Design and Realization of Multi-port Register File based on Schematic

Lihua JIANG
Dept. of Computer, Wuhan Polytechnic University, Wuhan Hubei, China
Shigao LI
Dept. of Computer, Wuhan Polytechnic University, Wuhan Hubei,China

ABSTRACT: In cpu design, the register file is a necessary device which save the instruction and data. In this paper, we propose a design method for multi-port register file design in the environment of single-cycle CPU system based on the MIPS instruction set and according to the characteristics of multi-port register file, and the Schematic is introduced in order to speed up the development cycle. Furthermore, we discuss the consideration and operational principle of design and realization in detail. The simulation results for the part constructed by FPGA are also presented. In this example which provides the learning and design innovation for other circuit designed.
KEYWORD: Multi-port register file; Schematic; Programmable Logic Device; FPGA

1 INTRODUCTION

With the development of EDA technology, the circuit design based on programmable logic device is simple and convenient, and is applied more and more widely[1]. Use Schematic combining FPGA device can greatly facilitate the design of the digital integrated circuit[2]. Among the single-cycle CPU System based on the MIPS instruction set, the multi-port register file is an important medium-scale device, and is the core of the RISC micro-processor, the reading and writing of all internal and external data have directly relation with it, It consists of a group of registers, if only the register number of the multi-port register file is given, the register content can be read or written.

In the single-cycle CPU System of the MIPS instruction set, which defines 32 32-bits general registers and three special registers. Because the three special registers have more specific use, which don't put them in the register file. In the MIPS architecture R type instructions require three register operands, so every R type instruction needs read two datas from the register file, and write a data to the register file. To every data being read from the register file, which will need a input to the register file, the input can sure to the number register being read, and needs a output from the register file too, the output will carry out the data being read from the register file. If which wants to write a data, which needs two inputs, one input will sure the number register being written, another will offer the data to the register being written. In despite of the register number in a "reading" register end, the register file always output its contents. However, the writing action must be controlled by writing control signal, and data is written on the edge of the cycle[3]. So the total needs seven inputs (a data, three register numbers, a clock signal, a reset signal and a writing permit), two outputs (both datas), namely the register file has one writing chunnel and two reading chunnels. The width of the register number is 5 bits (2^5=32), as shown in Figure 1, so 32 registers can be accessed, but the data bus width of one input and two outputs are 32 bits.

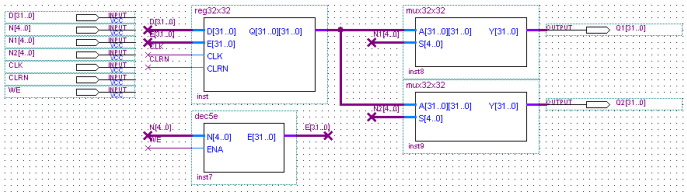

Figure 1. the register file's top level Schematic

2 DESIGN AND IMPLEMENTATION OF THE REGISTER FILE

Based on the MIPS32 instruction set still used widely at present, this paper design a register file with 32 32-bits general registers.

In 32 general registers there has two given special features:

- R0 is connected to 0. When reading R0, will always be 0; when writing R0, no action, no effect. So, if some instruction need 0 as one operand, R0 is used as the source data register.
- R31 is used as default purpose register when instruction such as JAL, BLTZAL, BLTZALL, BGEZAL, BGEZALL do not specify the purpose register.R31 is used as general register.

Of course, according to the need to increase or decrease and extend the register number, for example, the designing method of the register file composed of 64 registers is same, only the scale is lagrer. Of course, according to the need to increase or decrease and extend the register number, for example, the designing method of the register file composed of 64 registers is same, only the scale is lagrer.

2.1 *Use Schematic to Design the Register File*

Schematic editor input method is similar to the traditional method of electronic design, that is, to draw the circuit diagram that can complete specific function by EDA software graphic editor interface. Schematic diagram consist of the logic device (symbols) and connection wire, the logic device in diagram may be pre-function module in EDA software library, such as the and gate, trigger, a variety of function modules including 74 series device[3]. Even there has the similarly IP function module.

EDA software QuartusII provides a powerful, intuitive, convenient and flexible operation schematic design function, At the same time it is equipped with a variety of richer component library for the various needs, and it also provides the multi-level design function of the schematic input, so that users can design the more large-scale circuit system[4].

