
Stepping Motor Subdividing Drive System Design Based on FPGA

Zongzuo Yu*, Gaizhi Guo

College of Computer and Information Engineering, Inner Mongolia Normal University, No.81, Zhaowuda
Road, Huhhot, China, 0471-4392451

*Corresponding author, e-mail: yuzz@imnu.edu.cn

Abstract

The design idea to implement multi-subdividing drive system for stepping motor based on Field-Programmable Gate Array chip was proposed in this paper. On a piece of EPF10K10LC84-15, the system module was accomplished using Very-High-Speed Integrated Circuit Hardware Description Language and graph editor. The simulation results were shown on the platform MAX+plus^{II}. This method could control the stepper motor subdivision without external digital to analog converter, which made the circuit structure be greatly simplified. And the experiment results proved that this method was also with high control precision and good effect, so it was much better than the Micro Control Unit control method.

Keywords: *stepping-motor, multi-subdividing, PWM*

Copyright © 2013 Universitas Ahmad Dahlan. All rights reserved.

1. Introduction

As a kind of electrical impulses-angular displacement transformation components, stepping motor has many advantages such as inexpensive, easy control, and no error accumulation. So it has been widely used in machinery, instrument, and industrial control. Stepping motor subdivision drive technology is developed in the middle of the 1970's, can greatly improve its comprehensive performance. This technique could reduce the angle of the steps, improve the stability of the motor running, increase control flexibility [1].

At present, the Micro Control Unit (MCU) is usually used for the stepping motor subdivision drive control. This method requires additional D/A converter, the circuit structure is relatively complex, the control accuracy is relatively low, and it is not flexible enough. Based on this, this paper adopts the EDA design method, realizes a complete stepping motor multi-subdivision drive control system on FPGA chip EPF10K10LC84-15 [2]. This method can simplify the control circuit, improve accuracy and flexibility of the step angle control, and achieve a better control effect.

2. Design Principle

2.1. The Principle of Stepping Motor Multi-subdivision Drive

When each of the excitation windings of the stepping motor is energized in turn, the direction of synthesized inner magnetic field changes so that the stepping motor rotates. For four-phase motor, when each of the four phases (A, B, C, D) is energized, the magnetic field rotates. In general, when the internal magnetic field changes 360°, the rotor of the motor is turned a pitch.

To realize multi-subdivision of the stepping angle, the current of the stepping motor excitation winding must be controlled to be ladder up or down, that is, there must be multiple intermediate stable states between 0 to the maximum current. So the corresponding magnetic field vector amplitude has multiple intermediate state, and the direction of the synthesized magnetic field of the adjacent two or heterogeneous phases has multiple stable intermediate states [3]. When four-phase motor's angle is divided by 16, the current of each phase rises or falls with 1/8 steps. There are 15 stable states inserted between phase A and B, one step is displaced by 16 steps, realizes the 16-subdivision of step angle.

2.2. Pulse Width Modulation Subdivision Driving Circuit

In this paper, we use the pulse width modulation (PWM) multi-subdivision driving circuit to control stepping motor. Its working principle is that, calculate the current value through each phase winding according to the required step angle, store the value in ROM embedded in FPGA, then output corresponding data to D/A converter according to different address, the output controlling voltage of D/A converter will be the input of pulse width modulation circuit, pulse width modulation circuit will convert the entered the controlling voltage to the rectangular wave of corresponding width, control the on/off time of power amplifier tube, change the average current of the motor winding, realize the step motor subdivision.

The key of the PWM subdivision drive is the pulse width modulation, it modulates a given voltage signal to a mainly continuous signal [4]. Generally the angular speed fluctuation is proportional to the step angle, and inversely proportional to the multi-subdivision number.

Compared with other subdivision driving circuit, PWM subdivision driving circuit is of high control accuracy and high working frequency stability, but the circuit is more complex. So it is mainly used in higher comprehensive drive performance occasions.

2.3. The Structure of the Step Subdivision System

In this paper, we use the four-phase stepping motor 16-subdivision as an example to expound the complemented method of PWM subdivision drive control system of stepping motor on FPGA chip. Figure 1 is the current waveform of four-phase motor 16-subdivision, A, B, C, D are the four phase of the motor. In general, there are always two phases' windings are powered, one phase's current increases stepwise, another decreases gradually. For one step angle, the current changes 16 steps. That is the motor position is subdivided into 16 small angles. This is the working principle of the 16-subdivision of a step angle [5].

