

基于 FPGA 的步进电机正弦波细分驱动器设计

李宁

摘要：本文提出了一种基于 SOPC 片上可编程的全数字化步进电机控制系统，本系统是以 FPGA 为核心控制器件，将驱动逻辑功能模块和控制器成功地集成在 FPGA 上实现，充分发挥了硬件逻辑电路对数字信号高速的并行处理能力，可以使步进电机绕组电流细分达到 4096，且细分数可以自动调节，极大地提高了控制精度和驱动器的集成度，减小了驱动器体积。

关键词：FPGA，步进电机，正弦波，细分驱动器，可编程逻辑

1 引言

步进电机是一种将电脉冲信号转换成相应的角位移（或线位移）的机电元件，具有结构简单坚固耐用工作可靠的优点因此广泛应用于工业控制领域。由于脉冲的不连续性又使步进电机运行存在许多不足之处，如低频振荡、噪声大、分辨率不高及驱动系统可靠性差等，严重制约了其应用范围。步进电机的细分控制有效地解决了这一问题，但是传统的步进电机驱动系统大多数采用的是用单片机作为控制芯片，外加分立的数字逻辑电路和模拟电路构成。受单片机工作频率的限制，细分数不是很高，因此驱动器的控制精度较低，控制性能不是很理想。随着高性能数字信号处理器 DSP 的出现，以 DSP 为控制核心，以软件方式实现电机控制一度成为研究的热点。近年来随着可编程逻辑器件的飞速发展，使得可编程逻辑器件功能越来越强大从而促使高集成化高精度驱动器的出现。因此本文提出了一种基于 SOPC 片上可编程的全数字化步进电机控制系统，本系统是以 FPGA 为核心控制器件，将驱动逻辑功能模块和控制器成功地集成在 FPGA 上实现，充分发挥了硬件逻辑电路对数字信号高速的并行处理能力，可以使步进电机绕组电流细分达到 4096，且细分数可以自动调节，极大地提高了控制精度和驱动器的集成度，减小了驱动器体积。

2 步进电机细分驱动原理

步进电机的细分控制本质上是对步进电机励磁绕组中的电流进行控制，在普通驱动方式下，驱动电路只是通过对电动机绕组激磁电流的“开”和“关”，使步进电动机转子以其本身的步距角分步旋转。步进电动机靠定子、转子磁极间的电磁力来进行工作，当它处于“双拍”状态工作时，其定位位置是正好位于两通电磁极的中间，即依靠两通电磁极电磁吸引力的平衡而获

得的。由此可以推论:如果能够进一步仔细地控制两磁极电磁吸引力的大小,使转子磁极获得更多种由于两相定子磁极的电磁吸引力差异而形成的平衡定位位置。步进电机细分驱动方式就是应用了这一原理,在细分驱动时,细分控制器通过控制各相励磁绕组电流的逐步增大及逐步减小,让转子处于多个磁力平衡状态使电机内部的合成磁场为均匀的圆形旋转磁场,实现步距角变小、电动机的旋转得到细化的目的。合成的磁场矢量的幅值决定了电机旋转力矩的大小,相邻两个合成磁场矢量的夹角大小决定了该步距角的大小。对于三相步进电机而言,向 A、B、C 绕组分别通以相位相差 $2/3\pi$, 而幅值相同的正弦波电流(图 1), 则合成的电流矢量在空间做幅值恒定的旋转运动,其对应的合成磁场矢量也作相应的旋转从而形成旋转力矩(图 2)。