The register file makes up of five modules with three levels, the module under lowermost level is D trigger, the middle level module includes five bits address coder, 32-bits register and 32-to-1 multiplexer, the upper level module is the multi-port register file module[5]. Design adopts the method combining the act of modeling and structure, first using the way of the act modeling to establish the module under lowermost level, then using the way of the structure modeling to establish the higher circuit design, the following is which all the modules under level are introducted separatily.

2.2 *References Address Coder Model*

A coder is an important device in the combination logical circuit, if 32 registers need be accessed, which will change five input into 25, that is multi-output ends with 32 outputs, in all input changing combinations, each output is one which only is once, so output ends are the combination which is the input variable smallest.

Because the writing action of the register file must be controlled by writing control signal, the coder with ENA enable input end may be designed, just which make ENA enable input end as a writing control signal, that is when ENA equal 0, the coder do not work, its outputs are 0; when ENA equal 1, the coder works normally, which can carry out writing operation to the register file. Also which names the enable input end as high-level availability when input end ENA equal 1 and it works normally[6]. The following is the logical expression based on logic truth table, in which N[4..0] is the number register of the writing chunnel, ENA is the writing control signal, E[31..0] is the register of the writing chunnel.

$$E[0]=ENA \cdot \overline{E[0]} \cdot \overline{E[1]} \cdot \overline{E[2]} \cdot \overline{E[3]} \cdot \overline{E[4]}$$

$$E[1]=ENA \cdot E[0] \cdot \overline{E[1]} \cdot \overline{E[2]} \cdot \overline{E[3]} \cdot \overline{E[4]}$$

……

$$E[30]=ENA \cdot \overline{E[0]} \cdot E[1] \cdot E[2] \cdot E[3] \cdot E[4]$$

$$E[31]=ENA \cdot E[0] \cdot E[1] \cdot E[2] \cdot E[3] \cdot E[4]$$

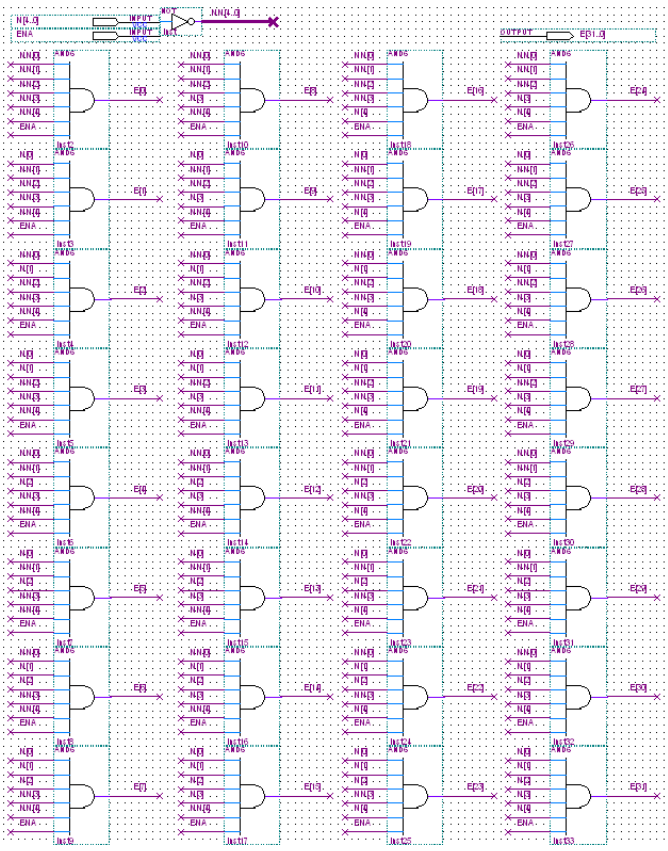

Figure 2. the five bits address coder's Schematic

According to the upper logical expression, which can get the schematic of the five bits address coder, as shown in Figure2. Selecting menu File Create/ Update, to click the Create Symbol Files for Current

item, the current file dec5e. bdf can become a component symbol which is saved (the default is dec5e.bsf) [7]. This is only the current opened file, in order to being loaded in the high-level design, note: that conversion component must exist in the path folder for the current project.

