

星载 SAR 实时成像处理器的 FPGA 实现

熊君君^{1,2}, 王贞松^{1,2}, 姚建平^{1,2}, 石长振^{1,2}

(1. 中国科学院计算技术研究所, 北京 100080; 2. 中国科学院研究生院, 北京 100039)

摘要: 本文提出了一种用 FPGA 实现星载合成孔径雷达实时成像处理器的方法, 用来实现星载 SAR 的 CS 算法(或 RMA 算法). 该实时成像处理器由 7 片 Xilinx 公司的商业 FPGA 实现, 其中 4 片作为并行的处理单元; 一片为 CS 因子的生成单元; 一片为 SDRAM 控制单元; 一片为系统的控制单元. 该系统将流水处理和并行处理相结合, 从而极大的减少了处理时间. 同时根据算法各运算对数据的精度要求不同, 将浮点运算和定点运算结合在一块, 减少了硬件开销. 该系统工作在 100MHz 时, 33 秒左右能完成 16k*16k 星载样本点的成像, 并对加拿大 Radarsat 的雷达原始信号进行成像处理, 成像质量能达到要求.

关键词: CS 算法; 实时成像处理器; FPGA; 流水处理; 并行处理

中图分类号: TN957.52 **文献标识码:** A **文章编号:** 0372-2112(2005)06-1070-03

The FPGA Design of on Board SAR Real Time Imaging Processor

XIONG Jun jun^{1,2}, WANG Zhen song^{1,2}, YAO Jian ping^{1,2}, SHI Chang zhen^{1,2}

(1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China;

2. Graduate School of the Chinese Academy of the Sciences, Beijing 100039, China)

Abstract: This paper designs the on board SAR real time imaging processor using seven pieces of Xilinx FPGA, which consists of four parallel process units, one CS factor unit, one SDRAM controller unit and one control unit. The system combines pipeline process and parallel process to reduce process time, also combines fixed point operation and floating point operation to reduce hardware resource. This system can make image within 33 seconds when works in 100MHz, and the imaging quality can fit for the requirement by testing with the Radarsat raw data.

Key words: CS arithmetic; real time imaging processor; FPGA; pipeline process; parallel process

1 引言

星载合成孔径雷达(SAR)^[1,2]可以全天时全天候工作,并能高分辨率大面积成像,已成为对地观测的重要手段.由于星载 SAR 所产生的原始回波数据量很大,数据不能及时向上传输,所以数据压缩成为减少数据量的重要手段.由于星载 SAR 的原始回波数据的相关性很差,压缩比一般只能达到 1:4,而实时成像后的图像数据,压缩比可以达到 1:16 或更高,可以减少数据量,从而解决星载 SAR 数据的传输瓶颈.所以星上的实时成像处理器,对星载合成孔径雷达至关重要.商用 FPGA 随着工艺的发展,规模越来越大(最大的达到 1000 万逻辑门),处理能力越来越强,同时 FPGA 有可配置的优点,可以根据成像算法映射得到适于 SAR 成像的计算体系结构,可以提供比 DSP 更高的并行度,从而提高处理速度,而且 FPGA 可以根据不同运算对精度的不同要求将定点运算和浮点运算相结合,从而减少系统的硬件开销.适于星载 SAR 的成像算法有:CS(Chirp Scaling)^[3]和 RMA(Range Migration Algo-

rihm)算法,且两者的流程有很强的相似性,这样就可能具有相同的体系结构,本文以 CS 为例加以阐述.

2 CS 算法的原理

CS 算法适用于星载 SAR 的大距离徙动高分辨率成像处理,且该算法不需要插值处理,仅通过复数乘和 FFT 就可以完成精确成像处理.该算法首先通过方位向傅立叶变换将回波信号变换到距离多普勒域(方位频率),与 Chip Scaling 因子 Φ_1 相乘;再通过距离向傅立叶变换将信号变换到二维频率(又称为波数

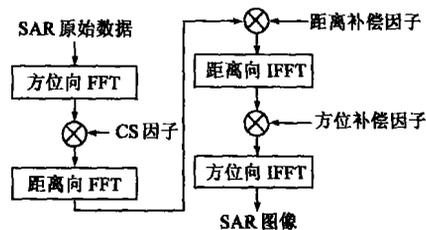


图 1 CS 算法流程图

域),与距离补偿因子 Φ_2 相乘;然后进行距离向傅立叶逆变换将信号变换回距离多普勒域,与方位补偿因子 Φ_3 相乘,完

收稿日期: 2004-04-30; 修回日期: 2004-09-27

基金项目: 国家自然科学基金(No. 60303017); 国家高技术研究发展计划(863 计划)课题(No. 2003AA135093).

成方位处理;最后利用方位向傅立叶逆变换将信号变换回时域,得到 SAR 图像.对算法进行分析,发现算法涉及的运算有:FFT (IFFT) 运算;三个 CS 因子 Φ_1 、 Φ_2 、 Φ_3 的生成;FFT (IFFT) 运算后的结果与三个 CS 因子的复数乘运算.其成像处理的流程图见图 1.