A、B、C 三相瞬时电流值如式 (1)、(2)、(3) 所示。

$$i_a = I_m \cos \omega t \quad (1)$$

$$i_b = I_m \cos\left(\omega t - \frac{2\pi}{3}\right) \quad (2)$$

$$i_c = I_m \cos\left(\omega t - \frac{4\pi}{3}\right) \quad (3)$$

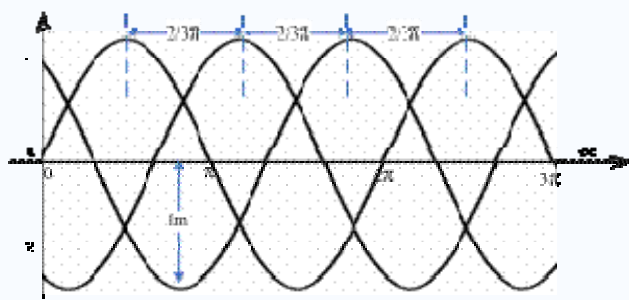


图 1 步进电机正弦细分三相绕组电流波形图



图 2 旋转力矩图

细分驱动方式下，由于步距角小，步进电机的控制精度明显提高，同时这种驱动方式又有效抑制低速运行中产生的噪声和振荡现象。

3 控制器总体设计方案

步进电机三相绕组的电流是正弦阶梯电流，通过改变给定电流的每一次变化的阶梯数可以实现可变细分功能。驱动器的任务就是控制绕组的电流，使之按正弦阶梯波的规律变化。每给一个步进脉冲，A、B、C 三相绕组的电流沿正弦阶梯波前进一步，电机转动一个步距角。

步进电动机驱动主回路图如图 3 所示 IA、IB 为两个霍尔元件。图中 6 个 IGBT 集成在电源控制模块 IPM 内。

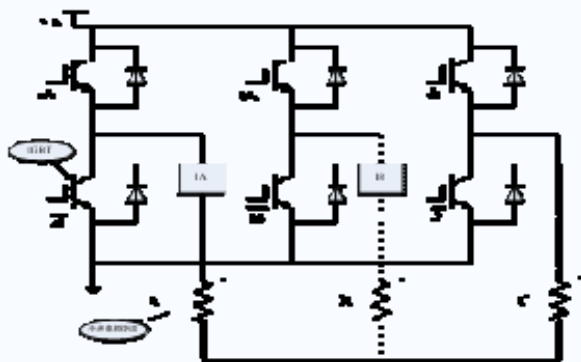


图 3 步进电动机驱动主回路图

步进电机控制系统框图如图 4 所示。采用 FPGA 作为主控制芯片，将控制器与驱动器的数字电路部分集成在一片 FPGA 上实现。为了控制绕组电流，在设计中引入电流跟踪型闭环反馈，反馈电流与给定的正弦电流(离散的正弦表)经过改进的比例积分 PI 调节后进行 SPWM 调制，输出 6 路 PWM 波，来控制驱动电路三个桥臂上的 6 个 IGBT 开通关断。如果忽略死区时间控制每个桥臂的上下半桥的两路 PWM 波互补即上半桥 PWM 波为高/低电平时，下半桥 PWM 波为低/高电平。系统采用 14 位宽度 200MHz 计数器产生 PWM 载波，载波频率 12.2KHz，电流数据全部采用 14 位精度进行离散化。200MHz 时钟由 50MHz 时钟经 PLL 倍频产生。FPGA 输出的 PWM 波经功率模块放大后，控制步进电机运行。步进电机运行状态(转速和转向)通过 LED 指示。步进电机转速是由查表速度决定的，CP 是用来决定查表频率，在细分等级一定的情况下 CP 速度越高电机转速越快。如果电机在高细分下高速旋转则 CP 脉冲频率就会很高，导致 PWM 脉宽过小，使功率模块 IGBT 控制桥臂频繁开关，其结果是开关损耗大为增加，功率模块过热。而高细分在步进电机高速旋转时其优势并不明显，所以在不影响电机运行精度的情况下，系统根据转速对细分精度在 4096、2048、1024、512、256、128、64、32 之间自动调节，使电机更加平稳可靠的运行。

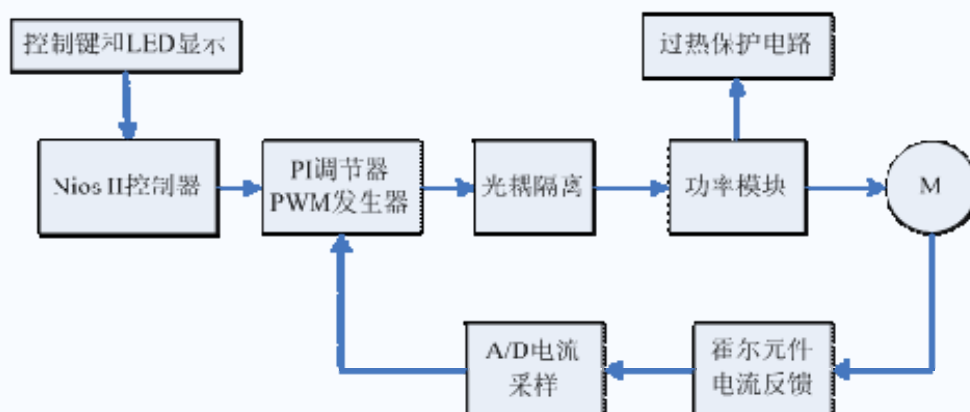


图 4 控制系统框图

3.1 FPGA 结构设计

本系统是由集成在系统采用层次化开发方式，顶层模块由多个子模块组合而成，FPGA 设计系统框图如图 5 所示。系统由 Nios II 微处理器模块和数字逻辑模块组成。