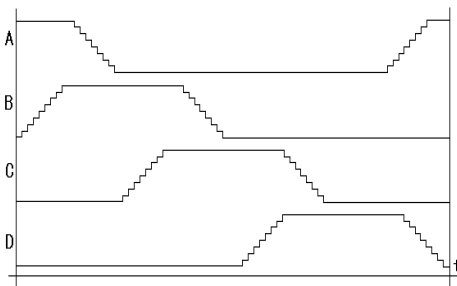


Figure 1. The Current Waveform for Sixteen Division of Four-phase Motor

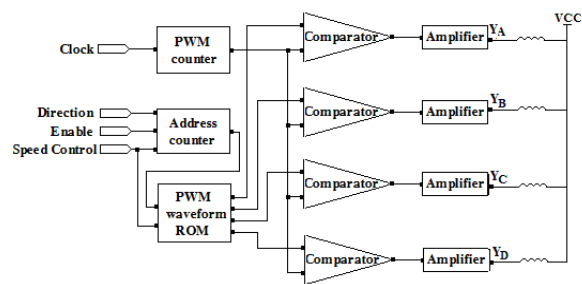


Figure 2. The Structure of Subdividing Drive System for Four-phase Motor

Figure 2 shows the system structure of stepping motor multi-subdivision drive. The PWM counter produces a periodic sawtooth wave as a rising ladder, it is loaded to one input end of each digital comparator; The output data of the ROM of PWM waveform is loaded to the other end of each digital comparator. When the output of PWM counter is less than the output data of ROM, the comparator outputs high level; otherwise the comparator outputs low level. So the output is the periodic PWM waveform. This kind of PWM signal controls on/off of each power tube drive circuit, it changes with the data in ROM, changes the duty cycle of the output signal, then changes the length of power-on time of the motor winding, and finally changes the average value of the current in the winding, realizes the multi-subdivision control.

3. Design Method

In this paper, we use the principle graph editor and hardware description language (VHDL) to realize a complete stepping motor multi-subdivision drive control system on FPGA chip EPF10K10LC84-15 [6, 7]. The platform is Altera's software MAX+plus□.

3.1. The Sixteen Subdivision Drive Module

The subdivision drive module (named “dianji”) is designed based on Figure 2. On MAX+plusII, we use the principle diagram editorial method to realize it [8]. Figure 3 is the circuit principle diagram of the module.

CNT16 is the PWM counter with three outputs, can produce a ladder wave with 8-level; CNT64 is a 64-counter, as the address counter of PWM waveform ROM. They are realized by the programming language VHDL [9].

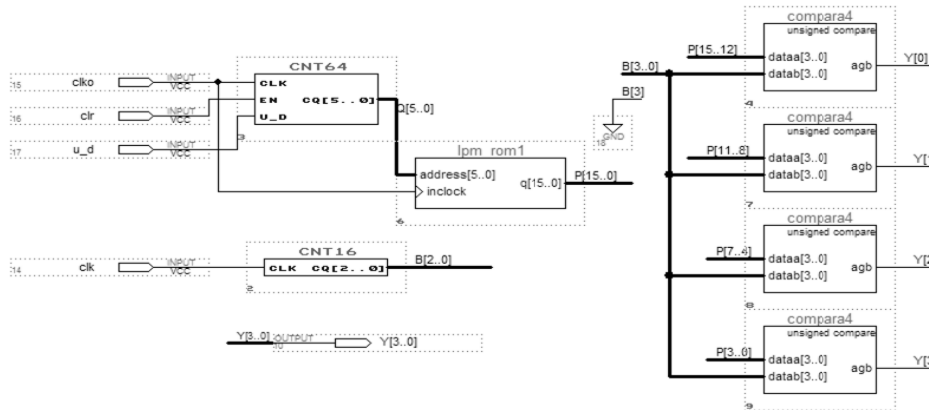


Figure 3. The Principle Diagram of Sixteen Division Drive Module

Compara4 is the 4-bit data comparison circuit; lpm_rom1 is the PWM waveform ROM. They are realized by LPM [10]. The configuration data file of lpm_rom1 is called “pwm_rom.mif”, this file is realized by text editor, and the program is as follows:

```
width=16; depth=64; -- Waveform data is 16-bit, a total of 64
address_radix=hex; data_radix=hex; -- Address and data are all hexadecimal
content begin -- The waveform data in ROM
0:8000;1:8100;2:8200;3:8300;4:8400;5:8500;6:8600;7:8700;8:8800;9:7800;a:6800;b:5800;
c:4800;d:3800;e:2800;f:1800;10:0800;11:0810;12:0820;13:0830;14:0840;15:0850;16:0860;
17:0870;18:0880;19:0780;1a:0680;1b:0580;1c:0480;1d:0380;1e:0280;1f:0180;20:0080;21:0081;
22:0082;23:0083;24:0084;25:0085;26:0086;27:0087;28:0088;29:0078;2a:0068;2b:0058;
2c:0048;2d:0038;2e:0028;2f:0018;30:0008;31:1008;32:2008;33:3008;34:4008;35:5008;36:6008;
37:7008;38:8008;39:8007;3a:8006;3b:8005;3c:8004;3d:8003;3e:8002;3f:8001;
end;
```

3.2. The Speed Control Module

The speed control module (named “crea_clk”) controls the motor run in “start-constant acceleration-constant speed- uniform deceleration-stop” mode, it consists of five sub-modules, as shown in Figure 4.

