VHDL Based Parallel/Serial Conversion

Qiqi Gong[1,a,*], Yueze Gu[1,b] and Yuanyuan Zhang[2,c]

[1]School of Electronic and Information Engineering, Beijing Jiaotong University
[2]Institute of NBC Defense, PLA Army
a. gongqiqi@bjtu.edu.cn, b. guyueze@bjtu.edu.cn, c. yoyozhang-mail@163.com
**corresponding author: Qiqi Gong*

Keywords: VHDL, parallel/serial conversion, FPGA.

Abstract:In this paper, the method of designing a process-observable 8-bit parallel/serial conversion system is introduced. Parallel/Serial conversion is a common concept in data processing and communication area. However, with the upgrading of FPGA, quicker process frequency is required and basic chips used to conduct parallel/serial conversion seem to fall behind due to low frequency and unreliable data transmission. Sometimes, the process frequency should be adjusted while the global clock has been given certain frequency. Also, there is a lack of research into VHDL based conversion system. The parallel/serial conversion system designed in this paper is composed of three parts: a 32-bit counter used to adjust clock frequency, a parallel-to-serial converter based on synchronous D flip-flops with an En pin and a serial-to-parallel converter. Reliable data transmission is realized by a periodic impulse Fn. When the transmission is completed, it sends out a high level signal to indicate compliment. Implementation of the system is conducted in VHDL. Simulations and physical experiments verify the rationality of the system.

1. Introduction

VHDL(Very High Speed Integrated Circuit Hardware Description Language) is a hardware description language used in electronic design automation to describe digital and mixed-signal systems such as FPGAs and ICs. It has been the industrial standard language in digital system designs.[1] In FPGA, it is a common practice to conduct data processing through parallel/serial conversion as data come in serially and stored in parallel method.[4] Additionally, it seems that data transmission inside is usually conducted in parallel method but the communications among devices need data transferred serially. In common practice, conversion of parallel and serial data is conducted by 74165 and 74HC595. The maximum working frequency of 74165 series is 36MHz. However, with the upgrading of FPGA, the processing frequency is becoming higher[5] and the current processing frequency of parallel/serial conversion could be hard to adapt new technologies. Additionally, 74165 and 74HC595 do not guarantee the reliable transmission of data. Also, frequency could be required to adjusted in different situations, and given frequency of global clock signal could be inconvenient.This paper proposed an 8-bit parallel/serial conversion system based on VHDL. It is composed of a 32-bit counter used to adjust frequency of clock signal of given FPGA. Analysis and method of designing each component is introduced in Section 3 and Section 4 displays the experiment results by conducting computer simulation and programming FPGA.

2. System Overview

The 8-bit conversion system is composed of a 32-bit counter, Parallel-to-Serial Converter and Serial-to-Parallel Converter. Each component is programmed with VHDL and generated on QuartusII software. Integrate each component together to form the whole system. Thus, the system has three input: a global clock(any selected bit of the counter), an En pin and an 8-bit wide data bus inputting the data in parallel along with an output: an 8-bit wide data bus outputting the data in parallel. At the rising edge of the clock signal, if the En signal is on high-level, external data will be loaded into the system otherwise data will be shifted. Clock signal works with a period of about 2.68 seconds which can be easier to control and observe on FPGA board. The serial data converted by the Parallel-to-Serial module act as the input of the Serial-to-Parallel module. The two converters work synchronously, using the global clock signal as their own clock. When the conversion is finished, the parallel-to-serial send a pulse signal to the serial-to-parallel converter to inform the completion. Figure 1 gives an overview of the whole system.

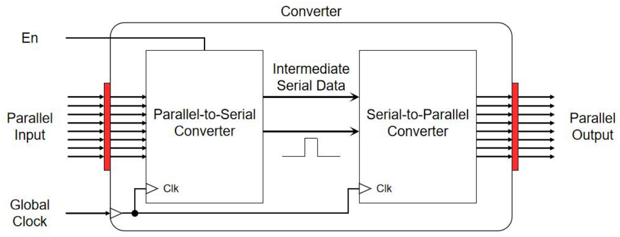


Figure 1: Block diagram of parallel/serial system.