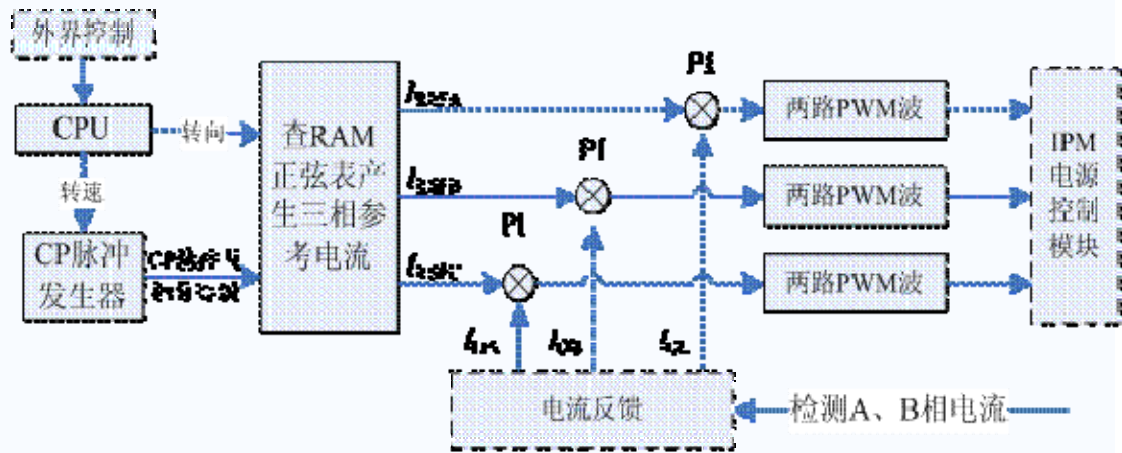


图 5 FPGA 设计系统框图

控制器采 Nios II 的 32 位 CPU 软核，由 SOPC Builder 生成。用来接收外界控制信号并把信号处理后送给脉冲发生器，并通过输出端口控制 LED 来指示系统工作情况。系统顶层设计图如图 6 所示，系统由 CPU 软核 (stepmotor_controller、CP 发生器(PatternGenerator)、正弦查表控制模块 (interface)、A/D 电流采样模块 (currenttop)、C 相电流计算模块 (Current_c) 调节模块、PI 调节模块 (pimodule)、PWM 发生模块 (PWM) 组成，各模块功能如下：

CPU: 接受外界控制信号把信号处理后得到相应的细分等级和与速度对应的控制信号送给 CP 发生器，把转向信号处理后送到查表控制模块,并把转速和转向信号送到 LED 数码管和发光二极管来指示电机运行情况；

CP 发生器: 接受转速控制信号发出与转速相对应的 CP 脉冲并给出相应转速下对应的最佳细分等级；

正弦查表控制模块: 根据 CP 脉冲和细分等级进行查表产生参考电流，根据转向控制信号来

enable 端口是 PWM 发生器使能端，功率模块温度过高热保护就会动作将 enable 置 0 停止发出 PWM 波，保护驱动器；

setspeed、rotate_dirc 分别为转速和转向控制端口；

LED_sel、speeddisplay、LED_indicator 为工作情况指示电路；

pl_a、ph_a、pl_b、ph_b、pl_c、ph_c 为 A、B、C 三相桥臂的上下半桥控制端；

3.2 Nios II 软核设计

在本控制器中利用 CPU 控制灵活的功能用来接受并处理外界控制号再将处理过的控制信号送到各个功能模块，因此 Nios II 软核采用了经济型即 Nios II/e，加入了用于输入输出端口（PIO）、片上存储器模块（On chip Memory）。控制器程序流程图软件程序工作流程如图 7 所示。