3 体系结构的建立

通过对算法的分析,将该算法映射到对应的体系结构,则体系结构由下面的几个单元组成:一是处理单元,完成 FFT (IFFT) 以及与 CS 因子的复数乘运算,采用四个并行的处理单元,每个处理单元由 Xilinx 公司的 xc2v1000 实现;二是 CS 因子的生成单元,该单元生成对应的三个 CS 因子,并将生成的因子分发到四个处理单元,该单元由 Xilinx 公司的 xc2v500 实现;三是 SDRAM 控制单元,该算法中涉及到距离向和方位向的交错,需要对数据进行转置处理,可以对数据进行跳变存储,从而将数据的转置和读写融合在一块,这样能节省数据的转置时间,该单元由 Xilinx 公司的 xc2v250 实现;四是系统的控制单元用来完成该系统与外界交互和协调板上各单元的有序工作,该单元由 Xilinx 公司的 xc2v500 实现.该系统有两条独立的总线:一条是数据总线,用来传送数据;一条是 CS 因子总线用来分发 CS 因子.两条总线相互独立,可以完全并行的运算.根据上面的分析,建立的体系结构见图 2 所示.

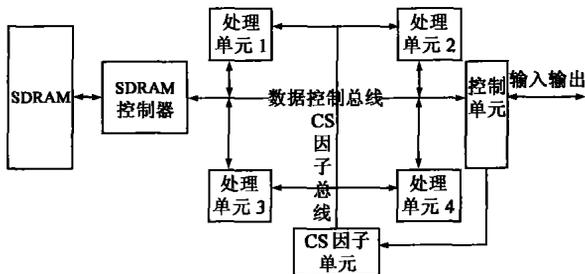


图 2 体系结构原理图

4 处理单元的流水实现

处理单元完成数据的 FFT (IFFT) 运算与三个 CS 因子的复数乘运算.对于 16k 样本点的 FFT (IFFT) 采用频率抽取的基 4 算法^[4],其表达式如下:

$$X(4r) = \sum_{n=0}^{N/4-1} [(x(n) + x(n + N/2)) + (x(n + N/4) + x(n + 3N/4))] W_N^{rn} \quad (1)$$

$$X(4r+2) = \sum_{n=0}^{N/4-1} [(x(n) + x(n + N/2)) - (x(n + N/4) + x(n + 3N/4))] W_N^{2n} W_N^{rn} \quad (2)$$

$$X(4r+1) = \sum_{n=0}^{N/4-1} [(x(n) - x(n + N/2)) - j(x(n + N/4) - x(n + 3N/4))] W_N^n W_N^{rn} \quad (3)$$

$$X(4r+3) = \sum_{n=0}^{N/4-1} [(x(n) - x(n + N/2)) + j(x(n + N/4) - x(n + 3N/4))] W_N^{3n} W_N^{rn} \quad (4)$$

从上面的四个式子可以发现,基四 FFT 可以分为两步,一步

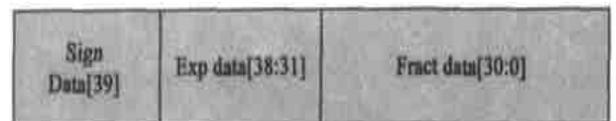
是四点的 FFT,对应于上式中的四点为 $x(n)$ 、 $x(n + \frac{N}{2})$ 、 $x(n + \frac{N}{4})$ 、 $x(n + 3\frac{N}{4})$,然后是四点运算的结果与四个旋转因子 (W_N^0 、 W_N^n 、 W_N^{2n} 、 W_N^{3n}) 的复数乘运算.考虑到 16k 的样本点要迭代 7 次,且第 7 次迭代只需完成四点的 FFT,不需要与旋转因子做复数乘运算.运算后的结果与 CS 因子做复数乘运算,所以可以将 CS 因子看成 FFT 第 7 次迭代的旋转因子,这样就将 FFT (IFFT) 处理与 CS 因子的复数乘形成了流水处理.该流水处理实际将 FFT (IFFT) 的运算时间与 CS 因子复数乘的时间融合在一块,减少了处理时间.考虑到对精度的要求,FFT 采用定点完成,且实部和虚部各 32 位,这样即满足了精度要求同时又发挥了定点运算时钟频率比较高的优点,实际测试时该单元能稳定工作在 100MHz.

