

基于FPGA的超高速FFT硬件实现

王林泉, 皮亦鸣, 陈晓宁, 肖欣

(电子科技大学电子工程学院 成都 610054)

【摘要】介绍了频域抽取基二快速傅里叶运算的基本原理;讨论了基于FPGA达4 096点的大点数超高速FFT硬件系统设计与实现方法,当多组大点数进行FFT运算时,利用FPGA内部大容量存储资源,采用乒乓结构进行流水线运算,提高FFT运算速度,同时保证结果的准确性;对实际硬件进行了FFT运算测试,测试结果证明了系统的可行性和正确性,并且利用该硬件系统成功完成了星载SAR实时成像处理。

关键词 基二; 快速傅里叶变换; 现场可编程门阵列; 超高速; 大点数

中图分类号 TN911.72 文献标识码 A

Realization of High-Speed FFT by FPGA

WANG Lin-quan, PI Yi-ming, CHEN Xiao-ning, XIAO Xin

(School of Electronic Engineering, UEST of China Chengdu 610054)

Abstract In this paper, the basic principle of decimation-in-frequency radix-2 FFT algorithm is briefly presented. Then, the design and realization methods of hardware system with FPGA are thoroughly discussed, with which it is easy to implement the FFT algorithm of large points - 4 096 points. The FFT-speed of many groups of large points is accelerated greatly through ping-pong-ram. Finally, this hardware system is turned out to be feasible and accurate through the FFT algorithm experiment. Furthermore, the real-time imaging procession of airborne SAR can be accomplished successfully by this hardware system.

Key words radix-2; Fast Fourier Transform; field programmable gate array; high-speed; large points

在数字信号处理的发展中,许多算法如相关、滤波、谱估计、卷积等都可以化为离散傅里叶变换(DFT)来实现^[1],因此DFT是处理数字信号如图形、语音及图像等领域的重要变换工具。快速傅里叶变换(FFT)是DFT的快速算法。

FFT算法的硬件实现一般有3种形式:1)使用通用DSP来实现;2)用专用DSP来实现;3)通过FPGA来实现。总体来讲,DSP速度较慢,接口不灵活,而且没有FFT运算所需要的巨量存储器,需外置特定的接口、控制芯片和RAM,限制了运算速度,但DSP开发相对简单,技术成熟,开发费用相对较低,目前大部分FFT硬件是用DSP来实现的;FPGA技术近两年才达到可以实现大点数FFT的水平,并且体积、速度、灵活性等各种性能都优于DSP,但开发难度大,研制费用高。本文将讨论基于FPGA的大点数超高速FFT算法。

1 FFT的基本原理及算法结构

对 N 点序列 $x(n)$,其DFT变换对定义为:

收稿日期:2004-06-30

基金项目:高校博士点专项基金资助项目(20030614001)

作者简介:王林泉(1978-),男,硕士生,主要从事信号与信息处理方面的研究;皮亦鸣(1968-),男,教授,博士生导师,中国电子学会高级会员,主要从事雷达成像,雷达与导航系统方面的研究。

$$\begin{cases} X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk}, k = 0, 1, \dots, N-1, W_N = e^{-j\frac{2\pi}{N}} \\ x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-nk}, n = 0, 1, \dots, N-1 \end{cases} \quad (1)$$

式中 $x(n)$ 为时域点； $X(k)$ 为频域点； W_N 为旋转因子。

FFT是利用了旋转因子的周期性和对称性，对DFT进行简化的运算。各种FFT算法可分两大类：一类是针对 N 等于2的整数次幂的算法，如基二算法、基四算法、实因子算法和分裂基算法等，另一类是针对 N 不等于2的整数次幂的算法，以Winograd为代表的类算法^[1]，有重要的理论价值，但是不适于硬件实现。FFT按分解方式的不同又可以分为按时域抽取算法和按频域抽取算法(Decimation In Frequency, DIF)两种^[1]。两种算法在本质上是完全相同的，在运算量和复杂性等方面都完全一样，可以任取其中的一种，本文将讨论的是基二算法和DIF形式。

