

文章编号:1000-582x(2001)03-0080-03

FFT 处理器的高密度可编程逻辑器件实现

唐治德, 姚玉坤, 刘晓明

(重庆大学 电气工程学院, 重庆 400044)

摘要: 为了提高快速离散傅立叶变换(FFT)的处理速度,研究了一种宜于高密度可编程逻辑器件(CPLD)实现FFT处理器的硬件结构,并利用CPLD FLEX10K设计和实现了128点FFT单片处理器。系统的仿真结果表明,该处理器运算结果正确,在系统时钟频率为20 MHz时,128点复数FFT处理器的计算时间小于230 μs。研究表明:CPLD与FFT的结合将提高FFT的处理速度,从而使FFT的应用更加广泛。

关键词: 快速傅立叶变换; 可编程逻辑器件; 逻辑设计

中图分类号: TN 492; TN 911.72

文献标识码: A

高密度可编程逻辑器件(CPLD)是当今最具活力、应用前景极好的集成电子器件,它具有速度快、集成度高、功能强、可重复定义和反复擦写、设计方法先进等突出优点,显示出诱人的应用前景^[1,2]。它的广泛应用正在深刻地影响着数字系统的实现手段和设计方法,极大地提高电子系统的质量和可靠性,强有力地推动着电子系统微型化单片化应用的发展。FLEX10K系列,是业界第一次实现嵌入式阵列的CPLD。嵌入式阵列由许多嵌入式阵列块(EAB—Embedded Array Block)构成。嵌入式阵列用来实现高效存储功能和复杂的逻辑函数,如数字信号处理、微控制器等功能;逻辑阵列用来实现通用逻辑函数,如计数器、加法器、状态机及多路选择器。

在数字信号处理中有许多实现快速傅立叶变换(FFT)的方法^[3,4],但由于FFT处理的数据量大、算法复杂,FFT处理器的计算速度不能满足高速实时处理系统的要求,特别是在变换点数较大时,“速度”更是成为困扰工程技术人员的难题。文中从硬件的角度探索提高FFT处理速度和性能价格比的新途径,即采用高密度可编程逻辑器件(CPLD)实现FFT。CPLD与FFT的巧妙结合必将使FFT的处理速度迈上一个新的台阶,从而使FFT的应用提高到一个新的水平。

1 CPLD实现的FFT算法框图

离散傅立叶变换为^[4]:

$$X(k) = DFT[x(n)] = \sum_{n=0}^{N-1} x(n) W_N^{kn},$$

$$k = 0, 1, \dots, N-1$$

$$W_N^{kn} = e^{-j\frac{2\pi}{N} \cdot kn} = \cos\left(\frac{2\pi}{N} \cdot kn\right) - j \sin\left(\frac{2\pi}{N} \cdot kn\right)$$

式中N为2的整数幂。以N=8为例,基2FFT时间抽取算法的信号流程图如图1所示^[5]。

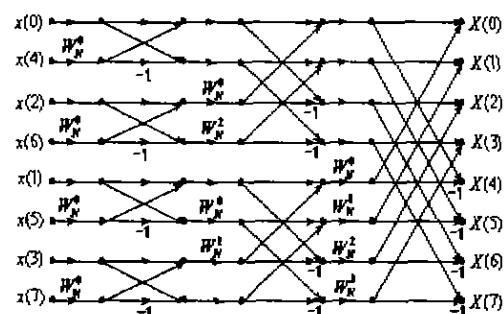


图1 8点FFT运算的信号流程图

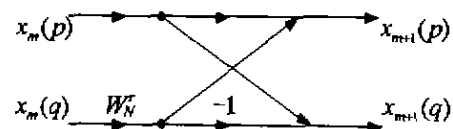


图2 时间抽取算法中的蝶形计算

FFT信号流程图由若干级迭代的如图2所示的蝶形运算组成,一个蝶形运算需要进行一次复数乘法和两

• 收稿日期:2000-08-30

作者简介:唐治德(1958-),男,重庆人,重庆大学教授,硕士。从事电子技术研究。

次复数加法运算, 对于一个长度为 $N = 2^M$ 的序列来说, 蝶形图共有 $M = \log_2 N$ 列(级), 每一列有 $N/2$ 个蝶形运算, 第 L 列中使用旋转因子的个数为 $2^L, L = 0, 1, 2, \dots, M - 1$ 。

