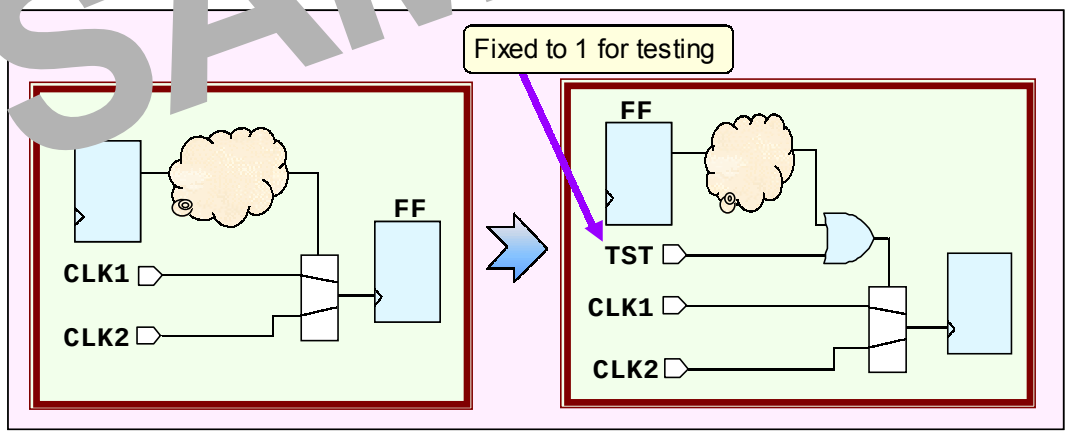

Figure 3-4 DFT for circuits wherein there are two selectable clock systems

* The output of random logic should not be used as a clock

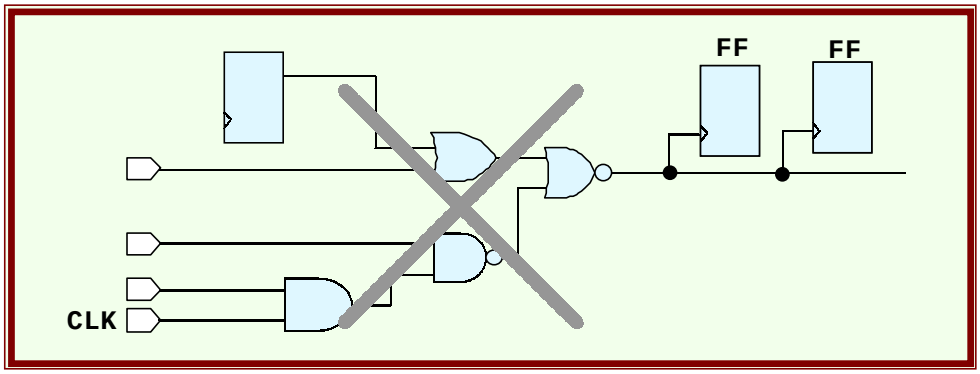

Figure 3-5 The output of random logic should not be used as a clock

As is shown in Figure 3-5, if the output of random logic is to be used as a clock, insert a selector at the final output of the random logic to make it possible to select an external clock.

The technique for using AND gates, OR gates, or selectors in clock lines is described in “3.3.5. DFT in Clock Lines”.

* The reset for the FFs must be directly controllable from an external input port

In addition to paying attention to the clock systems when inserting scans, the designer must also pay attention to the reset lines. DFT for reset lines requires that they be structured so that no reset is applied to the FF during the scan shift, which would cause it to lose its data. While it is possible to insert scans even in cases where test signals, etc., are used to force the reset lines within the LSI to be inactive, doing so will make it impossible for the ATPG tool to detect faults in the reset lines. As a result, it is necessary for the reset lines to be directly controllable from external input ports.

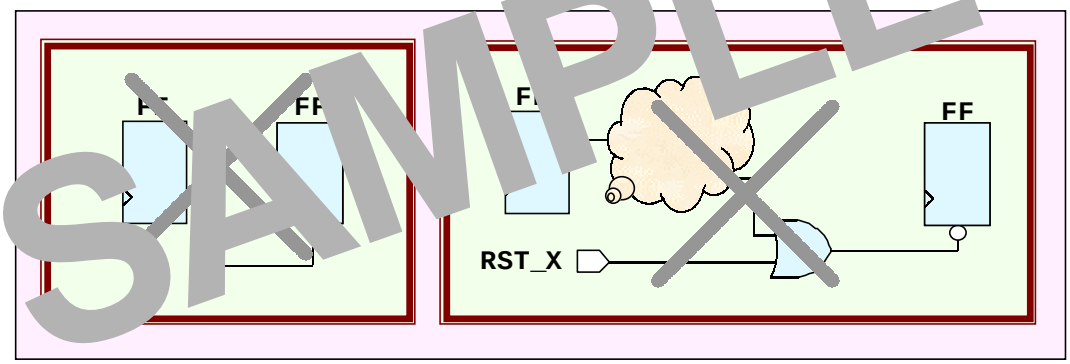

Figure 3-6 Internally generated resets and circuitry wherein there is no direct control from an external input port

As is shown on the left-hand side of Figure 3-6, when the output of a FF is connected to the reset pin of a FF, the FF will be excluded from scanning. In addition, in the right-hand figure in Figure 3-6, it is possible to disable the reset for the FF by tying RST_X to ’1’ and thus it is possible to insert a scan. However, in such cases, faults in the reset lines of the FF cannot be checked. As a result, it is necessary to make it possible to control these circuits from external input ports.

While generally resets are synchronized by inserting multiple stages of FFs as countermeasures for noise or timing problems, in such cases the resets cannot be controlled directly from an external port, and thus measures such as described above are required. The various problems pertaining to resets are described in “3.3.6.DFT in Reset Lines”.