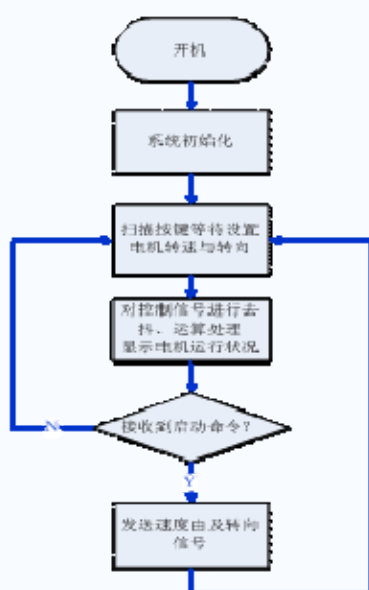


图 7 CPU 控制流程图

CPU 内部程序采用 C++ 语言编写，并对电机的控制状态建立类。通过对电机的控制状态对象的操作，即可输出各种控制命令到接口模块和 CP 脉冲发生模块，控制电机状态；同时在程序主循环中监视用户的各种输入，更新电机的控制状态对象的成员变量。

3.3 CP 发生器模块和查表控制模块

PatternGenerator 模块根据设置的电机转速产生相应的 CP 脉冲和在该转速下的最适宜的细分等级，并把这些信号传送给查表控制模块。CP 发生器模块和查表控制模块如图 8 所示。

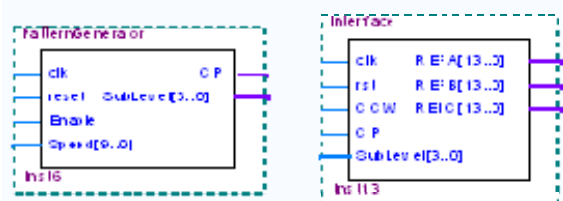


图 8 CP 发生器模块和查表控制模块

查表控制模块（interface）内部例化了一个只读存储器，存放一个周期的正弦数据表。正弦表数据宽度与系统数据流数据宽度一致，因此选择 14 位宽度。正弦表深度决定了驱动器的最大细分等级，关系式表示为： $N_{MAX}=2N$ ，当 $N=12$ 时，达到最高细分等级 4096 细分。外部输入 CP 和 CCW 信号后，通过顺序的给出三相绕组对应的地址数据，即可查得三相相电流的给定数据，并同时更新三个给定数据寄存器，送入下一个环节。转向交换也在本模块实现。输入的 CCW 信号将控制一个路选择器，控制正弦表的查表方向来改变步进电机转向。本模块通过输入的细分等级信号和 CP 脉冲信号共同决定三相绕组的地址数据的每次增量大小。当细分等级最高时，地址计数器每次增 1 或减 1；当处于非最高细分精度时，地址计数器每次递增 2 的 N 次幂，N 由当前的细分等级确定。

3.4 AD 采样模块与 C 相电流计算模块

步进电机 A、B 两相电流通过霍尔元件进行采集，AD 采样模块通过采集采样电阻上的电压来将电机 A、B 两相电流的模拟量转换为数字量，C 相电流模块由基尔霍夫电流定律 $i_a+i_b+i_c=0$ 计算得到。图 9 为两模块的端口。

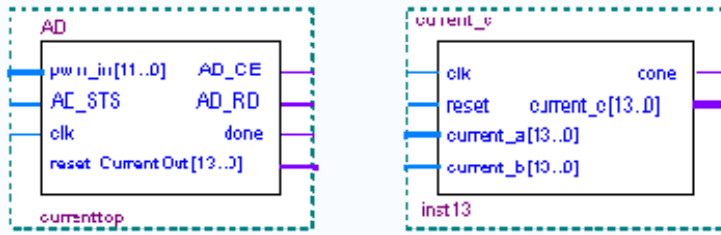


图 9 AD 采样模块与 C 相电流计算模块

3.5 PI 调节模块

PI 调节模块是对给定数据和反馈数据进行离散化 PI 调节。模块的数据宽度统一为 14 位，即给定数据/反馈数据/控制数据都用 14 位宽度无符号整型数据表示。本模块实现一个 PI 控制器功能，如图 10 所示。