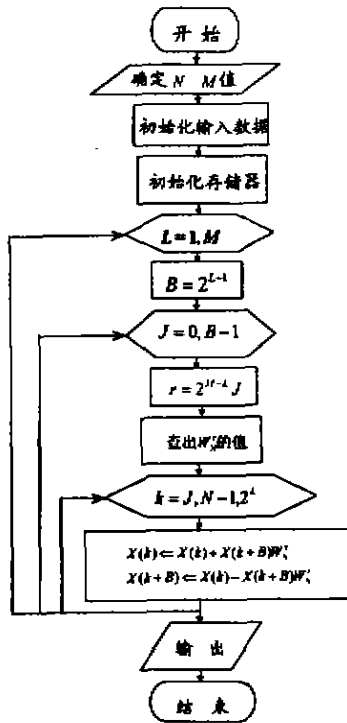


图 3 基 2 FFT 硬件实现处理流程

从蝶形运算可以看出, 第 m 列的 $x_m(p), x_m(q)$ 在经过蝶形运算后, 在第 $(m + 1)$ 列中的节点序号是

不变的, 即 $x_{m+1}(p)$ 和 $x_m(p)$ 可共享同一存储单元, $x_{m+1}(q)$ 和 $x_m(q)$ 可共享同一存储单元。

考虑到 FLEX10K 器件能实现片内存储器, 并可初始化。基于上述考虑, 设计了图 3 所示的便于 CPLD 硬件实现的基 2 FFT 顺序处理流程图。框图中 N, M 分别为 FFT 的点数和迭代运算的总级数, L 为运算级数循环变量, B 为每个蝶形内上下节点两个输入数据的间距; 由循环变量 J 控制依次查出 B 个旋转因子 $W_N^r (r = J * 2^{M-L})$, 循环变量 K 控制进行第 L 级中使用同一旋转因子的蝶形运算。“蝶形运算框”用两个输入数据 $X(k)$ 和 $X(k+B)$ 及旋转因子 W_N^r 进行蝶形运算后再将结果回存到存放 $X(k)$ 和 $X(k+B)$ 的存储单元 k 和 $k+B$ 中。

2 FFT 处理器的 CPLD 结构

根据图 3, 遵循自顶向下的设计思想, 设计了图 4 所示的 FFT 处理器的顶层框图。以 $N = 128$ 为例, FFT 处理器的工作原理为: 在“控制器”的协调下, “地址形成器”每次产生一组地址, 即 $x(k), x(k+B)$ 和 W_N^r 的地址, 这 3 个地址依次被“地址选择器”选中, 从“存储器”中读出数据 $x(k), x(k+B)$ 和 W_N^r , 送“输入寄存器”缓存, 3 个复数经“蝶形运算器”运算后送入“输

