IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 10, Issue 2, Ser. I (Mar. - Apr. 2020), PP 01-07
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Power reduction in SoC platform

Dr. V.Anandi[1], Dr.M.Ramesh[2]
[1]*Associate Professor, ECE Department, M S R I T,Bangalore, India*
[2]*Professor , Management Department, Presidency University, Bangalore, India*
[3] *PG Student, ECE Department , M S R I T,Bangalore, India*
Corresponding Author : Dr.V.Anandi
Associate Professor, Dept of ECE, M S R I T,
Bangalore- 560054,

Abstract: *Physical design is process of transforming RTL netlist into a layout which is manufacture-able [GDS or GDSII]. An efficient physical design flow is typically divided into five major steps: floorplanning, placement, clock tree synthesis, routing and timing closure. Power optimization is always one of the most important design objectives in modern nanometer IC design. System clock signal in electronics product consumes the important part of dynamic power, among them 70% is spent by clock buffers. This critical problem can be optimized using a technique namely clock gating. Recent studies have shown that applying MBFF is an effective means in reducing clock network power. This paper uses MBFF technique to optimize dynamic power and later clock gating on the same design to achieve better results. The aim of this project is to converge the Server SoC Partition and implement low power techniques such as MBFF and clock gating, to reduce power without degrading timing to a great extent.*

Keywords: MBFF; clock gating, dynamic power; power reduction; low power;

Date of Submission: 14-07-2020 Date of Acceptance: 29-07-2020

I. Introduction

Modern System on Chips (SoCs) integrate huge number of transistors in the design and thus power reduction or thermal minimization plays a significant role during the design process. Chips that require high performance are more prone to power related issues. Dynamic power, leakage power and short- circuit power are the three main types of power that are measured during the design process. Dynamic power is the largest among all and is dependent on square of the voltage, clock frequency, capacitance and switching activity factor. Clock circuitry contributes to a large amount of dynamic power due to the switching activity of the clock.
Power consumption has been increasing due to the following factors:

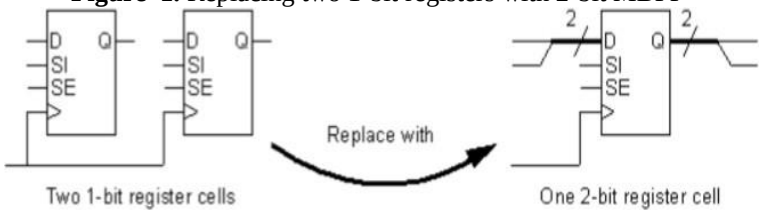
- Gate density increasing rapidly
- Supply voltage and interconnect trace length is reduced at a slower rate.
- Leakage power of the transistors.

There are many techniques that are commonly used to reduce power consumption of a SoC. Buffer sizing, clock gating, register sizing or banking, multi voltage design and voltage scaling are the most commonly used techniques to reduce dynamic power. Power gating has been used recently to reduce leakage power of the design. Recently, Multi-bit flip-flops (MBFF) have been used to reduce power and area.
Multi-bit flip-flop consists of two or more 1-bit flip-flops merged together sharing a common clock driver. Figure. 1 shows how two 1-bit register cells can be replaced with a single 2-bit register cell. Initially MBFF was implemented in the design phase of IC, then during physical synthesis of the design. MBFF can also be implemented in the placement stage of the physical design. Apart from power reduction at cell level, MBFF has many more advantages:

- The number of clock sinks in gate-level netlist with MBFF is much lower which leads to a simpler clock network. The number of clock buffers in the clock tree is reduced thus saving the clock dynamic power.
- MBFF provides efficient clock synthesis due to reduced number of levels thereby reducing the Clock Tree Synthesis (CTS) runtime, skew and latency.
- Reducing the clock and scan nets helps in improving the routing congestion.

Clock gating is another commonly used technique to reduce dynamic power. This technique aims to suppress or disable signal transitions to flip-flops in the clock network under a certain condition, which can be computed using the clock gating circuit. It helps in reducing the switching capacitance of the clock network and the switching activity of the registers as unwanted transitions are not loaded when the clock is inactive.