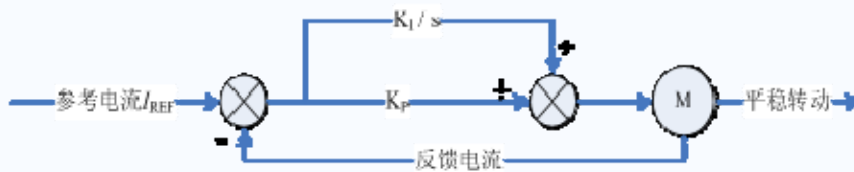


图 10 PI 控制逻辑框图

PI 调节离散化公式如(4)式所示。

$$I(K)=I(K-1)+K_P*(E(K)-E(K-1))+K_I*E(K) \quad (4)$$

其中， $1500 \leq I(K) \leq 14384$ ， $I(0) = 8192$ ， $E(K)=I_{REF}-I_{OUT}$

模块在收到启动信号后，先寄存给定数据/反馈数据，再对二者扩宽符号位做有符号减法，并启动 K_I 乘法器计算 K_I 乘以误差 e 的结果。对减法的计算结果取绝对值，并与阈值作比较（常数比较器），比较的结果存入寄存器。计算 e 的误差增量，即 $E(K)-E(K-1)$ ，并用 K_P 乘

法器计算 K_P 乘以误差增量的结果。将 K_I 乘法器计算的结果先送入累加器，然后再将 K_P 乘法器计算的结果送入累加器；模块的累加器在每次累加的时候会进行溢出处理和数据超出边界处理，无论哪种情况都会置为边界值(最大/最小值)。

3.6 PWM 发生器模块

PWM 发生器模块将 PI 调节后输出的数据转化为 PWM 波形输出给 IGBT 模块，通过 IGBT 的通断切换调整相电压，进而改变相电流按照给定进行变化。PWM 波形的生成是通过将给定控制数据和载波数据比较，通过比较二者大小情况确定 PWM 的高低电平，实现 PWM 调制，如图 11 所示。



图 11 PWM 模块总体框图

锯齿波作为 PWM 载波，驱动时钟频率为 200MHz，14 位寄存器宽度。为减少计数器的进位链长度，采用计数器，拆分为 3 位计数器和 11 位计数器，减少了组合逻辑延迟。死区保护发生器模块根据给定的死区长度数据，将输入的 PWM 信号输出为 PH 和 PL 信号，分别对应 IGBT 的上下桥臂，并在 PH 和 PL 数据的变沿插入死区保护(同时置为低电平)。插入的死区保护是通过上升计数器和下降计数器来实现的，保证上桥臂和下桥臂不同时导通。

4 仿真结果

对 PWM 模块进行功能仿真，仿真波形如图 12 所示。经分析，六路 PWM 波满足时序和死区保护的要求，实现了设计所要达到的目的。将图 12 进行放大可见 IGBT 死区如图 13 所示。

