A Study of Bezier Curve Used in CNC Based on DSP and FPGA

Ya-Wen Li*, Cai-Zhen Zhang

School of Electronic and Information Engineering, Lanzhou Jiao tong University

Lanzhou, 730070, Gansu, China

wydxxzj@163.com;liyawen0519@gmail.com

Received 29 December 2014; Revised 10 February 2015; Revised 25 March 2015; Revised 23 April 2015;

Revised 14 May 2015;Accepted 1 June 2015

Abstract: For the features and requirement of high-speed, high-precision, flexibility of modern CNC (Computer numerical control) machine, combined efficient data processing DSP (digital signal processor)capabilities with high-speed, parallel processing and reconfigurable characteristics of FPGA (Field Programmable Gate Array).We designed a two-stage interpolation control system by using the DSP to do the rough interpolation, then FPGA finish the accurate interpolation. Using the Bezier algorithm in three-dimensional CNC machining, and achieve any curve trajectory in space, expand the CNC trajectory from two-dimensional space to three-dimensional space and achieve high accuracy and good reliability, reduce system costs by combining with the use of DSP and FPGA .The system has a good computing ability, flexible architecture, good modular design, strong real-time, scalable and versatile features.

Keywords: CNC, Bezier curve, FPGA, Interpolation, DSP.

1 Introduction

Since the 1990s, Europe, America, Japan and other countries are scrambling to develop a new generation of high-speed CNC machine. With the key technology solution in range of technical fields such as ultra-high-speed cutting mechanism, ultra-hard cutting tool, long life materials and abrasives, large power high-speed spindle, high acceleration / deceleration linear motor driven feed components and high-performance control system (including monitoring systems) and protective devices etc provide the technical basis for the development and application of a new generation of high-speed CNC machine. The world's industrial powers direction is the development of precise machining, ultra-precise machining, and one of the core idea of modern manufacturing technology is a flexible manufacturing, and reconfigurable nature of CNC technology[1-2], which is a design method that depend on the application requirements to change their architecture[3].

In modern CNC applications, from the users' points of view, the most used linear and circular interpolation, due to the limitation of the algorithm itself, if need to machine arbitrary curve, it need for manual writing long G code to approach the target curve. From the hardware implementation of arithmetic, traditional motion control device adopts single-chip microcomputer directly or a single DSP to realize control, which peripheral circuit is complex, and has slow speed of calculation, real-time interpolation speed and precision are not ideal. From the study of interpolation algorithm, Wang Youmin et al[4] describes the application of Bezier curve interpolation in CNC systems, which derived the continuous conditions of curve segments at the junction both first and second derivation, but **have**n't considered the effect of acceleration, and only realized the algorithm in two dimension. Ye Bosheng et al[5] proposed interpolation parameter increment is constant during the process of interpolation, calculation is simple, but in the interpolation process, it will produce dramatic fluctuations of speed which affect the surface quality of the work piece. Yeh and Hsu et al[6] explained the feed rate chord error algorithm that is based on adaptive constraint. This algorithm calculate the maximum allowable speed according to the maximum error allowed by the system, and then compares with the current command feed rate, and choose the lower to be the feed speed to prevent chord error which exceeds the limitation**.** Consider all above, in this paper, interpolation control system combined with DSP and FPGA[7], the interpolation process achieved by hardware and two step of interpolation. DSP do the coarse interpolation, and then FPGA finish the accurate interpolation, communication between DSP and FPGA use the way of structure two FIFO in the FPGA for data communication. Since using the Bezier curve interpolation algorithm extend from two-dimensional space to three-dimensional space, and adding interpolation speed control. The hardware control system could achieve the path of any curve in space.

* Corresponding author：Ya-Wen Li, wydxxzj@163.com.

This design not only reduces the costs, and guarantees the high speed and precision, but also reflects the flexible design.

2 Design of Hardware System

Currently, most CNC interpolation devices use single-core controllers but single-core controllers have some limitations like its structure. The ordinary MCU (Micro Control Unit) device can be used as a controller but it limited its ability to control and the speed is very slow, so it is difficult to use it in high-speed control devices. By using DSP processor, the processing speed of the algorithm can be guaranteed, but the underlying interpolation speed and parallel processing cannot be achieved. If the algorithm was written directly in FPGA, by using FPGA we can control the interpolation, though it can ensure the interpolation speed, but takes more internal resources of FPGA, which increases the design cost. The system in this paper is designed by using DSP and FPGA hardware structure, which can finish the two-stage interpolation processing while letting the rough interpolation in DSP, and FPGA do the accurate interpolation. This system can assure the algorithm processing and underlying interpolation speed, and the two-stage interpolation could raise the precision of interpolation and system has a good performance.