3. Method of Design and Implementation

3.1. Counter

Counter is used to adjust the frequency of the clock signal in the system. It uses the embedded clock of FPGA whose frequency is 50MHz as its own clock signal. Its output is a 32-bit vector and becomes 0x00000000 after reaching maximum 0x11111111. When the 26th bit taken as the clock signal, high-level time *t*≈1.34s thus *T*=2.68s which is acceptable. Additionally, a quicker clock using the 24th bit can be chosen with a multiplexer. Figure 2 displays the structure of the counter.

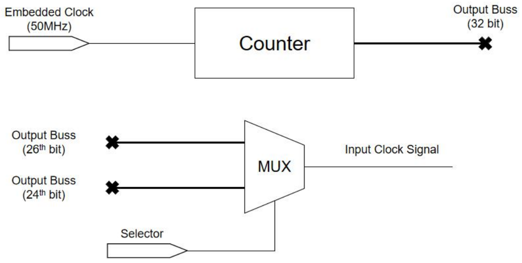
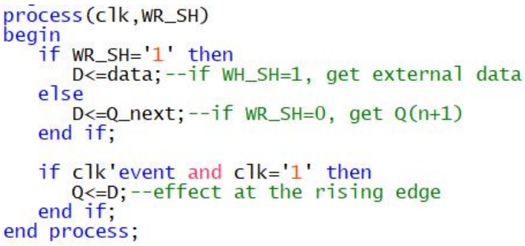


Figure 2: Structure of the counter.

3.2. Parallel-to-Serial Converter

Parallel-to-Serial Converter is a shift register with an En pin. It is a combination of eight synchronous D flip flops which are effective at the rising edge of the clock and load data externally if the En signal is on high level or data will be shifted. A a component named "*D_flipflop.vhd*" is created firstly and piece of code is shown in Figure 3.



```
process(clk,WR_SH)
begin
   if WR_SH='1' then
      D<=data;--if WH_SH=1, get external data
   else
      D<=Q_next;--if WR_SH=0, get Q(n+1)
   end if;

   if clk'event and clk='1' then
      Q<=D;--effect at the rising edge
   end if;
end process;
```

Figure 3: Code piece of D_flipflop.

Declare intermediate signal to connect D flip-flops in the main program. Use RTL viewer tool to check the structure of parallel-to-serial converter which is shown in Figure 4.

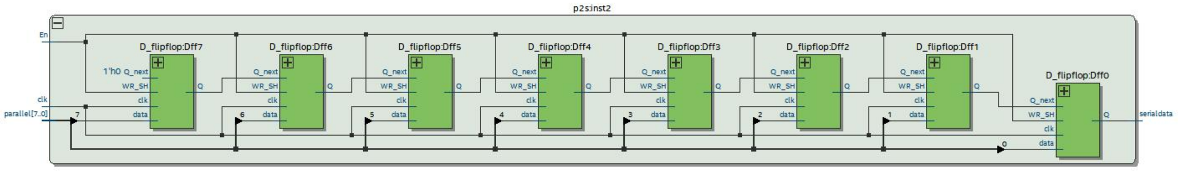


Figure 4: RTL viewer of parallel-to-serial converter.

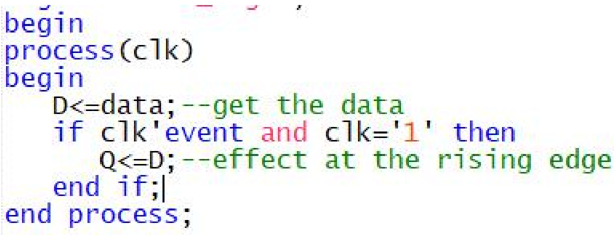
3.3. Serial-to-Parallel Converter

Serial-to-Parallel Converter is composed of 8 simpler D flip-flops. The output of the n-th D flip flop also acts as the input of the (n-1)-th D flip flop. Serial data are shifted one bit at the rising edge of the clock in each cycle, so the datum will be shifted to the LSB after 8 cycles. Table1 gives an example of the process(suppose the input data is 10100110).