Figure 1. Replacing two 1-bit registers with 2-bit MBFF

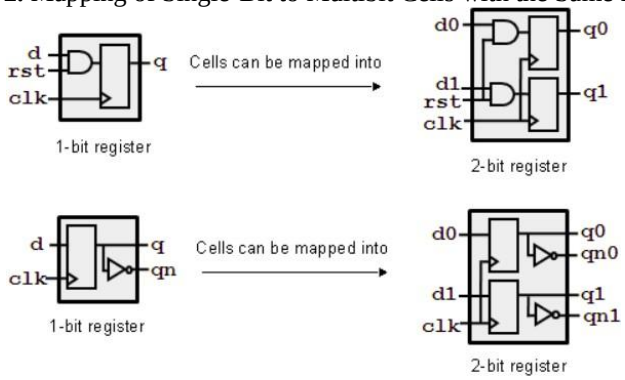


In this paper, we will experiment MBFF implementation in placement stage of physical design and in both synthesis and placement stages of physical design and compare the results. Section 2 will contain the details of MBFF implementation. Section 3 will contain implementation of clock gating post MBFF implementation in order to achieve better results. Section 4 will contain the results and analysis of the experiments and section 5 will conclude this paper.

II. MBFF Implementation

Synthesis and physical implementation tools can organize multiple register bits into groups called "multibit components" in the RTL bus inference flow or "banks" in the placement- aware flow, this process is also known as vectoring. To support multibit register flows, the logic library and physical library must contain both single-bit and multibit library cells, and the multibit cells must meet certain requirements so that the tools can recognize them as functionally equivalent to a group of single-bit cells.

Figure 2. Mapping of Single-Bit to Multibit Cells with the Same I/O Pins



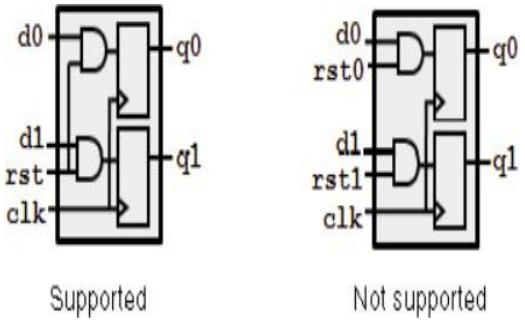
Design Compiler and IC Compiler of Synopsys were the tools used for implementation of MBFF.

2.1: Library requirements for implementing MBFF

(1) To perform mapping from single-bit to multibit registers, the tool checks for matching pin functions and naming conventions in the multibit register pins as shown in Figure 2. For example, if a single-bit register has Q and QN outputs, the tool can replace this register only with a multibit register that also has Q and QN outputs for each output register bit.

(2) The Design Compiler Graphical and IC Compiler tools do not support multibit registers that have a control pin for an individual bit, as shown in Figure. 3.

Fig 3. Multibit Registers with a Control Pin for an Individual Bit Are Not Supported



2.2: Implementation of MBFF

The following procedure was used to implement MBFF:

(1) Exclude the cells from getting swapped with its corresponding multibit version. The cell types include RTL specified cells, scan cells, macros, enable registers, debug cells, and design specific cells.

(2) Set the bounding bow for the pair search.

(3) Set the maximum capacitance difference allowed for clustering the cells.

(4) The size of the merged cell should be controlled based on timing or power requirements. For timing,the smallest cell with C_{max} bigger than the original cell is taken whereas for power the smallest cell with largest C_{max} smaller than the original cell is taken.

(5) The single-bit registers groups should be identified that can be replaced by multibit registers and then modify the netlist accordingly or generate a banking script file which can then be sourced back into the tool to replace single-bit cells with multibit cells. This is achieved using an input mbit-map file.

(6) New multibit cell from a list of registers or latches are created in the current design. All the single-bit cells in list are replaced by one multi-bit cell.

(7) It should be verified that the cells in the list exist in the design and have valid locations, and do not have a "dont_touch" or "fixed" attribute.

(8) The order of the specified cells determines the pin connection order of the new multibit cell. For example, a net connected to the third specified cell in the list will be connected to the third bit of the multibit cell inserted by the command.

(9) If the multi-bit library cell has a larger bit-width than the total bit-width of the specified cells, the pins of the unused bits of the cell are left dangling. If a pin of a specified cell does not have a corresponding pin in the multibit library cell, the pin is disconnected.

(10) For the final merging of cells, the multibit register is created, the single bit cells are disconnected from the nets and then those nets are re-connected to the multibit registers.