The system is divided into four modules: the PC interface control module; DSP circuit module; FPGA circuit module and communication module. PC control interface is written using visual C ++, running on a PC to control the lower interpolation controller to finish the work and use USB for communication between the upper and lower machine[8]. DSP get the control instruction which was sent by PC, and did the coarse implementation of the Bezier curve interpolation algorithm. The feedback signal of encoder was sent by FPGA to control the speed and position of the motor, between the FPGA and control signals ports, using optocoupler to isolate them to protect the FPGA chip, and by using cooling liquid to prevent the overheating temperature damaging CNC device tool in CNC machining process. In the last, FPGA finish accurate interpolation, show output pulses and direction signal to each axis of servomotor. The block diagram of hardware system is given below in Fig.1.

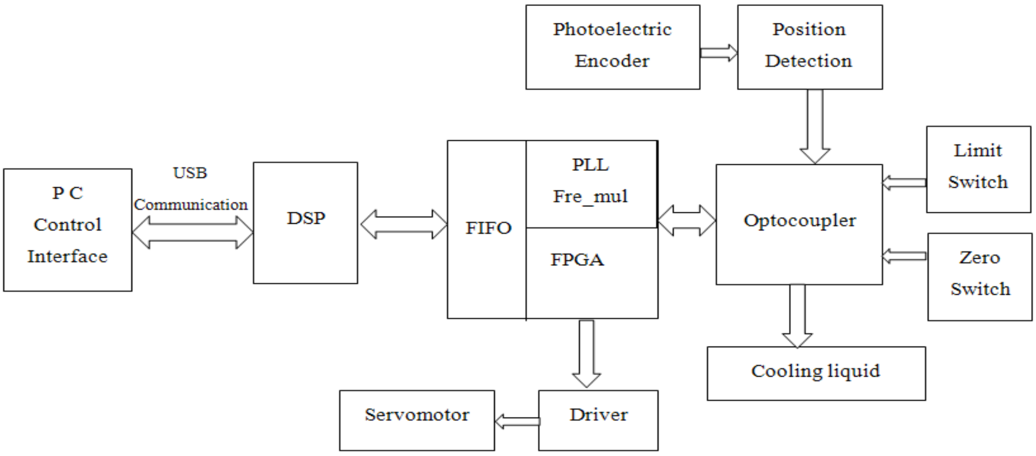

Fig.1. Block diagram of hardware system

2.1 DSP and Peripheral Interface Circuit

The DSP chip chosen for the curve algorithm is TMS320F28335, it is a series of floating-point DSP of TI's TMS320C28X controller, and we chose it as a control chip due its high precision, low cost, low power consumption, high performance, highly integrated peripherals, clock frequency is up to 150MHz, with 32 for the floating point capability, six DMA channels supporting ADC, McBSP, Epwm, XINTF and SARAM, 256K * 16 size of FLASH program SARAM memory and 34K * 16 size, PWM output with a 18-way, using 1.9V core voltage and 3.3V supply voltage peripheral interface[9].

In this system, DSP is the master processor as compared to the FPGA, its main advantage is reflected in the complexity of the algorithm and floating-point operations[10], so DSP deal with algorithm in the system, and on the one hand its function is communicate with PC via USB and receives data and order to deal with the algorithm and the corresponding operations, on the other hand, DSP communicate with FPGA, read status and position information based on FPGA, meanwhile send the interpolated data to the FPGA to complete the accurate interpolation process.

2.2 FPGA and Its External Interface Circuit

The FPGA chip chosen to apply on the curve interpolation controller is EP4CE10E22C8, one of the CycloneⅣ E FPGA of Altera, and it has 10320 LEs, 46 elements Embedded multiplier, 2 PLL, up to 423936 bits of RAM, and more than 92 programmable I/Os[11]. Using the EDA tool QuartusⅡ 13.0 to write the Verilog HDL hardware description language and TestBench file in FPGA， and the timing simulation is done in ModelSim-Altera 10.1d semulator by using TestBench file.

FPGA has good real-time operations as compare to DSP; its internal logic unit can be freely programmed. It can be programmed to simulate a variety of special signal processing chip, and has obvious advantages in a large amount of data parallel processing at the bottom. In the system, FPGA is mainly used in motion controller to achieve a variety of interfaces. The function of FPGA in system is to communicates with DSP, receive the data and control commands, output the corresponding value of switch, pulses to the driver of servomotor, pass the position information of encoder, status information of limit and zero switch to let DSP judges the information and control the system.

2.3 Module of Communication Circuit

The system has two communication modules, one is the communication between PC and DSP by using the USB; the second one is the communication between DSP and FPGA, which is the structure of two FIFOs in FPGA for communication.

1) Communication between PC and DSP

The way of communication between PC and DSP is USB communication in system, CY7C68013A was chosen as the USB chip in this system, which provides 480Mb / s transfer rate, with its internal FIFO buffer has four endpoints. We can configure the EPxFIFOCFG(internal part of CY7C68013A) appropriately for two times, three times, four times buffer. This design can decrease delay time for the data packet to minimum, thereby increasing bandwidth throughput. The chip has three working modes, normal port mode, FPIF master mode and FIFO slave mode[12]. In this paper, CY7C68013A working in FIFO slave mode, DSP read the internal FIFO directly, When the DSP detects a non-null signal, set read enable signal and the read signal to low, and read out a 16-bit signal at each rising edge of the clock signal, when the detected signal is empty, stop reading. The connection circuit diagram for USB is given below in Fig.2.

