

Examination of HDL coding styles to reduce power consumption for FPGAs

Ryohei Kobayashi[†] Kenji Kise[†][†]Graduate School of Information Science and Engineering
Tokyo Institute of Technology

1 Introduction

The advantages of using Field Programmable Gate Arrays (FPGAs) are to easily change design by users and to reduce development time. Although users can get these benefits, the FPGA has a disadvantage of high power consumption. FPGAs with high power consumption incur high packaging costs, shortens chip life-times, expensive cooling systems, and decreases of system reliability. Therefore, it is truly important to reduce power consumption of FPGAs.

In this paper, we examine the hardware description language (HDL) coding style, which has already been proposed to reduce power consumption of FPGAs, and discuss other effective ways.

2 Prior Work

We describe a prior work [1] about HDL coding styles to reduce power consumption for FPGAs. Minimizing the number of Flip-Flop updates is one of the ways to reduce power consumption, such as not overwriting a register with the value that exactly match the value that is already stored at the register.

Figure 1 shows a circuit of normal coding style. Value from a user logic circuit is stored into D Flip-Flop (DFF). Since the clock input of DFF is connected to the clock signal, the DFF is clocked even when it is not necessary. For example, when the D input has the same value as the Q output ($D = Q$), the DFF does not need to be clocked. This unnecessary Flip-Flop update wastes power.

To address this problem, a new coding style is proposed in the prior work. Figure 2 shows a circuit employing the coding style. The feature of the circuit is same as that of the circuit of normal coding style. The T Flip-Flop (TFF) is clocked only when the Q output and input from the user logic circuit are different value. Thus, since the TFF is clocked only when it is needed to update storage data, it is possible to reduce power consumption.

In addition to this benefit, circuits employing the new coding style can run faster than generated circuits from normal coding style. In generated circuits from normal coding style, the DFF can be clocked properly if its D input is stable at least before its setup time. In circuits employing the new coding style, since the T input of the TFF is always connected to logic '1', the TFF is always ready to be clocked. As a result, circuits employing the new coding style can run faster than generated circuits from normal coding style.

In the prior work, the Microelectronic Center of North Carolina (MCNC) benchmark circuits [2] are used to evaluate the new coding style. Each MCNC benchmark circuit is converted into VHDL files to represent the new coding style and normal coding style.

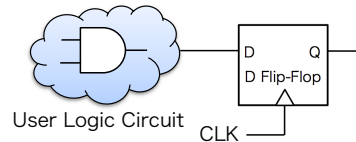


Figure 1: A circuit of normal coding style.

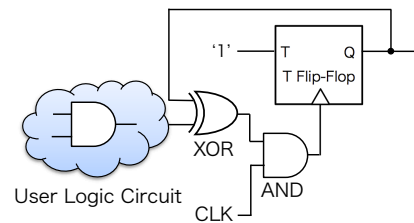


Figure 2: A circuit employing the new coding style in the prior work.

The circuits are implemented on an FPGA (Altera Stratix EP1S10F484C5) using an FPGA tool (Altera Quartus II 6.0). The new coding style reduces total power consumption by 13-90% and runs 2-20% faster compared to normal coding style.

3 Examination of Prior Work

In order to confirm benefits of the prior work, we examine it using test circuits shown in Figure 3. Figure 3a shows a normal n-bit counter circuit, and Figure 3b show an n-bit counter circuit employing the new coding style of the prior work. In this examination, we write 1000-bit ($n=1000$) counter circuits with and without the new coding style in Verilog HDL, and we measure power consumption of two 1000-bit counters. As shown in Figure 3b, note that the TFF consists of one XOR and one DFF because TFF is not provided as FPGA primitives in the used FPGA for this examination.

We use Xilinx ISE 14.7 for logic synthesis for written circuits, DIGILENT Atlys Board with FPGA Xilinx Spartan-6 XCS6SLX45 for implementation of these circuits. We set FPGA frequency as 120MHz. To measure power consumption of these circuits, we use Adept2.13.1 software system (32/64-bit Windows) shown in Figure 4. This tool can measure power consumption of 3.3V, 2.5V, 1.8V, and 1.2V supplies. Since power consumption of the test circuits correspond to that of 1.2V, we measure power consumption of the test circuits driven by 1.2V supply.