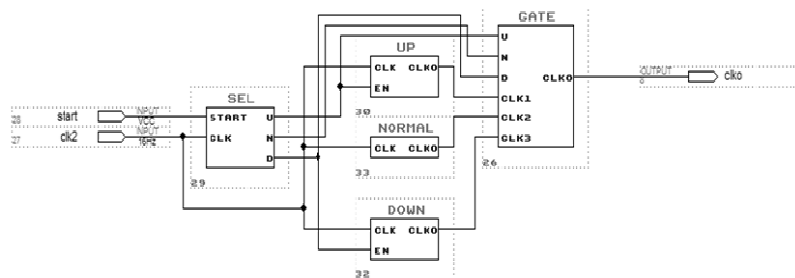


Figure 4. The Speed Control Module

SEL is the time control module, when detecting the rising edge of “start” signal, each of its output ends “U, N, D” will output certain time high level, as the control signal of subsequent modules to control the length of constant acceleration, constant speed and uniform deceleration paragraphs; UP、NORMAL and DOWN are respectively the clock modules of constant acceleration, constant speed and uniform deceleration; GATE is the clock on/off control block to choose one clock signal according to the states of “U, N and D” as the address translation clock signal of ROM. These five modules are all realized by programming with VHDL. For example, the program of the module “UP” is as follows:

```
entity up is
port(clk:in std_logic; en:in std_logic; clko:out std_logic);
end entity;
architecture behav of up is
signal f:integer range 0 to 11 :=11;
signal n:integer range 0 to 11 :=11;
signal cl:std_logic;
begin
process(clk,en)
begin
if clk'event and clk='1' then
if en='1' then
if f=n then f<=0; cl<=not cl;
if n>0 then n<=n-1;
else n<=11;
end if;
else f<=f+1;
end if;
end if;
end if;
end process;
clko<=cl;
end behav;
```

3.3. The Top Module

The top module is composed of the above two modules (“dianji” and “crea_clk”), it is shown in Figure 5.

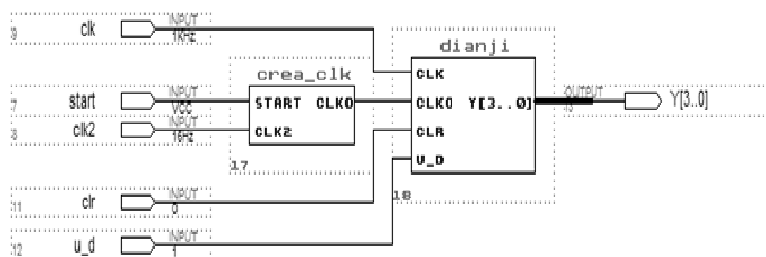


Figure 5. The Top Module

“clk” and “clk2” are two clock input signals respectively for 1 KHz and 16 Hz; “start” is the input end of start signal; “clr” is the input end of reset signal; “u_d” is the rotation direction control end; “Y[3..0]” are the output ends, can output four control signals simultaneously, respectively connect to the four windings control terminals of stepping motor to control the length of the winding power-on time.

4. Results and Analysis

The simulation results are obtained on MAX+plus platform. Figure 6 shows the simulation result (0 to 2.4s) of the sixteen division drive module. Under the proper input, the four output ends of this module can output periodic pulse width modulation signals. If they are added to the external drive circuits, these signals will control on/off of each external drive circuit, and change the duty cycle of the output signal, then change the length of power-on time of the motor winding, finally change the average value of the current in the winding, realize the multi-subdivision control. Because the original simulation result is too large, this figure is only a short period of it, from 0 to 2.4s. In this figure, we can see the changes of the two signals-“Y0” and “Y1”, and the changes accord with our theoretical analysis.

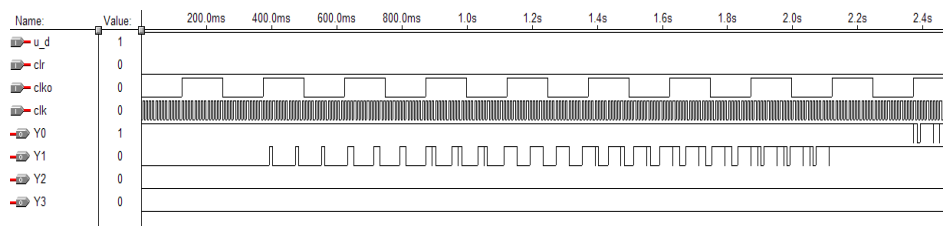


Figure 6. The 0~2.4s Simulation Result of the Sixteen Division Drive Module

Figure 7 shows the simulation result (4s to 17s) of the speed control module. In this figure, we can see that with the input signals-“start” and “clk2”, the pulse width of the output signal “clk0” automatically changes. If this output signal is used as the input clock signal of the waveform ROM address counter in the sixteen subdivision drive module, it will control the motor automatically run in “start-constant acceleration-constant speed- uniform deceleration-stop” mode.