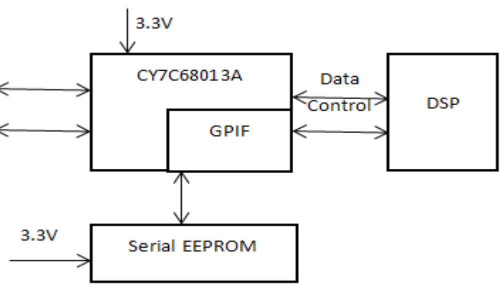

Fig. 2. Circuit module of communication

2) Communication between DSP and FPGA

Asynchronous FIFO can quickly and easily transfer real-time data between two different clock systems. Using different asynchronous clocks of FIFO; Asynchronous FIFO can generally be divided into four parts, the write address, read address, data storage and the part of empty / full flag generation. The specific implementation process in the FIFO is: at the rising edge of writing clock, when the writing enable sign is effective, write the data to the corresponding position of dual port RAM, at the rising edge of read clock, when the reading enable sign is effective, always output the data in read address of the corresponding dual-port RAM to the read data bus. The empty / full flag generation is generated by compared the read and write address. The core issue of design is to judge the FIFO is empty or full.

3 Design of Bezier interpolation algorithm

Bezier curve are very important parametric curve in computer graphics, many researchers use it in a two-dimensional CNC machining, and some people used two-dimensional applications in Bezier CNC machining. But in this paper, we use Bezier curve[4] in 3D CNC machining to solve some of the irregular curves or surfaces in CNC machining processing problems.

3.1 Cubic Bezier Curve

According to the general formula of Bezier curves:

$$P_{\mathrm{i}}^{k} = \begin{cases} P_{\mathrm{i}} & k=0 \\ (1-t)P_i^{k-1} + tP_{i+1}^{k-1} & k=1,2,\cdots,n-k \end{cases} \tag{1}$$

Equation (1) can be used for the cubic Bezier curve expression equation (2), assume P_0, P_1, P_2, P_3 four points, define a cubic Bezier curve in three-dimensional space. Curve step from P_0 to P_1, and came from the direction of P_2 to arrive at P_3. But it will not pass P_1 or P_2; the two points just provide directional information. The distance between P_0 and P_1 determines the length of curve turned to P_2 before approaching P_3, n is the parameter, P_0, P_1, P_2, P_3 is a set of vector points.

$$P(u) = [u^3\, u^2\, \mathrm{u}\, 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix} \tag{2}$$

3.2 The interpolation process

This paper uses the data sampling interpolation[13], which is approximate the trajectory by the feed chord length in fixed period. The interpolation principle is: according to the feed speed to calculate the feed rate in every interpolation period, do coarse interpolation, and then calculate the increment of each linkage axis in every interpolation cycle, then transform the increment into the value of uniform motion to achieve accurate interpolation.

According to the data sampling interpolation principle, the feed increment each axis is:

$$\begin{cases} \Delta\overrightarrow{P_{\mathrm{i}}} = \overrightarrow{P_{i+1}} - \overrightarrow{P_i} \\ \Delta L_i = \left|\overrightarrow{\Delta P_i}\right| \end{cases} \tag{3}$$

The ΔLi is the step length of feed rate, while the interpolation period of CNC system is T, the current feed rate is V_i, then $\Delta Li = V_i * \mathrm{T}$.

To obtain next interpolation point, calculate the component of each axis, so as to get the new interpolation point, the component of x axis is:

$$\mathrm{x}(u_i) = \frac{\Delta x_i}{\Delta u_i} \Rightarrow \Delta x_i = x(u_i) * \Delta u_i \tag{4}$$

$$\Delta L = \sqrt{\Delta \mathrm{x}_i^2 + \Delta \mathrm{y}_i^2 + \Delta \mathrm{z}_i^2} \tag{5}$$

Put the formula (4) into (5):

$$\Delta \mathrm{u}_i = \frac{VT}{\sqrt{(x(u_i))^2 + (y(u_i))^2 + (z(u_i))^2}} \tag{6}$$

The components of y and z axis are the same as x axis, when the next interpolation period is coming, $u_{i+1}=u_i+\Delta u_i$, then can get the coordinates of the interpolation points.

