

离散傅立叶变换的 CPLD 快速实现

何伟¹, 杨宝强², 唐斌¹

(1. 电子科技大学, 四川 成都 610054; 2. 空军工程大学 训练部, 陕西 西安 710051)

摘要:基于离散傅立叶变换系数的性质,利用 FPGA 仿真软件,提出了一种离散傅立叶变换的快速硬件实现技术。8 点离散傅立叶变换的时间在 200 ns 左右。

关键词:离散傅立叶变换;CPLD 仿真;快速实现

中图分类号:TN602 **文献标识码:**A **文章编号:**1009-3516(2003)06-0078-04

信号处理中,离散傅立叶变换(DFT)是最常用的处理方法之一,具有很高的使用频率。但是,在进行信号处理的工程实现时,由于傅立叶变换运算量较大,处理时间较长,不能满足实时要求,从而限制了其应用。一般硬件实现傅立叶变换都由专用的 DSPS 数字处理芯片完成,常见的如 TI 公司的 TMS320 系列,其 DFT 或 FFT 都由 C 或其他算法语言编程后实现,虽然最新的芯片指令周期可以在 100 ns 量级,但 8 点 FFT 变换时间也在毫秒量级^[1]。在实时性要求较高的环境下,上述速率仍不能满足运算要求。CPLD 是一种目前比较新颖的复杂用户可编程器件,具有集成度高、速度快等优点,在信号处理硬件实现方面越来越受到广大设计和工程人员青睐。目前用 CPLD 实现离散傅立叶变换运算虽然有一些研究成果,但实现时间到底有多少?实现的结构如何?这些问题尚未见具体报道。本文直接利用 DFT 系数性质,并用 FPGA 仿真语言 Verilog 及新型的 CPLD 器件,完成离散傅立叶变换的快速硬件实现,该方法具有实现时间短,硬件结构简单特点。

1 离散傅立叶变换及 FPGA 实现

1.1 离散傅立叶变换分析

采样信号 $x(n)$, $n=0,1,2,\dots,n$ 的离散傅立叶变换为 $X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}$, $k=0,L,N-1$ 。其中 $W_N = e^{-j2\pi/N}$ 。设数据点为 $x(n)$, $n=0,1,2,\dots,7$,其离散傅立叶变换为 $X(k) = \sum_{n=0}^7 x(n) W_8^{kn}$, $k=0,L,7$ 。用矩阵表示,有 $X=Hx$ 。其中, $X=[X(0),X(1),L,X(7)]^T$, $x=[x(0),x(1),L,x(7)]^T$

$$H = \begin{bmatrix} 1 & 1 & L & 1 \\ 1 & e^{-j2\pi\frac{1}{8}} & L & e^{-j2\pi\frac{7}{8}} \\ M & M & M & M \\ 1 & e^{-j2\pi\frac{7}{8}} & L & e^{-j2\pi\frac{49}{8}} \end{bmatrix}$$

H 为 8×8 系数矩阵,其具体值头四列为

1.000 0	1.000 0	1.000 0	1.000 0
1.000 0	0.707 1 - 0.707 1i	0.000 0 - 1.000 0i	-0.707 1 - 0.707 1i
1.000 0	0.000 0 - 1.000 0i	-1.000 0 - 0.000 0i	-0.000 0 + 1.000 0i
1.000 0	-0.707 1 - 0.707 1i	-0.000 0 + 1.000 0i	0.707 1 - 0.707 1i
1.000 0	-1.000 0 - 0.000 0i	1.000 0 + 0.000 0i	-1.000 0 - 0.000 0i
1.000 0	-0.707 1 + 0.707 1i	0.000 0 - 1.000 0i	0.707 1 + 0.707 1i
1.000 0	-0.000 0 + 1.000 0i	-1.000 0 - 0.000 0i	0.000 0 - 1.000 0i