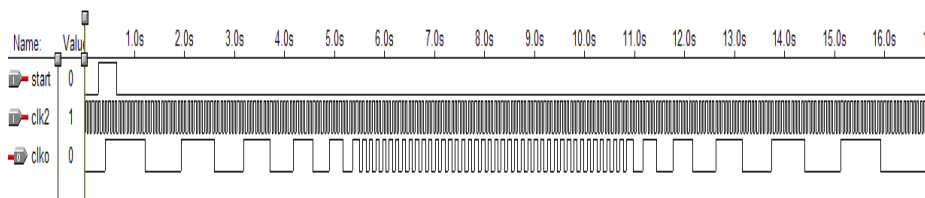


Figure 7. The 4s~17s Simulation Result of the Speed Control Module

Figure 8 shows the simulation result (6.6s to 7.0s) of the top module. It is the result of the complete stepping motor subdividing drive system. It is only a part of the original simulation figure. Here, we can see a period of the changes of output ends in the system. These output signals can realize precise control for the motor angle.

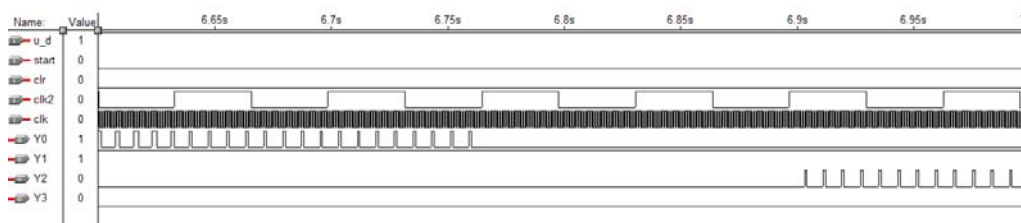


Figure 8. The 6.6s~7.0s Simulation Result of the System

5. Conclusion

Used modular design method, a completed four-phase stepping motor 16-subdivision drive control system is accomplished using VHDL and graph editor on a piece of FPGA chip EPF10K10LC84-15. It can control the motor run in “start-constant acceleration-constant speed-uniform deceleration -stop” mode. The simulation results of each sub-modules and the whole system are shown on MAX+plus^{II}. The experimental results indicate that using FPGA to control stepping motor can realize precise control of the motor angle, and needs no external D/A converter, greatly simplifies the control circuit. This method is simple and with high precision.

References

- [1] Takashi Kenjo, Akira Sugawara. Stepping Motors and Their Microprocessor Controls. Oxford: Clarendon Press. 1995.
- [2] Song P, Jiye H. EDA Technology and Practical Tutorial. Second Edition. Beijing: Science Press. 2005.
- [3] Xigeng M, Fuyu L, Yang L, Hong L. Design of Power Supply for Subdivision Control of Stepping Motor with Transference Syntonization. *Journal of the University of Petroleum. China (Edition of Natural Science)*. 2000; 24(6): 77-79.
- [4] Shahrin Md Ayob, Zaenal Salam, Abdul Halim M. Yatim. Non-Sinusoidal PWM Method for Cascaded Multilevel Inverter. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2012; 10(4): 670-679.
- [5] Idkhajine L, Monmasson E, Naouar W, Prata A, Bouallaga K. Fully Integrated FPGA-Based Controller for Synchronous Motor Drive. *IEEE transactions on industrial electronics*. 2009; 10: 4006-4017.
- [6] Ligong Wang. Based on PLC Stepping Motor Travel Control. *Combination machine tools and automatic processing technology*. 2008; 44-46.
- [7] Li Zhaohui, Shen Gang. Design of Digital Pulse Width Modulator Based on CPLD. *Electrical Drive Automation*. 2004; 6(3): 7-9.
- [8] Guangyou Yang, Zhijian Ye, Yurong Pan, Zhiyan Ma. The Implementation of S-curve Acceleration and Deceleration Using FPGA. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(1): 279-286.
- [9] William Kleitz. Digital Electronics with VHDL (Quartus II Version). London: Prentice Hall. 2006.
- [10] Danli X, Xiaohong Z, Yong W. Design of ROM Look-up Table for Arbitrary Waveform. *Modern Electronic Technology*. 2008; 31(11): 172-173.