3.3 3D Bezier interpolation process

By formula (2) can get the component form of Bezier curve:

$$\begin{cases} \mathrm{x}(u_i) = (1-u_i)^3 x_0 + 3u_i(1-u_i)^2 x_1 + 3u_i^2(1-u_i)x_2 + u_i^3 x_3 \\ \mathrm{y}(u_i) = (1-u_i)^3 y_0 + 3u_i(1-u_i)^2 y_1 + 3u_i^2(1-u_i)y_2 + u_i^3 y_3 \\ \mathrm{z}(u_i) = (1-u_i)^3 z_0 + 3u_i(1-u_i)^2 z_1 + 3u_i^2(1-u_i)z_2 + u_i^3 z_3 \end{cases} \tag{7}$$

Using formula (7) to get the derivation of "u_i":

$$\begin{cases} \mathrm{x}'(u_i) = \frac{\mathrm{d}x(u_i)}{du_i} = (-3x_0 + 3x_1) + (6x_0 - 12x_1 + 6x_2)u_i + (-3x_0 + 9x_1 - 9x_2 + 3x_3)u_i^2 \\ \mathrm{y}'(u_i) = \frac{\mathrm{d}y(u_i)}{du_i} = (-3y_0 + 3y_1) + (6y_0 - 12y_1 + 6y_2)u_i + (-3y_0 + 9y_1 - 9y_2 + 3y_3)u_i^2 \\ \mathrm{z}'(u_i) = \frac{\mathrm{d}z(u_i)}{du_i} = (-3z_0 + 3z_1) + (6z_0 - 12z_1 + 6z_2)u_i + (-3z_0 + 9z_1 - 9z_2 + 3z_3)u_i^2 \end{cases} \tag{8}$$

Using first derivative to get the small increase of *x*, *y*, *z*, ignore the high order of *u*, get the feed rate component of each axis.

$$\begin{cases} \Delta \mathrm{x}_i = x'(u_i)\Delta u_i = [(-3x_0+3x_1) + (6x_0 - 12x_1 + 6x_2)u_i]\Delta u_i \\ \Delta \mathrm{y}_i = y'(u_i)\Delta u_i = [(-3y_0+3y_1) + (6y_0 - 12y_1 + 6y_2)u_i]\Delta u_i \\ \Delta \mathrm{z}_i = z'(u_i)\Delta u_i = [(-3z_0+3z_1) + (6z_0 - 12z_1 + 6z_2)u_i]\Delta u_i \end{cases} \tag{9}$$

Put the formula (8), (9) into (5):

$$\Delta \mathrm{u_i} \approx \frac{VT}{3\sqrt{\begin{aligned}&(x_1 - x_0)^2 + 4(x_1 - x_0)(x_0 - 2x_1 + x_2)u_i + (y_1 - y_0)^2 \\ &+ 4(y_1 - y_0)(y_0 - 2y_1 + y_2)u_i + (z_1 - z_0)^2 + 4(z_1 - z_0)(z_0 - 2z_1 + z_2)u_i\end{aligned}}} \tag{10}$$

Using Δu_i can obtain new interpolation point:

$$\begin{cases} \mathrm{x}(u_{i+1}) = \mathrm{x}(u_i + \Delta u_i) \\ \mathrm{y}(u_{i+1}) = y(u_i + \Delta u_i) \\ \mathrm{z}(u_{i+1}) = z(u_i + \Delta u_i) \end{cases} \tag{11}$$

4 MATLAB Simulation of Bezier Curve Interpolation in Space

According to the algorithm, the interpolation points are all on curve, there is no radial error. Contour interpolation error from feeding step size to bring short straight approach for the actual curve and small parameter variation Δu_{j} formula chord error δ is created by giving up high order volume[13]], According to the principle of interpolation, ideal path is a part of arc, and the actual interpolation trajectory is a section of chord, as Fig.3 shown.

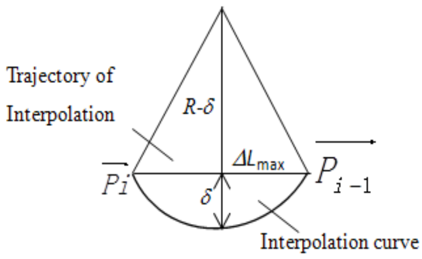

Fig.3. Contour control of interpolation curve

Assume curve curvature is R, interpolation feed step isΔL, so only in accordance with the feed rate, it generate the interpolated path interpolation cycle that cannot give the actual results, but it creates the contour control error, regard the curve as an approximation arc. The formula for R (radius) for curvature is:

$$R = \frac{\sqrt{(x'^2 + y'^2)^3}}{x''y' - x'y''} \tag{12}$$

The chord error is δ, curve radius of curvature is R, the relationship between chord error and the step sizes ΔL[13] is:

$$\delta = R - \sqrt{R^2 - \frac{\Delta L^2}{4}} \tag{13}$$

Since R is fixed by curve control, the chord error can be controlled by controlling step size. By controlling the contour error, if the step size will be small[14] then the chord error will also be small. So control the change of Δu_j and it can control the contour interpolation error within a reasonable range.