Table 1: Example of serial-to-Parallel conversion.

Cycle	Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Q_0
1	0	X	X	X	X	X	X	X
2	1	0	X	X	X	X	X	X
3	1	1	0	X	X	X	X	X
4	0	1	1	0	X	X	X	X
5	0	0	1	1	0	X	X	X
6	1	0	0	1	1	0	X	X
7	0	1	0	0	1	1	0	X
8	1	0	1	0	0	1	1	0

Only clock signal is included in the sensitive list of D flip flops, effective at the rising edge of clock signal, used in this converter . Code piece is displayed in Figure 5.



```
begin
process(clk)
begin
   D<=data;--get the data
   if clk'event and clk='1' then
      Q<=D;--effect at the rising edge
   end if;
end process;
```

Figure 5: Code piece of D flip flops.

3.4. Parallel/Serial Conversion

Combination of the system is implemented in a *bdf* file which is created as the main file of the project, allowing to connect components in a diagram. The counter is a 32-bit counter, whose 24th and 26th bit can be selected as the clock signal, convenient to observe the system's physical performance on FPGA board. Figure 6 displays the method to connect each module. Pins named "clk_led" and "serialoutput" are also designed to help observe.

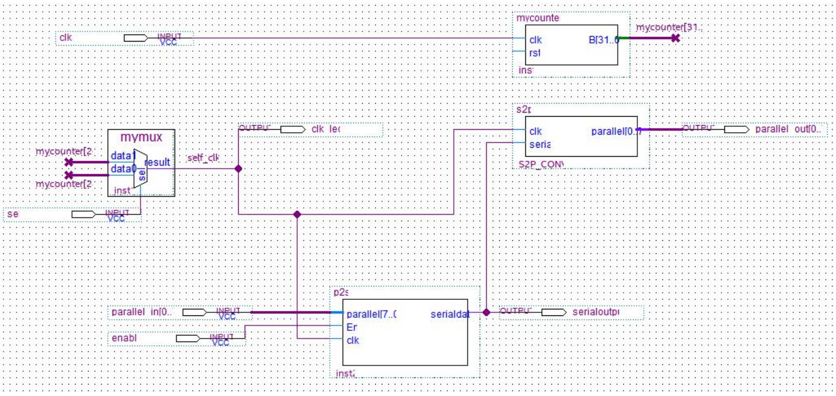


Figure 6: Block diagram of parallel/serial conversion.

4. Experiment Results

Simulation is conducted in ModelSim. ModelSim is an HDL simulation environment supporting various languages such as VHDL, Verilog and System C. Simulation is performed using the graphical user interface in ModelSim.[2,3] It seems a common practice to create testbench files to conduct simulations and in this project, a testbench file is created to conduct simulation as verification. Figure 7 displays the simulated result which is in accordance with theoretical analysis. It can be seen that data are loaded into the converter at the rising edge of the first cycle due to the high-level of En and in the next following 8 cycles, data are transferred to serial-to-parallel converter bit by bit. At the 9th cycle, intermediate serial datum shows D_7's state at the 2nd cycle. However, no datum was loaded then, so the it is unknown and the state is transferred to serial-to-parallel converter at the 10th data, so D_7 is unknown then. During the low level of the 10th cycle, Fn gives an impulse to indicate the compliment of data conversion then, the system can send the converted data out and receive next group of data. The transmission frequency is 100MHz.

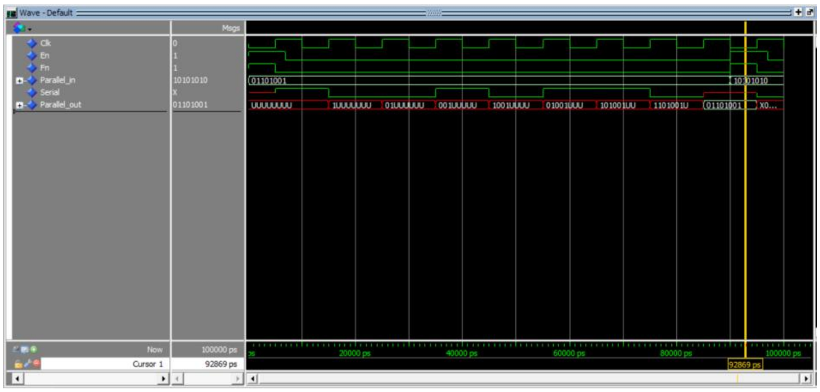


Figure 7: Simulated result.