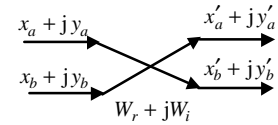


图1 基二蝶形运算单元示意图

FFT运算的基本单元是蝶形运算单元，基二蝶形运算单元如图1所示。其方程式为：

$$x'_a + jy'_a = x_a + x_b + j(y_a + y_b) \quad (2)$$

$$x'_b + jy'_b = [(x_a + jy_a) - (x_b + jy_b)](w_r + jw_i) \quad (3)$$

解式(2)和式(3)得到基二蝶形运算单元输出结果表达式为：

$$x'_a = x_a + x_b \quad (4)$$

$$y'_a = y_a + y_b \quad (5)$$

$$x'_b = (x_a - x_b)w_r - (y_a - y_b)w_i \quad (6)$$

$$y'_b = (x_a - x_b)w_i + (y_a - y_b)w_r \quad (7)$$

从上面的公式可以得出，基二蝶形运算只需两次复数乘法，则 $N=2^n$ 个点的DFT复数乘法量由 N^2 次降为 $N/2 * \lg N$ 次，复数加法由 $N * (N-1)$ 次降为 $N * \lg N$ 。所以在大点数DFT运算时，使用FFT将极大的降低运算量，提高运算效率。

2 超高速FFT算法的硬件实现

整个FFT运算模块在FPGA内部进行配置，本文所讨论的基二FFT运算模块配置框图如图2所示。图中，控制模块用来产生所有的控制信号，存储器2和3分别作为时刻 n 和 $n+t$ 时对应输入 N 点数据的存储器，存储器1作为中间结果存储器，用于存储Butterfly运算模块计算出的奇数级的结果，旋转因子存储器中存储的是 $N/2$ 点旋转因子。

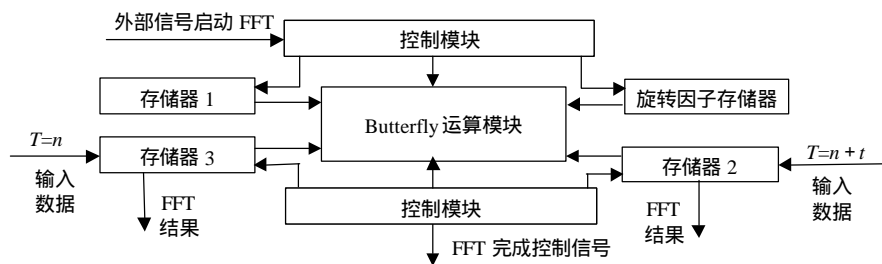


图2 FFT运算模块硬件实现框图

在FFT运算过程中，地址产生是FFT运算模块的关键问题之一，存储器读数据和写数据都要对应相应的存储器地址。在控制模块中定义一个时钟计数器和一个级数计数器，级数计数器随级数的增加自加，在每完成一个FFT之后清零，时钟计数器随每一个时钟自加，在每完成一级FFT之后清零，通过这两个计数器的加减和移位可以产生所有需要的地址。

地址产生中的位反序是FFT运算的最关键问题，DIF形式的FFT输入数据 $x(n)$ 地址为顺序，但由于在运算过程中对 $x(n)$ 作奇、偶分开，导致输出数据地址不再是原来顺序。例如对于8点DIF形式的FFT，其第一级输

入数据地址是正序0, 1, 2, 3, 4, 5, 6, 7。最后一级数据输出数据地址为反序0, 4, 2, 6, 1, 5, 3, 7。为了得到正确的输出数据, 必须通过二进制码位反转将反序变为正序。在控制模块中, 数据的地址都是由二进制数表示, 反序0, 4, 2, 6, 1, 5, 3, 7分别由三位二进制数表示为000, 100, 010, 110, 001, 101, 011, 111, 将每个数的第2位和第0位交换, 第1位保持不动, 可以得到000, 001, 010, 011, 100, 101, 110, 111, 即0, 1, 2, 3, 4, 5, 6, 7, 即将反序变为正序。对于其他点数的FFT, 如果数据地址由 n 位表示, 位反转的规则为: 第 $n-1$ 位和第0位交换, 第 $n-2$ 位和第1位交换, 第 $n-3$ 位和第2位交换, ……依此类推就可以将反序转换为正序。