Implement the Bezier algorithm by using MATLAB Programming, in this algorithm; we added the dispersion pretreated parameters of deceleration in this algorithm.

```
Program [cb] = bezierInterp(P0, P1, P2, P3, Pn)
 {Assuming the points of bezier curve P := (X, Y, Z)}
  Prompt the user for the coefficients coordinate X, Y and Z.
   if the number of input points < 4 then
       disp('At least four input arguments are required')
      return
   end if
   Read X, Y, and Z
   Read X, Y, and Z reversely for pretreatin speed planning.
   Interpolation preprocessing for the relationship of Pi.
   Interpolation process.
   Assuming chord error is h, anticipate step length is Lp,
            step length determined by chord error is Lh,
            step length at acceleration section is La,
            interpolation step length is L.
   if acceleration section then
      L := min([L0,Lh, La])
   elseif uniform section then
          L := min([0.3, Lh, La])
   else deceleration section then
        L := min([L0, Lh, La])
   end if
  if step error == 0.001 then
     interpolation step + 1
  else parameter modification
  end if
  if interpolation end then
     break
  end if
end
```

The points of Bezier curve are:

P={(10,10,0), (60,80,10), (100,90,60), (150,30,30), (200,50,20), (300,120,50)}

The results of interpolation MATLAB simulation algorithm is given below in Fig.4

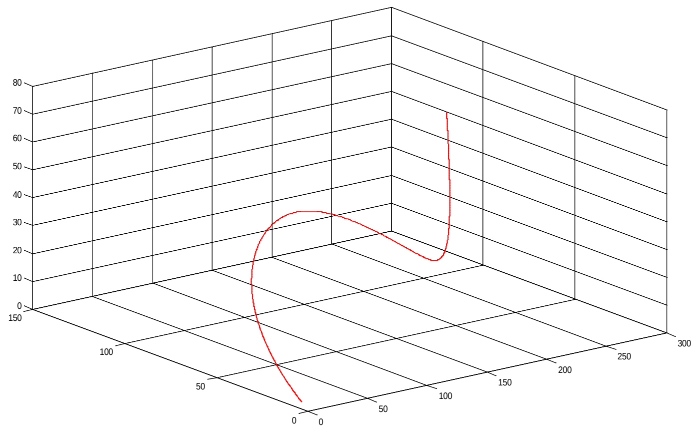

Fig.4. MATLAB simulation of Bezier algorithm interpolation

According to the principle of the algorithm above, MATLAB calculated all the interpolation points on the Bezier curve based on the points of Bezier curve, as interpolation outputs shown in Table 1, which meet the limitation of chord error 0.001mm after 1365 times interpolation operation. In Fig.4, the red curve is 3-dimensional dynamic simulation track of Bezier curve algorithm at the highest speed of 12m / min. Analyzing the outputs of all the interpolation points of the MATLAB in Table 1, and accurately complete the interpolation process from start point to the end point, the simulation results meet the requirements of Bezier algorithm, which ensure the correctness to the hardware design.

Table 1. The interpolation points of Bezier curve.

Step	Interpolation coordinate points		
	X	Y	Z
i=1	10.0000	10.0000	0
i=2	10.0000	10.0000	0.0000
i=3	10.0000	10.0001	0.0000
i=4	10.0001	10.0002	0.0000
i=5	10.0003	10.0004	0.0001
i=6	10.0006	10.0008	0.0001
i=7	10.0010	10.0014	0.0002
i=8	10.0016	10.0022	0.0003
i=9	10.0024	10.0033	0.0005
i=10	10.0034	10.0047	0.0007
…	…	…	…
i=1365	300.0000	120.0000	50.0000

5 System Software Design

5.1 Software Flowchart

Based on the timing analysis of the entire system, the whole program of system is controlled in a same timing. First send the control information required by the interpolation through the PC control interface, which include the curve points and error constraints. The sampling period is given early in the program. After DSP receives these signals, it decides the current position information of the motor and revises the motor to the interpolation start point. The start coordinate points were calculated according to uploaded interpolation points, and start to calculate the points of interpolation according to the upload points and judge the accuracy to meet the requirements. After that, it send the data to FPGA make it through the appropriate switch control and send the output pulses to the motor for completing the interpolation process, FPGA judge the end of interpolation is finished or not preliminarily, and then feedback this signal to DSP, the final end of interpolation instruction issued by DSP. The C language is used for DSP programming and the software used for it has the version of CCS4.1.3. The Verilog HDL hardware description language is used for FPGA programming and we used Quartus II 13.0 software. System flow chart is shown in given below Fig.5.