收稿日期:2003-04-23

作者简介:何伟(1972-),男,四川成都人,讲师,博士生,主要从事通信与信息系统研究。

1.000 0	0.707 1 + 0.707 1i	-0.000 0 + 1.000 0i	-0.707 1 + 0.707 1i
后四列:			
1.000 0	1.000 0	1.000 0	1.000 0
-1.000 0 - 0.000 0i	-0.707 1 + 0.707 1i	-0.000 0 + 1.000 0i	0.707 1 + 0.707 1i
1.000 0 + 0.000 0i	0.000 0 - 1.000 0i	-1.000 0 - 0.000 0i	-0.000 0 + 1.000 0i
-1.000 0 - 0.000 0i	0.707 1 + 0.707 1i	0.000 0 - 1.000 0i	-0.707 1 + 0.707 1i
1.000 0 + 0.000 0i	-1.000 0 - 0.000 0i	1.000 0 + 0.000 0i	-1.000 0 - 0.000 0i
-1.000 0 - 0.000 0i	0.707 1 - 0.707 1i	-0.000 0 + 1.000 0i	-0.707 1 - 0.707 1i
1.000 0 + 0.000 0i	-0.000 0 + 1.000 0i	-1.000 0 - 0.000 0i	-0.000 0 - 1.000 0i
-1.000 0 - 0.000 0i	-0.707 1 - 0.707 1i	-0.000 0 - 1.000 0i	0.707 1 - 0.707 1i

从系数矩阵 H 可以看出,离散傅立叶变换乘法次数很多,但参与乘法的数据实际上除数据点以外,只有 $\pm 0.707 1$ 。我们在硬件实现时,可以只设计一个被乘数固定为 $0.707 1$ 的乘法器件。16 点傅立叶变换也有类似的情况,只是参与乘法的除 $\pm 0.707 1$ 外,还有 $0.382 7$ 、 $0.923 9$ 。

1.2 硬件实现方案

基于上述特性,本文采用图 1 的硬件实现方案。

采样数据存储以后,可以对应 $X(k)$ 进行 8 路并行处理:

第一路 $X(0)$,第三路 $X(2)$,第五路 $X(4)$,第七路 $X(6)$ 为 8 个数据点的加减运算;其余路出现有乘法处理。



图 1 硬件实现原理框图

减法可以通过二进制补码运算实现。并行处理中乘法器可以调用软件 MAXPLUS II 自带的参数化模块库的参数化乘法器 `lpm_mult` 来实现^[2]。将乘法器的一个输入端 `dataa[]` 为采样数据点输入端,另一个输入端设为常量(即 $0.707 1$),即可完成 8 点傅立叶变换乘法运算。以 $X(1)$ 为例,有 $X(1) = x(0) + x(1)[0.707 1 - 0.707 1i] + x(2)(-1i) + x(3)[-0.707 1 - 0.707 1i] + x(4)(-1) + x(5)[-0.707 1 + 0.707 1i] + x(6)i + x(7)[0.707 1 + 0.707 1i]$ 。经整理,可得 $X(1) = x(0) - x(4) + [x(1) - x(3) - x(5) + x(7)]0.707 1 - \{x(2) - x(6) + [x(1) + x(3) - x(5) - x(7)]0.707 1\}i$ 。

从上式可知,一路含乘法的运算实际只需要 2 次乘法即可。这种方法与 FFT 比较起来在减少运算量方面效果是相同的,而且实现结构相当简单。在硬件实现时,数据的实数部分和虚数部分同时并行处理。每一路比原始运算减少 6 次乘法运算,只增加了 2 次加法运算。在 FPGA 设计中,一个 8 位加法器消耗 14 个逻辑单元 `Lcs`,而一个 8 位乘法器消耗约 150 个逻辑单元 `LCs`。节约的资源很可观。而且在 FPGA 设计中加法运算速度很快(一般为几个 ns),相对减少的乘法器数目,节约的资源来说,增加的时间是可以接受的。

1.3 仿真时序

设 8 个采样数据点为 $0.821 4, 0.444 7, 0.615 4, 0.791 9, 0.921 8, 0.738 2, 0.176 3, 0.405 7$, MATLAB 中离散傅立叶变换结果为 $4.915 5, -0.581 1 - 0.504 7i, 0.951 5 + 0.014 7i, 0.380 2 + 0.373 6i, 0.154 4, 0.380 2 - 0.373 6i, 0.951 5 - 0.014 7i, -0.581 1 + 0.504 7i$ 。

$x(0) - x(4) + [x(1) - x(3) - x(5) + x(7)] \times 0.7071$ 为 $X(1)$ 的实部。应用前述方法得到的仿真时序见图 2、图 3。