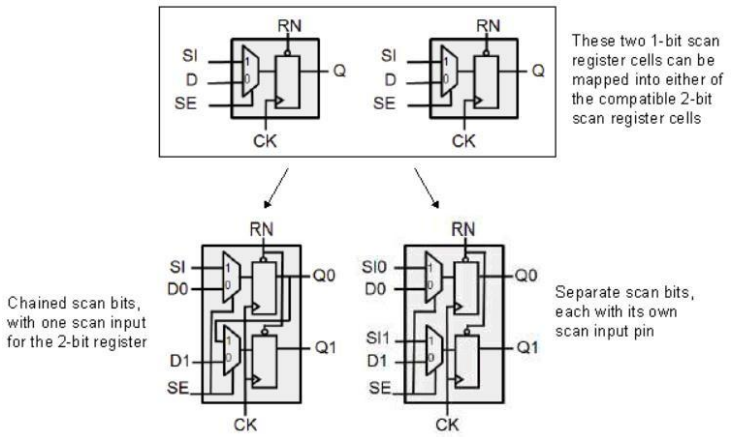
(11) Legalize placement of the multibit flip flops.

In experiment 1, MBFF was implemented only in the placement stage and results were tabulated. In experiment 2, we tried implementing MBFF in both synthesis and placement stage and results were tabulated as shown in Table 1.

Table 1. MBFF Results

(Power in mW)	No Vectoring	Vectoring in Place	Vectoring in syn+Place
Fub A	31	27.5	26.86

Fig 4. Single-Bit Scan Cell and Compatible Multibit Scan Cells



2.3: Commands used for MBFF

(1) Identify_register_banks

The identify_register_banks command assigns the single- bit registers in the design into groups according to the input map file and register group file. It does not actually replace the single-bit registers. Instead, it writes out a banking script file containing create_register_bank commands. To perform register banking and change the design netlist, you need to execute the script file.Figure. 5 shows the full syntax of identify_register_banks command.

Fig 5. Syntax of identify_register_banks command

```
identify_register_banks
  [-input_map_file file_name]
  -output_file file_name
  [-register_group_file file_name]
  [-maximum_flop_count integer]
  [-minimum_flop_count integer]
  [-exclude_instances exclude_cells]
  [-wns_threshold percentage]
  [-wns_threshold_file file_name]
  [-common_net_pins names]
  [-name_prefix prefix]
  [-exclude_library_cells library_cells]
  [-exclude_size_only_flops true | false]
  [-exclude_start_stop_scan_flops true | false]
  [-multibit_components_only]
```

(2) Create_register_bank

To prepare for replacing single-bit registers with multibit registers, the tool writes out a banking script containing create_register_bank commands. You need to execute the script to carry out the actual replacement of single-bit cells with multibit cells. You can insert, delete, and edit the create_register_bank commands in the script file to modify the banking behaviour of the script. You can also execute the create_register_bank command by itself to perform a specific banking task.

III. Clock gating implementation

Although MBFF has reduced the power consumption, we have gone one step ahead to implement latch-based clock gating on the design in which MBFF was implemented in both synthesis and placement stages to optimize the dynamic power to a greater extent. The below steps are used to insert clock gates in the design and perform synthesis on the gated logic:

(1) RTL is read in the form of .v or .sv file.

(2) 'compile_ultra –gate_clock' command is used to compile the gated design.

(3) The clock gate is inserted in the registers qualified for clock-gating during the compilation process. By default, the compile_ultra command uses the set clock gating style command's default settings during clock-gate insertion, and also honors the setup, hold, and other restrictions specified in the technology libraries. Use the set clock gating style command before compiling your design to override the setup and hold values specified in the library.

(4) The scan enable and test mode ports and pins are connected using the 'insert_dft' command.

(5) 'report_clock_gating' command is used to obtain the list of gating cells and registers in the design.

(6) 'report_power' command illustrates the dynamic power of the design post clock-gating implementation.

3.1: Issues in implementing clock gating

- The gating cell must not alter the clock period, it must only turn on or turn off the clock.
- Timing violations that occur after clock gating implementation has to be fixed using any of the general timing fixation techniques as used in general physical design flow.
- Generally, clock skewing/buffering near the endpoint of the data-path is used to fix timing violations.
- If there is a dividing clock in the design, the designer should be careful while implementing clock gating

and not alter the phase of the clock.