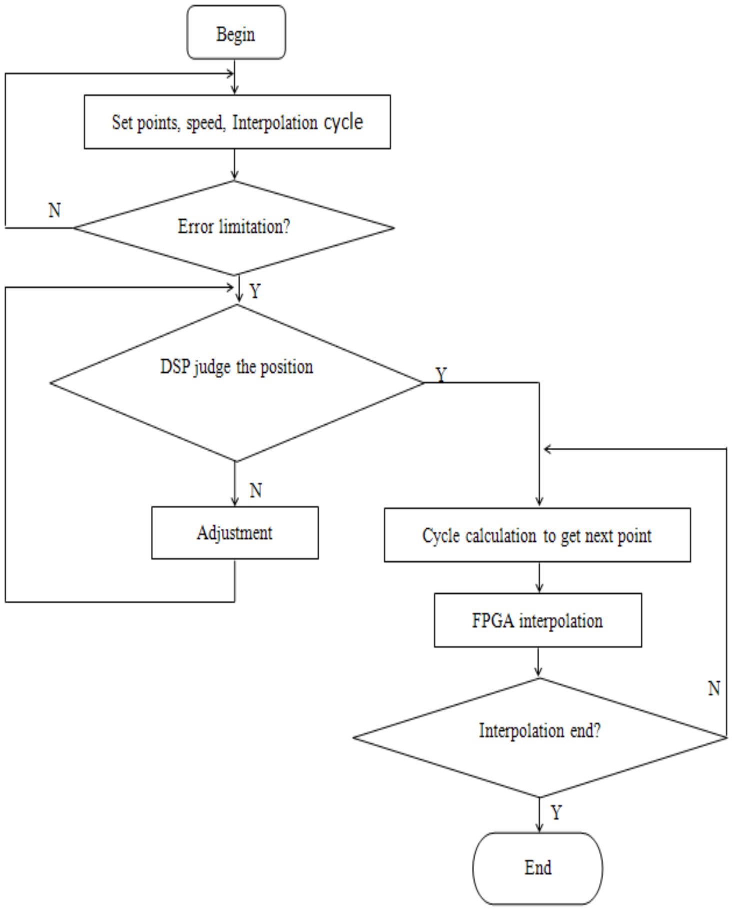

Fig.5. Software flowchart

5.2 The Timing Simulation of Bezier Algorithm in FPGA

Simulation in ModelSim-Altera 10.1d, the testbench is shown as follow:

```
Program tb_cb_ctrl();
  define clk, rst_n, bezier_req, as a type of register data;
  define P1_x,P1_y,P1_z reg rst_n,
         P2_x,P2_y,P2_z,
         P3_x,P3_y,P3_z,
         P4_x,P4_y,P4_z,
         P5_x,P5_y,P5_z,
         P6_x,P6_y,P6_z, as a type of 32-bit register data;
  define bezier_ack,
         axis_x_pulse,
         axis_y_pulse,
         axis_z_pulse,
         axis_0_pulse,
         axis_1_pulse,
         axis_2_pulse,
         axis_x_dir,
         axis_y_dir,
```

```
          axis_z_dir,
          axis_0_dir,
          axis_1_dir,
          axis_2_dir, as a type of net wire data.
   Clock generation
   Reset generation
   Instantiate the interface signal of bezier_inst
   Bezier curve interpolation of FPGA procedure
   Initialization
   Trigger definition
  end
```

Fig.6 is the timing simulation of this system, the using system clock is 50MHz. In the process of the interpolation, when start to interpolate, bezier_req signal is converted from low to high, and FPGA judge whether the interpolation can begin or not and then change the bezier_ack signal to high. The output pulses of six-axis are converted by the interpolation coordinate points, the simulation end time is 50ms, and the time scale is 1um, as can be seen in the figure, it could output pulses correctly. The module of curve interpolation control the output calculation results of servo motor drive which controls the servo motor to complete the interpolation process. After compiling the whole project, the total number of using LEs are 9927, the resource utilization rate of FPGA chip is more than 96%. This system ensures all the interpolation points are on the curve, and the graph shows that the design of the space curve interpolation process is accurate in Fig.6.

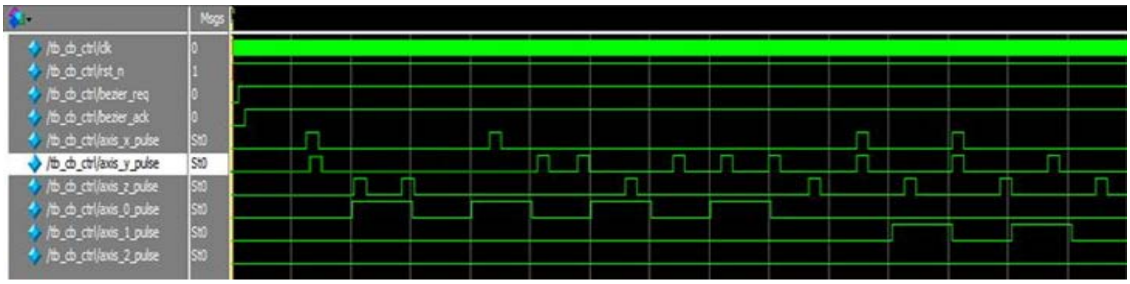

Fig.6. The timing simulation of Bezier algorithm

In Fig.6, the description of each signal is shown in Table 2. In the table, we can know the name of each signal, the direction of I/O interface and the function of each signal.

Table 2. The signal of FPGA.