5 SDRAM 控制器的实现

对于 CS 算法,每一次距离向和方位向之间的变化都需要进行数据转置,所以总共需要三次数据转置,用专用的转置板来完成转置功能在时间上是不现实的.采用地址的跳变存储,将数据按一定的映射规律读和写,将数据的转置和读写结合在一块,能节省时间.以 16k * 16k 样本点和 8192 行、4096 列的 512Mbit 内存颗粒为例来阐述具体的读写过程.对于距离向的读写,每次打开两个连续的 bank,例如 bank 0 和 bank 1.每一个 bank 写一条距离向数据中连续的 64 点数据,一个 bank 写完后,写下一个打开的 bank,一行写完后写下一行.第二条距离向的数据将从 64 列开始,按照上面的规律写.这样 bank 0 和 bank 1 的前 128 行可以写 64 条距离向的数据.将距离向的第 65 到第 128 条数据按照上面的规律写到 bank 2 和 bank 3.这样就相当于该内存颗粒的前 128 行所写的数据来自距离向的前 128 条数据.以 128 为一个小的单元,将如下的数据写到剩下的空间.对于方位向的读写,每一次打开两个相隔的 bank,例如 bank 0 和 bank 2.每一个 bank 以 64 为跳变的间隔读写 64 点的数据,其列的变化如下所示:0、64、128、192、256 等,读出的数据对应于 64 条距离向数据的第一点,也就是方位向数据的前 64 点.然后按此规律读写打开的另外一个 bank 的数据.这样就从 bank 0 和 bank 2 分别读写了 64 点数据,以此规律读写下一行的数据.该单元由 Xilinx 公司的 xc2v500 实现,能稳定工作在 100MHz.

6 CS 因子的浮点实现

CS 成像算法中 CS 因子的精度对成像质量有很大的影响,所以高分辨率成像对 CS 因子的计算精度有很高的要求,同时因子的计算也涉及到乘除或开方等超越运算,所以该运算单元采用浮点实现,同时为了提高精度采用 40 位的浮点格式,该格式与 IEEE754^[5] 部分兼容,其格式如下:



最高位是符号位,1 表示负数,0 表示正数,8 位指数位,31 位

的尾数位, 这样比 32 位的浮点数多了 8 位的尾数, 从而可以提高精度. 在计算中用到了浮点的加、减、乘、除、开方、 \sin (\cos) 等运算. 对于除法、开方等运算可以按泰勒展开等近似运算代替, 必不可少的运算是浮点的加减和乘法运算. 加法运算采用三级流水的结构来提高工作频率. 乘法运算涉及到 32 位尾数的相乘运算, 32 位乘法运算是一个很耗时的运算, 是该系统工作频率的一个瓶颈, 为了提高工作频率, 将该运算分解成四个 16 位的乘法运算, 这样就极大的提高了系统的工作频率. 得到的 CS 因子表示的是角度, 可以用 2^{20} 表示 360 度, 从而将浮点的 CS 因子转换成 32 位的定点数. 该单元由 Xilinx 公司的 xc2v500 实现, 由于采用的高精度的浮点运算, 能稳定工作在 75MHz.

7 实际应用

应用上述方法设计的处理板对雷达数据进行实时成像处理, 当样本为 $16k * 16k$, 且实部和虚部各为 32 位时, SDRAM 空间为 2GByte. 当处理单元工作在 100MHz 时, 完成整个的处理需要 33 秒左右. 图 3 为对加拿大斜距分辨率为 5 米的 Radarsat 原始数据进行处理后形成的图像, 该成像区域为颐和园的天鹅湖区域, 图中的三个黑色区域为颐和园的三个湖, 黑色区域上面的亮点为湖上的小岛, 整个的外形看起来很像一只天鹅, 成像的质量与使用一般的 32 位通用计算机地面成像处理成像没有差别, 可见该系统在时间和精度上都能满足成像要求.



图 3 颐和园成像图

8 总结

本文首先对 SAR 成像算法——CS 算法的原理进行了分析, 根据该算法映射出一种能高效实现该算法的体系结构,

并用 Xilinx 公司的商用 FPGA 实现. 该处理系统由 7 片 FPGA 完成, 其中 4 片为并行的处理单元, 1 片为 CS 因子单元, 1 片为 SDRAM 控制单元, 1 片为系统的控制单元. 该系统根据算法各运算的时间关系将流水处理和并行处理巧妙的结合在一块, 从而极大的提高了处理速度, 同时该系统根据算法各运算对精度的不同要求将定点运算和浮点运算结合在一块, 在保证运算精度的情况下, 减少了硬件开销. 最后, 应用设计的处理板对星载雷达数据进行实时成像处理, 成像质量能达到要求.

参考文献:

- [1] 张澄波. 综合孔径雷达: 原理、系统分析与应用[M]. 北京: 科学出版社, 1989. 38- 45.
- [2] 魏钟铨, 王贞松, 陈元藻, 李春生, 等. 合成孔径雷达卫星[M]. 北京: 科学出版社, 2001. 53- 60.
- [3] 黄岩. 高分辨率星载合成孔径雷达成像处理技术研究[M]. 北京: 北京航空航天大学, 1999. 43- 50.
- [4] 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, 1998. 143- 150.
- [5] John P. Hayes. Computer Architecture and Organization[M]. 北京: 清华大学出版社, 2001. 266- 270.

作者简介:



熊君君 男, 1977 年 4 月生于湖北省天门市, 现为中科院计算所系统结构室博士, 主要研究方向为实时信号处理和芯片设计. E-mail: xjj@ict. ac. cn.



王贞松 男, 1945 年 11 月生于浙江省奉化, 现为中科院计算所系统结构室教授、研究员、博士生导师, 从事 SAR 系统设计、实时信号处理与计算机通信.