Download the program to FPGA board, observing the conversion process more directly. Figure 8 is the initial state of the system and the relevant pins or outputs has been noted. Set the input data when En is at low-level and set En on high-level before the rising edge of the clock signal.Figure 9(a) is converting D5, and SerialOutput's led is on. Figure 9(b) is the completed moment of conversion.

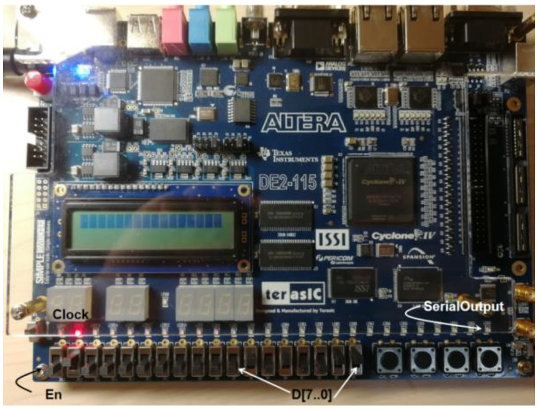
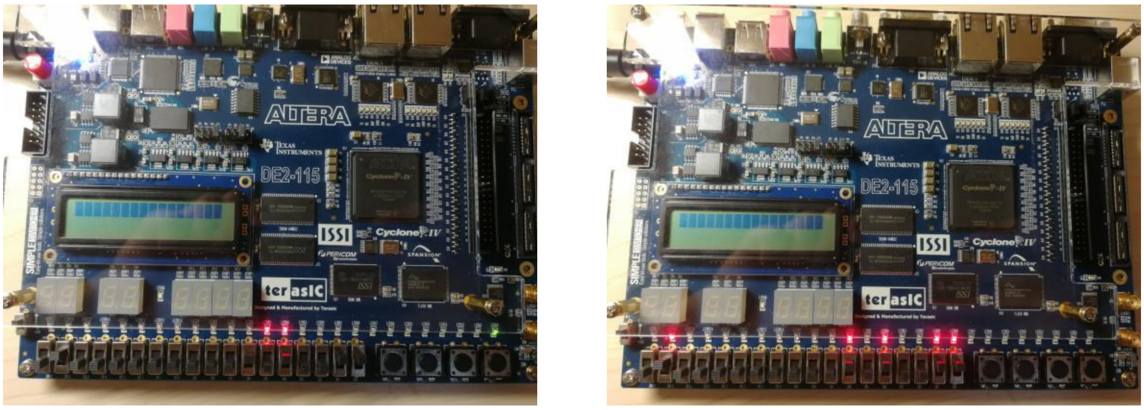


Figure 8: Initial state.



(a)

(b)

Figure 9: Converting process.

5. Conclusions

Parallel/Serial conversion is a kind of technology widely used in FPGA and current chips to conduct conversions seem to fall behind. This paper propose a frequency adjustable parallel/serial conversion system which is composed of a 32-bit counter, parallel-to-serial converter with a synchronous control pin En and a serial-to-parallel converter based on VHDL. Reliable data transmission is promised by the periodic impulse of Fn pin. Data can also be grouped by the receiver according to the Fn impulse. However, in future application of the system, the frequency of En signal should be carefully designed in case of raising errors in transmissions.

References

[1] Guoping Wang. Teaching Sequential VHDL Models by Synthesis and Examples. 2005.

[2] Tom Granberg. Handbook of digital techniques for highspeed design. 2004.

[3] White and Randall L. Reconfigurable embedded control systems. ACM SIGSOFT Software Engineering Notes, 36(5):53.

[4] Peter Wilson. Design Recipes for FPGAs. 2013.

[5] Xilinx Design Flow for Intel FPGA and Soc Users. 2018.