Signal	I/O Direction	Description
clk	Input	system clock
rst_n	Input	reset signal,
bezier_req	Input	interpolation request signal
bezier_ack	Output	interpolation acknowledge signal
axis_x_pulse	output	x-axis component of interpolation increment
axis_y_pulse	output	y-axis component of interpolation increment
axis_z_pulse	output	z-axis component of interpolation increment
axis_0_pulse	output	auxiliary axes of x-axis
axis_1_pulse	output	auxiliary axes of y-axis
axis_2_pulse	output	auxiliary axes of z-axis

6 PC Control Interface

CNC software interface has a very important role in CNC system. The best control interface can directly affect the efficiency of CNC machine. This control interface system is based on study and understanding the actual CNC machine. The interface is intuitive and concise design. By using so many people, it is proved that you can achieve accurate results.

By designing the interface, the software interface follows the rules of design that are given below[15].

1) Principle of consistency: The interface should have a familiar look, user friendly and information display formats, etc, between different operating systems.

2) Provide feedback.
3) Maintain a reasonable layout of the interface.
4) Reasonable use of color and graphics.
5) Select the appropriate font size.

So this interface follows the all rules that we have explained above. It also helps users to learn the function of the machine, and reduce the error rate. Meanwhile the interface has also the information of location and current status. It has user-friendly processing for timely observation and adjustment.

In addition, for the position information of encoder feedback, the interpolation speed, direction and position feedbacks are very well and timely manipulate. As can be seen from Fig.7, connect hardware and computer by USB cable, and click the Open Device button, then upload the points of Bezier curve which are written in the file of txt format, choose the Output Mode, after setting all control parameters, then click Six Axis output to begin the process of Bezier curve interpolation. After several tests, the control interface can be stably controlled interpolation controller to complete the interpolation process. According to the principle color using, the interface uses light blue, so that user feels comfortable in terms of visual experience. According to design principles of CNC machine interface that is combined with algorithm by using Visual C ++, this interface is reliable as shown in Fig.7.

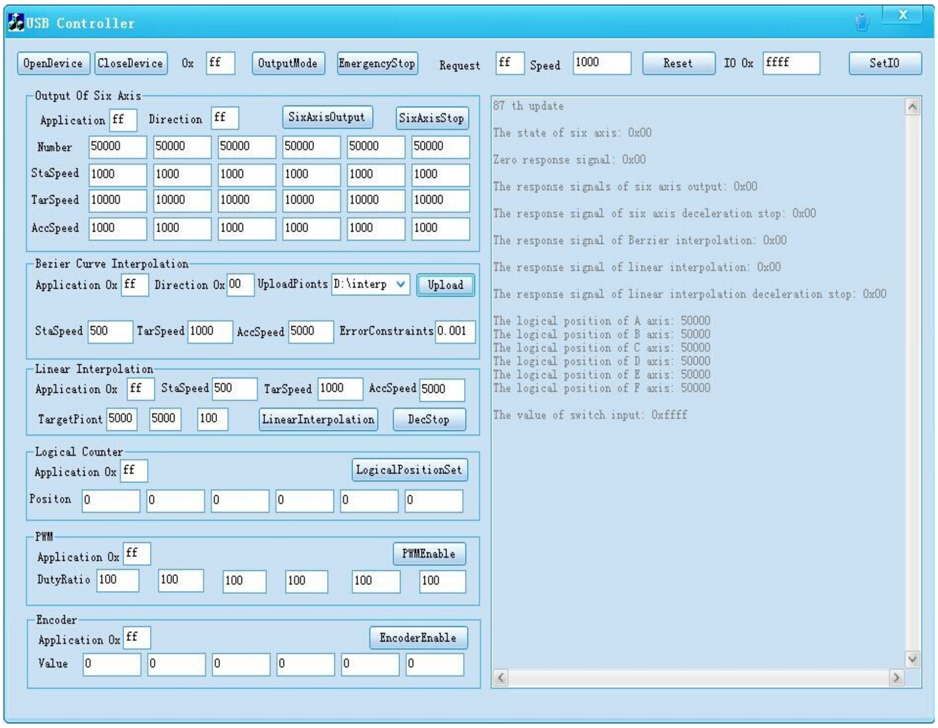

Fig.7. PC control interface

7 Conclusion

In this paper, we used DSP and FPGA for design and implementation of spatial interpolation Bezier curve control system. It explains the hardware and software system design for the algorithm by using DSP and FPGA structure. Compared with the single chip like DSP, this interpolation algorithm has solved the limitation like high speed of interpolation etc. If using single FPGA[16], the algorithm would occupy more logical units, which would cost more than combine DSP and FPGA. So this design not only ensures the accuracy and speed of modern CNC control requirements, but also reduces the design costs. The savings FPGA resources can be reconstructed to other application modules, and provide the conditions for future multi-functional extension. From the procedures and final test results, we can obtain that the highest frequency of the pulse output is more than 1MHz, which is better than the current general CNC 200 KHz or 512 KHz.