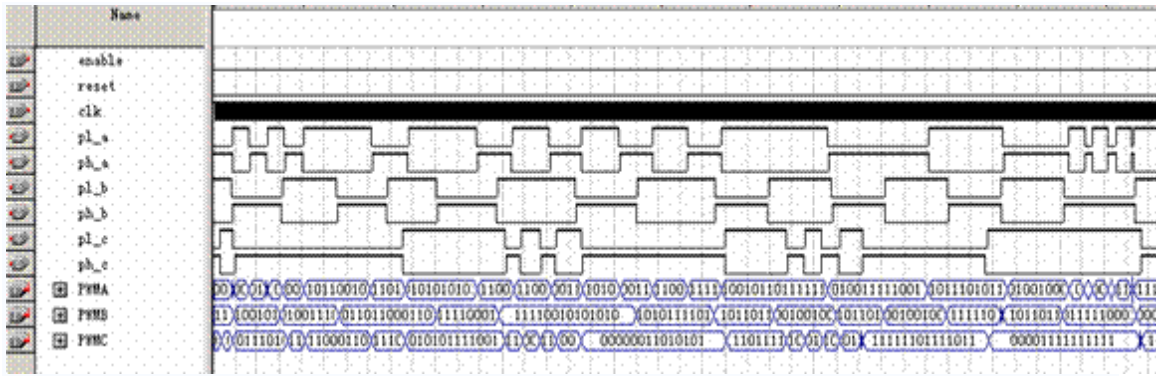


图 12 PWM 功能仿真波形

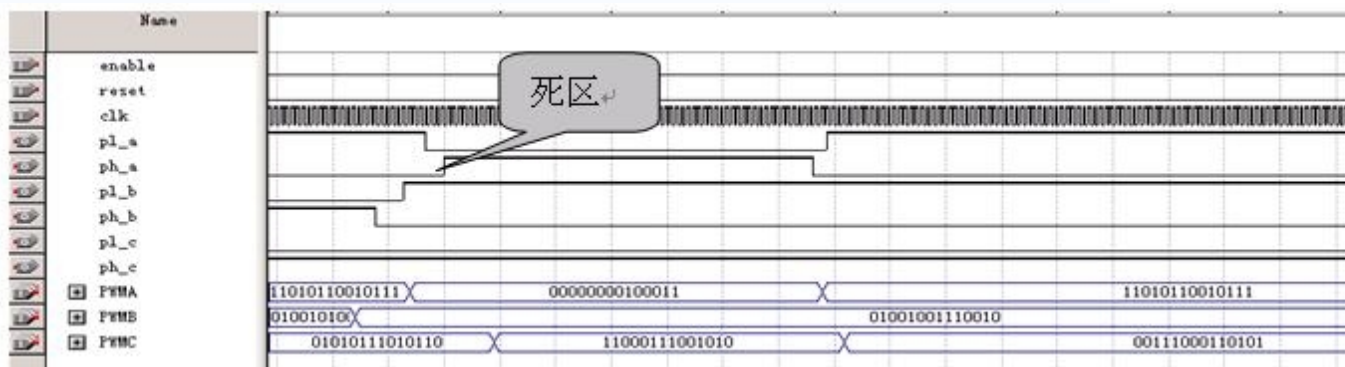


图 13 IGBT 控制信号死区

5 电机运行波形及分析

把 FPGA 配置好之后，对系统调试电机运行时三相电流波形如图 14 所示，电机换向时进行反转查表，电机转速反转波形如图 15 所示。

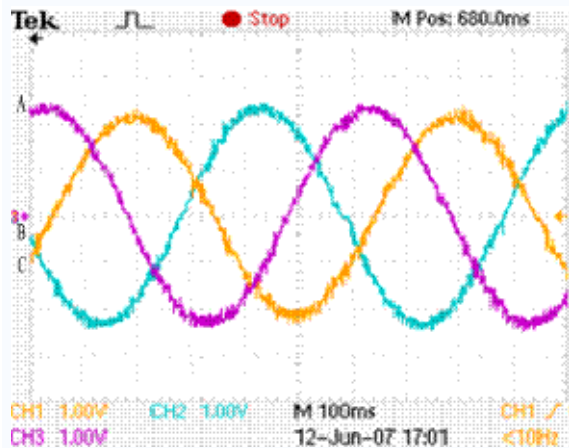


图 14 步进电机运行时三相电流

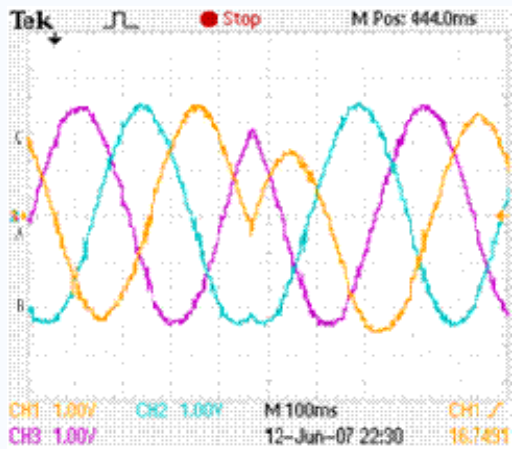


图 15 步进电机反转时电流波形

实验结果表明步进电机运行时三相电流相位差 120 度，三相电流为正弦波，电机运行平稳，电机反转时进行反相查表，转换方向时反应时间短，电流波动小。

6 结论

本控制器采用 Altera 公司嵌入 Nios II 处理器软核的 FPGA 实现了步进电机正弦波可变细分控制，大大提高了系统的可靠性，降低了设计成本。

控制部分采用 Nios II CPU 软核是系统控制更加方便灵活。CPU 和各个控制模块集成在单片 FPGA 内，使驱动器集成度大为提高，减小了驱动器的体积。大大提高了细分精度，实现了步进电机在是低转速下的 4096 高细分的运行，消除了步进电机存在的低频振荡问题，控制精度显著提高，在高精密数控机床中有广泛的应用前景。

来源：ICBUY