2.3 *32-Bits Register* Model

Just as its name implies, the general registers may be used according to the user wishes, to modify their values generally do not have a great influence to the computer running, the address in the register may point to the position in the memory, that is finding address, and also can read and write data to computer peripheral devices.

Register is the important part of the central processor unit, and it has a very high reading and writing speed, so it is very quickly among register transmitting data. Register is the limited high-speed memory, it can be used to temporarily hold instructions, datas and addres. Each register has a name R0 to R31, but not have similarly memory's address numbers.

Register commonly do not store a lot of data, and these data are also often change, before new datas are written, old datas are retained invariability, so which use the low levels D trigger to realize 32-bits register[8].

In order to finish 32-bits register, which must first of all design 8-bits trigger. Therefore, eight asynchronous reset and synchronous enable D-triggers are used. The eight 1-bit D-triggers share one enable input, one clock input and one reset signal. Every setting signal connect high level VCC, which means that D-triggers may deposit data from input, the input and output of eight D-triggers respectively regard as every single bit of eight bits data, which schematic file is dffe8.bdf, similarly create the component symbol dffe8.bsf.

Calling four dffe8.bsf component symbols in the new schematic file, four 8-bits D-triggers share one enable input, one clock input and one reset signal. Data's input and output respectively regard as four 8-bits of the 32-bits data, which schematic file is dffe32. bdf, similarly create the component symbol dffe32.bsf.

Using 32 32-bits D-triggers dffe32.bsf to design 32 32-bits registers, this schematic reg32x32.bdf is shown in Figure 3.Similarly create the component symbol reg32x32.bsf, in order to being loaded in the high-level design.

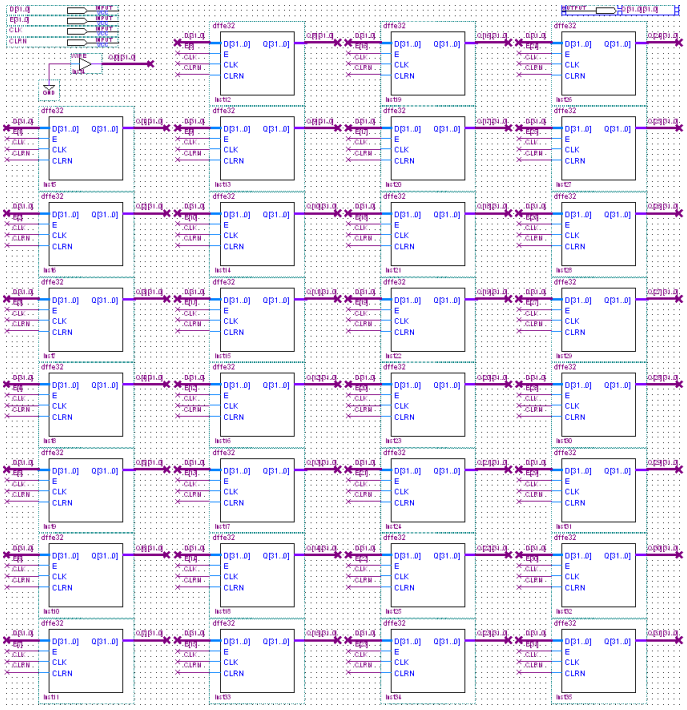

Figure 3. the 32 32-bits Registers's Schematic

2.4 *32-to-1* Multiplexer Model

In addition, the LPM(Library Parameterized Modules) in QuartusII library functions can also be applied to design 32-to-1 multiplexer, which is macro function module library being parameterized. Application of these functional module libraries can greatly enhance the efficiency of IC (Integrated Circuit) design. Entering LPM_MUX in the importing component's text box of the schematic, You can see LPM_MUX Module, then set the module's "how many 'data' inputs do you want? ",that is, data input lines is 32(2^5), this time selection control code will be automatically turned into five (S[4..0]). Setting the module's "how wide should the 'data' input and the 'result' output buses be?", that is, input and output data width is 32, the width of its corresponding data automatically is turned into 0-31, Other settings are default. Results gained the 32-to-1 multiplexer mux32x32.bdf, Similarly create the component symbol mux32x32.bsf.

The above design modules are instantiated in the register file's top level Schematic, which can achieve the multi-port register file.