- Improper clock gating can cause glitches in the clock waveform which will affect the proper functioning of the circuit.
- Also, major functional problems can occur if the control signal of the clock gate is not designed properly.
- Clock gating implementation is tough when the design is complex and also increases the silicon area of the chip.

IV. Results and Discussion

Fig 6. Layout of design

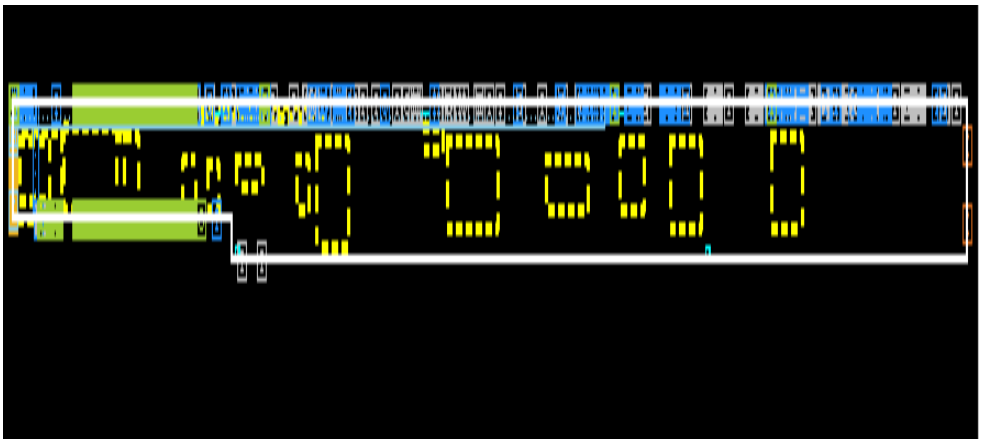
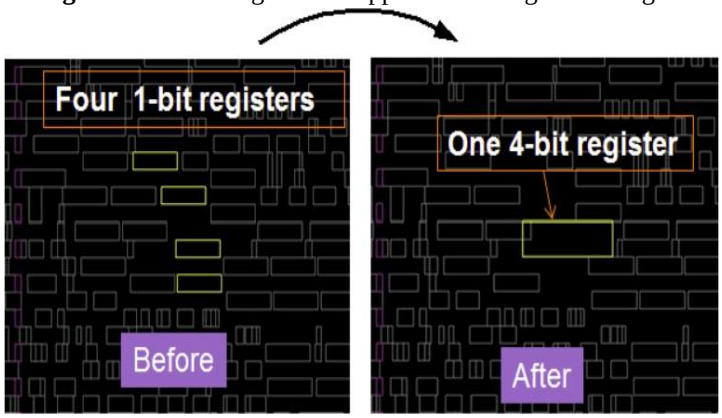


Figure. 6 shows the layout of the design on which experiments were performed. Figure. 7 shows how for 1-bit registers are swapped into a single 4-bit register from the same technology library. Table I shows the results of Multi-bit flip flop implementation. Initially without any power optimization the total power of the design was 31mW. When vectoring (use of MBFF) was implemented only in placement stage of physical design the power was 27.5mW. We had recovered 11.3% of total power. Later in another experiment on the same design, we implemented vectoring in both synthesis and placement stages of physical design and the result obtained was 26.86mW of power i.e. 13.35% of power was recovered.

Fig 7. Four 1-bit registers swapped into a single 4-bit register



Hence we conclude that implementation of MBFF in both synthesis and placement stages will yield better results than implementing only in placement stage. Power and area can be recovered at the cost of increased runtime in the latter case. Figure. 8 shows the report of multi-bit implementation in both synthesis and placement generated using the 'report_multibit' command in IC Compiler. The vectoring percentage of sequential cells, also known as sequential cells banking ratio is 64.93% and that of flops, known as flops banking ratio is 67.03%.