This design achieved the algorithm of 3D Bezier curve interpolation, and added speed control in the process of interpolation, it could meet the highest speed that is 12m/min. Cubic Bezier curve interpolation algorithm can be directly used in the list of the curve data points converted to Bezier control points, and the interpolation has no multiplication, only addition (equation (5)), so the interpolation speed is high and has small amount of processing. Chord error meets the requirements through control incremental parameters, feed rate error is small (the maximum error of interpolation is 0.001mm while most of other motion control systems are 0.004mm~0.008mm). Compared to the traditional CNC motion controller, using this system to realize arbitrary curve interpolation could reduce artificial workload of curve path planning, and ensure the real-time interpolation. The results from the simulation waveform and theoretical analysis shows that the system performance is fast, compare with interpolation results and theoretical analysis, which proved the system design is accurate.

This paper has important theoretical value and practical value in improving the performance of CNC, as the short time and personal level limited, there are a lot of following work: the first is the optimization of control algorithm, to make the Bezier interpolation algorithm control easily in operation, expand the space curve to space surface, and improve the speed control method, combine with FPGA parallel processing capabilities and high-speed arithmetic processing capabilities, improve the performance of CNC interpolation controller to a large extent.

8 Acknowledgements

This work was financially supported by the Basic Scientific Research Professional Foundation for Colleges and Universities in Gansu Province (Grant no. 620027).

References

[1] W. Tai-yong, L. Bo, W. Shu-min, L. Hong-wei, "Reconfiguable CNC system based on fieldbus," *Computer Integrated Manufacturing Systems*, Vol.12, No.10, pp. 1662-1667, 2006.

[2] H.-Y. Luo, S.-J. Wei, D.-H. Guo, "RROCN: An On-Chip Network with Regular Reconfigurable Topology for Chip-Multiprocessors," *Journal of Computers*, Vol.24, No.1, pp. 36-46, 2013.

[3] D. Haitao, Z. Shujun, C. Bing, C. Youping, "Design of reconfigurable CNC systems based on ARM+DSP+FPGA," *Huazhong Univ. of Sci. & TECH. (Natural Science Edition)*,Vol. 40, No. 8, pp. 82-87, 2012.

[4] W. You-ming, Y. Guo-tai, "An Application of Bezier Curve Interpolation Algorithm in CNC Systems," *Mechanical &Electrical Engineering Magazine*, Vol.17, No.6, pp. 45-47, 2000.

[5] Ye Bosheng,YangShuzi, "A research on high speed of 3-spline curve interpolation in CNC system," *China Mechanical Engineering*, Vol.9, No.3, pp. 42-43, 1998.

[6] S.S. Yeh, P.L. Hsu, "Adaptive-feedrate interpolation for parametric curves with a confined chord error," *Computer-Aided*, Vol.34, pp. 229-237, 2002.

[7] Z. Guo-juan, J. Hong-li, S. Fu-gen, "Research of motion controller based on DSP and FPGA," *Electronic Design Engineering*, Vol. 20, No. 19, pp. 189-192, 2012.

[8] Z. Lin, M. Lingjun, Y. Lei, Z. Yuan, "USB2.0 high speed interface design based on CY7C68013A," *Computer Technology and Its Applications*, Vol. 40, No. 1, pp. 131-133, 2014.

[9] L. lingshun, G. Yanli, Z. Shutuan, W. Fang, "TMS320F28335DSP yuanli ji kaifa biancheng," *BEIHANG UNIVERSITY PRESS*, Beijing, pp. 1-3, 2011.

[10] S. Usenmez, B.R. Mutlu, U. Yaman, E. Kilic, M. Dolen , A.B. Koku, "A New Hardware-in-the-Loop Simulator for CNC Machine Applications," in *Proceedings of 2013 IEEE International Conference on Mechatronics (ICM)*, pp. 658-663, 2013.

[11] W. Chen, Z. Wen, Z. Xu, J. Wang, "Implementation of 2-axis Circular Interpolation for a FPGA-based 4-axis Motion Controller," in *Proceedings of IEEE International Conference on Control and Automation*, pp. 600-605, 2007.

[12] G. Song-qing, X. Fei-yun, J. Min-ping, "Design of the interface between TMS320VC5402 and CY7C68013," *MECHANICAL & ELECTRICAL ENGINEERING MAGZINE*, Vol. 24, No. 12, pp. 85-88, 2007.

[13] Y. Li,XIE Ming-hong, "Cubic Bezier Curve Interpolation Algorithm and Its Error Analysis," *Journal of Chongqing Institute of Technology(Natural Science)*, Vol. 22, No. 7, pp. 34-39, 2008.

[14] H.-T. Yau, J. Bin, "Fast Bezier interpolator with real-time look ahead function for high-accuracy machining," *International Journal of Machine Tools and Manufcrure*, Vol. 47, No. 10, pp. 66-68, 2007.

[15] X. Yu-qing, Y. Sheng-yuan, "The Design Principle Sudy of Human-machine Inrerface of Numerical Control Machine," *Machine Tool & Hydraulics*, No. 12, pp. 118-119, 2004.

[16] D.-Y. Huang, T.-W. Lin, W.-C. Hu, "Implementation of Multilevel Thresholding Process Using Histogram Valley Estimation Method Based on FPGA," *Journal of Computers*, Vol. 23, No. 3, pp. 1-2, 2012.