Figure 5 shows power consumption of the test circuits. "Normal" represents power consumption of the normal two 1000-bit counters, and "Employing prior

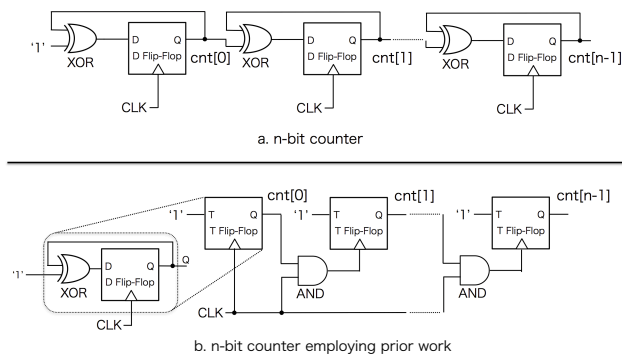


Figure 3: Test circuits with and without the new coding style in the prior work.



Figure 4: Measurement of power consumption by Adept2.13.1 software system.

work” represents power consumption of the two 1000-bit counters with the new coding style. The normal circuit consumes 213mW, the circuit employing the prior work consumes 564mW and 165% higher than Normal. In the next section, we discuss the reason why the new coding style does not achieve lower power consumption.

4 Discussion and Summary

Implemented circuits on Xilinx FPGA Spartan-6 consist of configurable logic blocks (CLBs) and switch matrixes connecting between CLBs. A CLB has logic cells, called “Slices”, and each slice has Lookup Tables (LUTs), Flip-Flops (FFs), and single clock input port [3]. LUTs are used to realize combinational logic circuits, and FFs are used to realize sequential logic circuits.

In our test circuit with the new coding style, different clock per FF is fed the clock input. Since each slice has single clock input port, the Xilinx logic synthesis tool cannot place the FFs fed with such different clocks in the same slice. In fact, the occupied slice ratio of Employ prior work is 29 % while that of Normal is 11%. This causes the increase of the number of the occupied slices. The more the number of switch matrixes increases, the higher the power consumption of the FPGA is. Therefore, it is better to use coarse-grain approach like feeding different clock per 32-bit register rather than to use fine-grain approach like feeding dif-

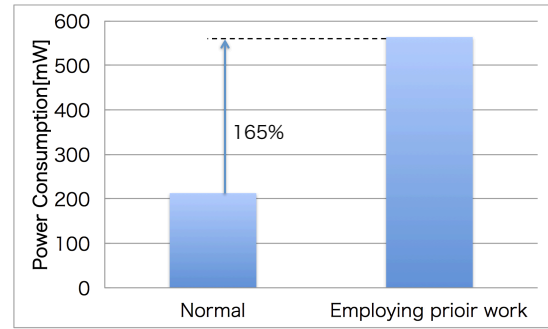


Figure 5: Power consumption of the normal two 1000-bit counters and the two 1000-bit counters employing prior work.

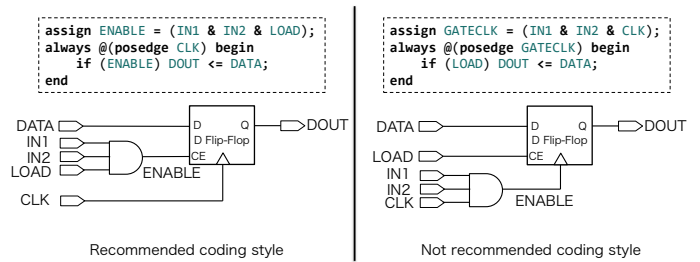


Figure 6: Recommended and not recommended coding styles.

ferent clock per FF.

It is not necessarily good to gate the clock port, because gated clocks can increase clock delay, clock skew, and cause glitches and other undesirable effects. From these reasons, Xilinx recommends using the dedicated clock-enable (CE) port instead of gating the clock port [4]. Figure 6 shows recommended and not recommended coding styles by Xilinx. Using the clock enable can improve timing characteristics and analysis of the design. Therefore, it is promising to use clock enable.

In the future work, it is crucial to seek better HDL coding styles based on obtained findings from this examination.

References

- [1] Thomas Marconi, Dimitris Theodoropoulos, Koen Bertels, and Georgi Gaydadjiev. A Novel HDL Coding Style to Reduce Power Consumption for Reconfigurable Devices. In Jinian Bian, Qiang Zhou, Peter Athanas, Yajun Ha, and Kang Zhao, editors, *FPT*, pp. 295–299. IEEE, 2010.
- [2] Saeyang Yang. Logic Synthesis and Optimization Benchmarks User Guide Version 3.0, 1991.
- [3] 7 Series FPGAs Configurable Logic Block User Guide. http://www.xilinx.com/support/documentation/user_guides/ug474.7Series.CLB.pdf.
- [4] HDL Coding Practices to Accelerate Design Performance. <http://www.eng.utah.edu/~cs3710/xilinx-docs/wp231.pdf>.