3 ACHIEVE THE SIMULATION AND DOWN L -OAD OF REGISTER FILE

Simulation is simulate EDA design according to certain algorithm and simulation library. Here using functional simulation, the benefit is that the design time is short.

After the engineering design file is correct, we just establish the Vector Waveform File vregfile 32x32. vwf; then we can run menu Processing

Generate Functional Simulation Netlist, to generate functional netlist; Finally, we select menu Processing Simulator Tool, Figure 4 shows the simulation waveform results.

If the register file through the Quartus II simulation is correct, the file will be compiled. The downloaded specific device is the Cyclone list EP1C6Q240 Model of the Altera company, using QuartusII to process circuit synthesis and program download. After the device pins are setted, the progam must be compiled again, Ultimately creating the target file which can be downloaded. Then connecting the corresponding pins of the device setting and the drive circuit of the LED, finally the target file is downloaded the device, and the LED shows the read and write results of the register file. The tests shows that the actual circuit work result is consistent with the simulation result, which meet the design requirement.

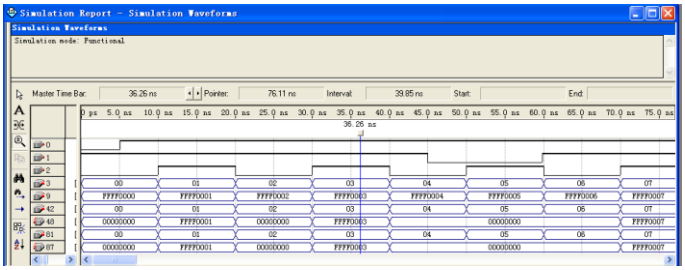

Figure 4. the register file's waveform simulation report

4 CONCLUSION

VHDL description logic function of the text input is strong, it can describe the top-level design, the project described by this way can achieve the final performance which has great relational with the designer's level, experience, and integrated software. the EDA technology development in a certain degree realize hardware design softwarely[9].

Currently Schematic input is one of the important EDA design, its integrated software request is not high, which use the ready-made circuit with sophisticated IP core and small and medium-scale integrated circuit[10], the hardware work speed and chip utilization are high[11].

This paper sufficiently use the QuartusII's powerful edit feature to ensure the design correctness of register file, in the single-cycle CPU System based on the MIPS instruction set, designed register file is applied successfully, and it has good stability.

ACKNOWLEDGEMENTS

The paper is supported in part by the Natural Science Foundation of China under Grant No.61201452.

REFERENCES

[1] LIU Yu-xiao etc. EDA technology and VHDL circuit development application practice, Beijing: The Press of electronics, 2009.

[2] LUO Li-fan etc.FPGA development based on VHDL induction quickly, skill and instance, Beijing: The Press of posts &telecom, 2009.

[3] LIU Zhao-han. VHDL chip design, Beijing: The Press of Tsinghua University, 2006.

[4] XIN Jian-pin, ZENG Fan-tai. VHDL Programming tutorial (the third edition), Beijing: The Press of Tsinghua University, 2005.

[5] WANG Xu-ying, LU Ying-hua, ZHANG Li-kun. Design and Implemention of high-speed real-time data acquisition system based on FPGA. The Jounal of China Universities of Posts and Telecommunicaions, 2008, 13(4):61-66.

[6] Ma Qungang, Yang Yintang, Li Yuejin. New Design Method for the Optimal Energy Delay Product of FPGA Based on the Interconnect. Journal ofXidian University, 2009, 31(1): 32-35.

[7] Wang P, Franca JE. Multirate switched-capacitor circuits for 2-D signal processing. Kluwer Acadmic Publishers, 1998.

[8] IEEE Standard VHDL Language Reference Manual. IEEE SUT, 2009, 1067-1987.

[9] ROMAN L.FANK V A Study of the speedups and competitiveness of FPGA Soft Processor Cores using Dynamic Hardware/Software Partitoning, 2010.

[10] Volnei A. Pedroni-Circuit Design with VHDL. MIT Press, 2009.

[11] CHENG C C, HUANG C T. Multiple-lifting scheme: memory-efficent VLSI implementation for line-based 2-D DWT: Proceedings of International Symposium on Circuits and Systems, Kobe, Japan, May 2005. Kobe, 2009: 23-26.