逆FFT的实现同样可以采用FFT运算模块, 首先将输入数据求共轭, 再作FFT运算, 最后将得到的结果取共轭除以总点数就是输入数据的IFFT运算结果。因而FFT和IFFT可以由同一硬件模块完成。

由前面基二蝶形运算的分析可以得出理论上基二蝶形运算只需4个32位乘法器, 但实际硬件中, 需要将旋转因子 W_N 和 W_N^* 由有符号小数归一化为有符号整数。在本文中, 是将 W_N 和 W_N^* 按32位有符号定点数归一化(即乘以 $2^{31}-1=2\ 147\ 483\ 647$)后存储到旋转因子存储器, 比较式(4)~(7)可得出, 为保持数据一致性, 在Butterfly运算中 x'_a 和 y'_a 也应乘以 $2\ 147\ 483\ 647$, 因此基二蝶形运算共需6个32位乘法器。STRATIX系列EP1S25芯片提供了80个8位内置乘法器, 由8个8位乘法器可以组成一个32位乘法器。所以EP1S25一共可以提供10个32位乘法器。在本设计中, FFT运算需要6个32位乘法器, 如果需要在频域进行复数乘法运算则需要4个32位乘法器, 一共需要10个32位乘法器, STRATIX系列的EP1S25刚好满足要求。考虑到该因素, 本设计选用了Altera公司的Stratix系列EP1S25芯片。

Stratix系列FPGA主要特点包括: 高性能体系、大容量存储资源、高带宽DSP模块、支持多种I/O标准、高速接口、时钟管理、终端技术、NiosTM软内核嵌入处理器、器件配置和远程系统升级^[2]。EP1S25芯片中包含的DSP单元, 可以完成较为耗费资源的乘法器单元功能。另外, 该芯片包含的大量存储单元, 可保证旋转因子的精度。其主要内部资源如表1所示^[2]。这就是本文选择这一芯片的主要原因。

表1 EP1S25内部资源表

逻辑单元	M512 RAM模块	M4K RAM模块	MegaRAM模块	RAM总量/bit	DSP模块	嵌入式乘法器(9 bit × 9 bit)
25 660	224	138	2	1 944 576	10	80

本文系统在微机与FPGA间的数据通信中采用了32 bit PCI总线接口。PCI总线的数据通信过程包括读传送、写传送、传送终止等, 通过PCI总线实现了高速数据传输, 同时PCI总线协议也确保了数据通信的可靠性和完整性, 从而保证了整个系统的高速性和稳定性。

整个FFT系统以FPGA(EP1S25)为核心。先把程序代码通过JTAG下载入EPC8(该芯片是用来配置SRAM工艺FPGA的EEPROM), 当上电时EPC8自动配置FPGA(EP1S25)由电脑发出控制信号是用来协调原始数据的输入和运算结果数据的输出, 原始数据(由实部32 bit和虚部32 bit组成)是由电脑送出, 经PCI总线由PCI9054传入FPGA做FFT/IFFT运算, 当运算结束后, 运算结果再输出到PCI9054, 由PCI总线送出到电脑。

本文系统特点是:

- 1) 为提高数据精度, 系统全部数据采用32 bit。
- 2) 每次处理对象可以是4 096点数据, 实现了大点数FFT运算。

3) 实现了FFT运算的快速流水操作。采用乒乓RAM的方式, 当多组数据进行FFT运算时, 可由存储器2和存储器3交替接收数据, 如此类推形成乒乓结构的流型运算, 进行FFT运算的同时, 存储器也在接收数据。即在计算存储器3中第 n 组数据的同时, 存储器2则正在接收第 $n+1$ 组数据。这种方式决定了实现FFT运算的最大时间。对于4 096点操作, 其接收时间为4 096个数据周期, 这样FFT的最大运算时间就是4 096个数据周期。另外, 由于输入和输出数据是以一定的时钟为周期依次输入或输出的, 而FFT运算时钟是由FPGA芯片所决定, 故可以利用较高的内部时钟来提高内部FFT运算速度, 从而节省了处理数据的时间, 提高了整个FFT运算效率。