Fig 8. Vectoring percentage

```
icc2_shell> report_multibit
****************************************
Report : report_multibit
Design : chtopaonhn4
Version: M-2016.12-SP5-2
Date   : Thu Jan 17 11:10:46 2019
****************************************
Total Number of Registers:18672
        Number of Single bit Flops:8823
        Number of Single bit Latches:873
        Number of Multibit Flops:8970
        Number of Multibit Latches:6
Total number of single-bit equivalent registers:
        (A) Single bit flops:8823
        (B) Single bit latches:873
        (C) Multibit flops:17940
        (D) Multibit latches:12
        Sequential cells banking ratio
        (C + D) / (A + B + C + D): 64.93%
        Flops banking ratio
        (C) / (A + C):  67.03%
Multibit Register Decomposition
```

Clock gating results are shown in Table 2. It is evident that clock gating efficiency is highest when minimum bitwidth is taken as 2 bits i.e. 2 flip-flops are gated together. If there is a constraint on area, then choosing minimum bitwidth to be 4 is ideal. Clock gating helps to reduce the dynamic power of the design and has minimum impact on leakage power. Slight variation in leakage power values is caused due to types of cells in the technological library. Clock gating efficiency is a percentage which defines the amount of time a register remains gated for a specific switching activity. For an entire design, CGE is calculated as the average of gating efficiencies of all registers. CGE is inversely proportional to the switching activity, thereby reducing the switching activity can help to improve the clock gating efficiency.

Table 2. Clock gating results

Minimum Bitwidth	Clock gating efficiency	Dynamic power (mW)	Leakage Power (mW)	Total Power (mW)
2	81.36	20.83	4.57	25.4
4	79.99	21.53	4.46	25.99
8	77.86	21.9	4.45	26.35

V. Conclusion and future scope

Experimental results on the technology used indicate that MBFF is very effective and efficient method in deep sub microdesign to reduce power, save area and improve routing of clock tree. Amount of power being reduced is ~5 to 15% when the power is between ~25 to 75mW. (It is also design specific).

In this paper, we have implemented MBFF and clock gating on the design. In our design, the power before applying MBFF was 31mW and after applying MBFF in both synthesis and placement, the power is 26.86mW. We have recovered 13.35% of power using MBFF.We have implemented clock gating on the results of MBFF and the power obtained when minimum bit-width was taken as 2 is 25.4mW. Thus implementing MBFF in both synthesis and placement followed by clock gating with minimum bitwidth of 2 bits has recovered 18% of total power.These techniques can be further implemented in complex cores and IPs as well as in partitions where the area constraints are more. MBFF with 4-bits and 8-bits can be used if the cells are available in the technology library. Also, Power gating technique can be used in combination with MBFF in order to achieve better results.

References:

[1]. Lekbir Cherif, Mohamed Chentouf, Jalal Benallal, Mohammed Darmi, Rachid Elgouri, Nabil Hmina "Usage and impact of multi-bit flip-flops low power methodology on physical implementation", 4th International Conference on Optimization and Applications (ICOA), DOI: 10.1109/ICOA.2018.8370498, 26-27 April 2018

[2]. Madhushree and Niju Rajan, "Dynamic Power Optimization Using Look- Ahead Clock Gating Technique", 2nd IEEE International Conference On Recent Trends in Electronics Information & Communication Technology (RTEICT), May 19-20, 2017, India.

[3]. Mark Po-Hung Lin, Chih-Cheng Hsu, and Yu-Chuan Chen, "Clock-Tree Aware Multibit Flip-Flop Generation During Placement for Power Optimization", IEEE Transactions On Computer-Aided Design Of Integrated Circuits And Systems, Vol. 34, No. 2, February 2015

[4]. Jitesh Shinde, S. S. Salankar, "Clock gating — A power optimizing technique for VLSI circuits", Annual IEEE India Conference, DOI: 10.1109/INDCON.2011.6139440, 16-18 Dec. 2011

[5]. Chih-Cheng Hsu, Yu-Chuan Chen, Mark Po-Hung Lin, "In-Placement Clock-Tree Aware Multi-Bit Flip-Flop Generation for

Power Optimization", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Volume: 34 , Issue: 2 , Feb. 2015
[6]. Cristiano Santos, Ricardo Reis, Guilherme Godoi, Marcos Barros, Fabio Duarte, "Multi-bit Flip-flop Usage Impact on Physical Synthesis", 25th Symposium on Integrated Circuits and Systems Design (SBCCI), 2012
[7]. Design Compiler User Guide, B-2008.09 ed., Synopsys Inc., September 2008.
[8]. IC Compiler User Guide: Implementation, M-2016.12-sp4 ed., Synopsys Inc., March 2016

Dr. V.Anandi. " Power reduction in SoC platform." *IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)*, vol. 10, no. 2, 2020, pp. 01-7.