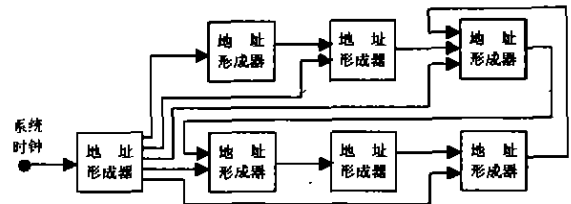


图 4 基 2 FFT 处理器系统框图

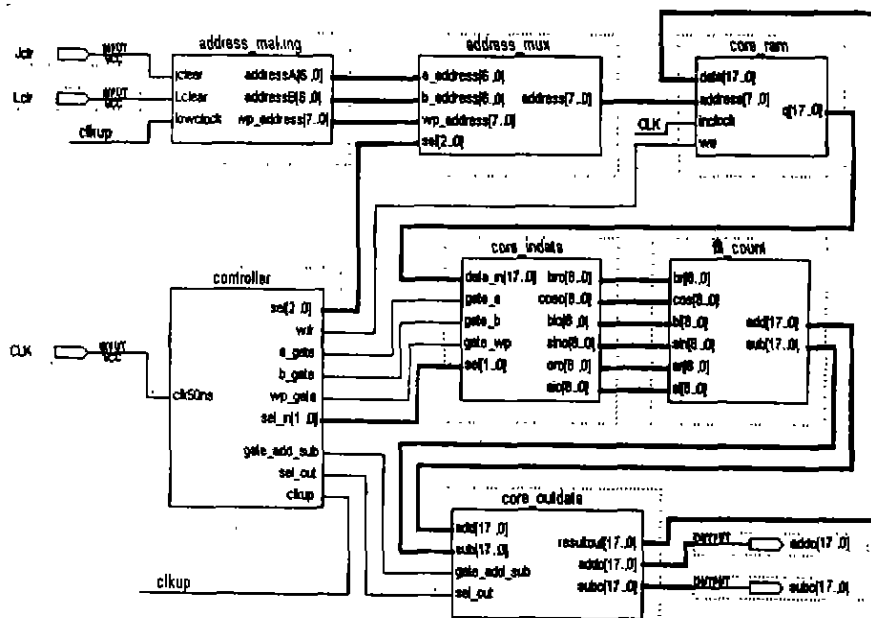


图 5 128 点 FFT 系统顶层原理图

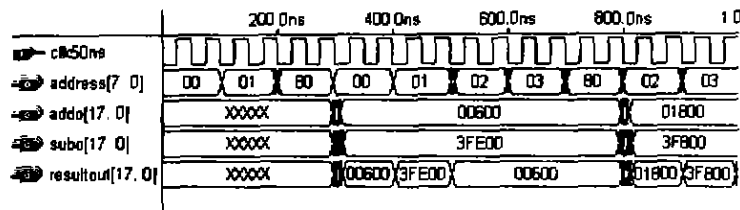


图6 128点FFT处理器仿真结果

出寄存器”,并按“原址计算”写入存储器中。重复上述过程,直至 7×64 个蝶形运算全部完成。

3 FFT处理器的逻辑设计与仿真^[6]

FFT处理器的顶层原理图如图5所示,图中输入引脚“jclr”与“lclr”为两个“清零”输入端,“CLK”为时钟输入端,输出引脚“addo[17..0]”与“subo[17..0]”为一个蝶形运算的两个结果输出端。模块 address-making 是地址形成器;模块 address-mux 是地址选择器;模块 core-ram 是存储器;模块 controller 是控制器;模块 core-indata 是输入寄存器;模块 fft-count 是蝶形运算器;模块 core-outdata 是输出寄存器。图6是仿真结果。由图知:系统实现了128点复数FFT处理功能;当时钟频率为200MHz时,完成一次蝶形运算需要500ns,128点FFT处理器共需要 $7 \times 64 \times 500$ ns = 230 μ s。

4 结论

本文中使用了CPLD FLEX10K30器件实现了128点复数FFT数字处理系统的单片集成,并对单片集成

的128点复数FFT数字处理系统进行了模拟仿真,仿真结果表明:

1) 系统实现了128点复数FFT处理功能。

2) 当系统时钟频率为20MHz时,达到了在230 μ s内完成128点复数FFT计算的高速度。

参考文献:

- [1] 刘宝琴. ALTERA可编程逻辑器件及其应用[M]. 北京:清华大学出版社,1995.
- [2] 宋万杰. CPLD技术及其应用[M]. 西安:西安电子科技大学出版社,1999.
- [3] MEYER R, SCHWARZ K. FFT Implementation on DSP-Chip-Theory and Practice[J]. IEEE Proc ICASSP, 1990, 4: 1 503-1 506.
- [4] LI Z, SORENSEN H V, BURRUS C S. FFT and Convolution Algorithms on DSP Microprocessors[J]. IEEE Proc. ICASSP, 1986, 4: 93-107.
- [5] 程佩青. 数字信号处理教程[M]. 北京:清华大学出版社,1995.
- [6] ALTERA CORPORATION MAX + PLUSII Programmable Logic Development System AHDL [M]. San Jose: Recycled Paper, 1997.

Complex Programmable Logic Devices Implementation of FFT Processor

TANG Zhi-de, YAO Yu-kun, LIU Xiao-ming

(College of Electrical Engineering, Chongqing University, Chongqing 400044, China)

Abstract: In order to accelerate the working speed of a FFT processor, this paper presents a hardware structure of a FFT processor, which is suitable for CPLD, and the 128-point FFT single-chip processor is realized by means of FLEX10K CPLD and MAX + PLUS II software. The calculation time of the processor is less than 230 μ s. The research proves that the combination of CPLD and FFT increases the working speed of FFT processors and raise the level of application.

Key words: FFT; PLD; logic design

(责任编辑 李胜春)