3 硬件试验结果分析及实例应用

3.1 硬件试验结果分析

由FFT运算公式可知,一个方波经FFT运算后为 $Sa(w)$ 函数,图3所示正是由实际硬件对一个数据总长度为512点、脉冲宽度为20点方波进行FFT运算求模归一化后的结果。

由硬件试验结果可说明FFT运算是正确的。由于系统采用基二FFT运算,其核心FPGA及其外围器件都是高速器件,同时内核计算采用并行处理,所以系统可实现大点数FFT高精度与高速运算。

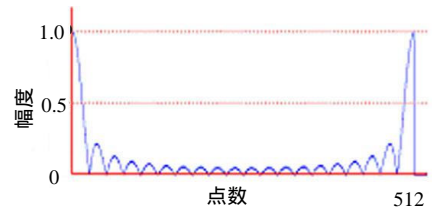


图3 方波的FFT运算结果

3.2 实例应用

星载SAR实时成像处理过程实际上是一个二维解卷积过程^[3],因此可以利用该系统进行处理,即回波信号经过模数转换后,进行距离和方位匹配滤波过程,其数学模型为:

$$\hat{S}(x, r) = S(x, r) \otimes h_r^{-1}(x, r) \otimes h_a^{-1}(x, r) \quad (8)$$

$S(x, r)$ 为回波信号, $\hat{S}(x, r)$ 为地表的散射系数的估值, $h_r^{-1}(x, r)$ 和 $h_a^{-1}(x, r)$ 分别为距离向和方位向线性调频函数,则成像处理的结构功能框图如图4所示。

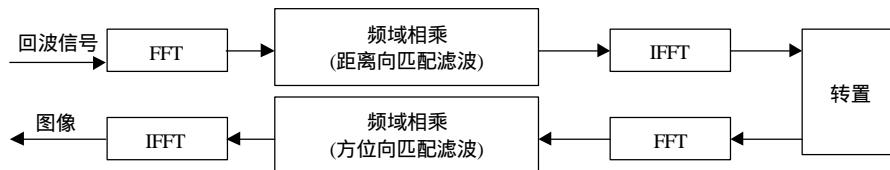


图4 SAR成像处理的结构功能框图

一帧原始数据(4 096*4 096个复数点)经过本系统硬件距离向、方位向处理,可得到如图5所示成像结果,同样原始数据经过软件处理得到的成像结果如图6所示。两个结果几乎完全一样,但硬件成像速度远远快于软件成像速度,由此证明该FFT处理系统在星载SAR实时成像处理中得到了成功应用。



图5 硬件成像结果



图6 软件成像结果

4 结束语

本文对基于FPGA的超高速FFT运算的硬件实现做了详细的介绍,设计并完成了该算法的硬件实现系统。硬件运行结果表明,本文提出的大点数超高速FFT系统是切实可行的,获得了比较满意的效果。随着可编程器件规模、速度的不断提高,采用高档FPGA实现大点数超高速FFT运算越来越具有可行性,并且在体积、速度、灵活性等方面具有很高的优越性。

参 考 文 献

- [1] 胡广书. 数字信号处理-理论、算法与实现[M]. 北京: 清华大学出版社, 1997
- [2] Stratix programmable logic device family[S]. Ver.2.1. Altera Corporation, 2002
- [3] 邹 琪, 皮亦鸣, 黄顺吉. 极化SAR图像的多纹理最大似然估计[J]. 电子科技大学学报, 2001, 30(2): 120-123

编 辑 徐安玉