图中所示二进制代码 $1111, 1000, 0101, 1000, 1110$ 为 $-0.581 1$ 的二进制代码原码表示。(虚数部分仿真时序与实数部分类似,略去)。

从图中可以看出,实现离散傅立叶变换的一路含乘法运算只需 150 ns 左右。整个 8 位 DFT 变换可综合到 FLEK10kA 的 EPF10K50AGC599-1 器件中。

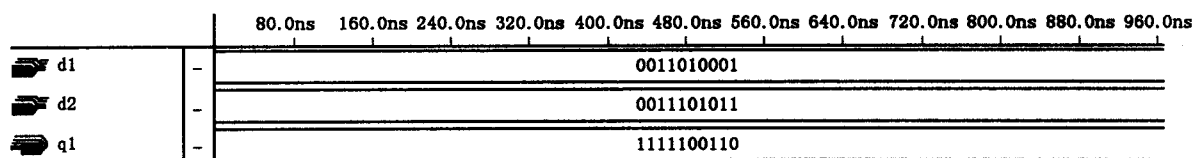


图 2 $x(0) - x(4)$ 的仿真时序(二进制表示)

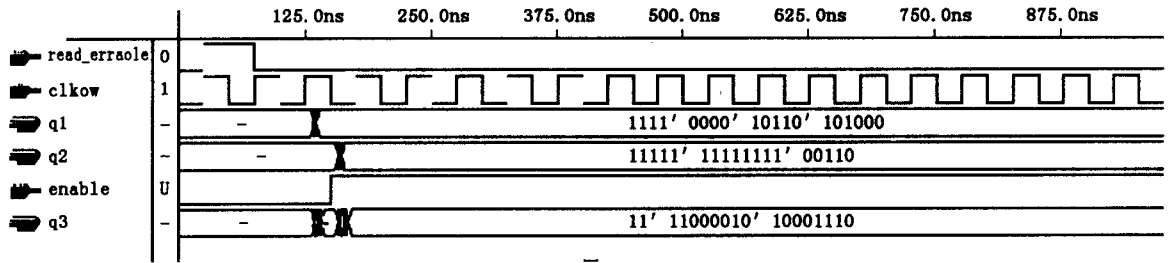


图3 $x(0) - x(4) + [x(1) - x(3) - x(5) + x(7)] \times 0.7071$ 的仿真时序

2 硬件仿真的改进

用 FPGA 实现离散傅立叶变换时,我们发现受器件输入时钟等各方面条件的限制,实际的一个乘法器的运算速度在 50 ns 左右,而且整个运算处理的速度受乘法器处理速度限制,要想提高运算速度,就必须从提高乘法器本身的运算速度着手。

同时 FPGA 设计中,一个 8 位加法器消耗 14 个逻辑单元 Lcs,而一个 8 位乘法器消耗 160 个逻辑单元 LCs 左右,从节约器件资源的角度要求尽可能少用乘法器。

以 0.48×0.7071 为例,即假设 0.48 为采样数据,用 8 位二进制数来表示:0.48 为 01111010,0.7071 表示为 10110100。用参数化乘法器 lpm_mult 方法可得乘法结果,见图 4。

乘法器乘法时间为 82 ns(时钟频率为 100 ns),消耗器件的逻辑单元 LCs 156 个。

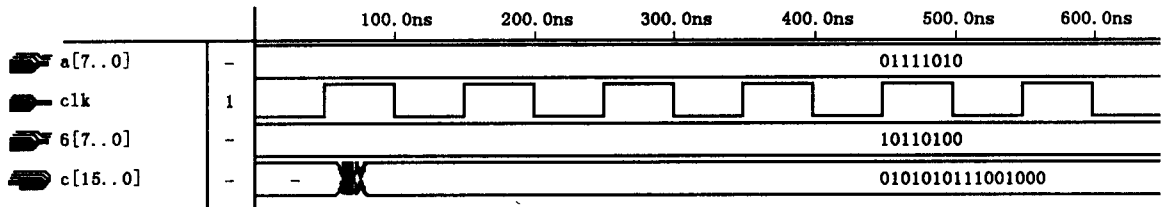


图4 0.48×0.7071 仿真时序

二进制乘法可以用移位加的方法实现,如 01111010 乘以 10110100,可以表示为 01111010 左移 7 位 + 01111010 左移 5 位 + 01111010 左移 4 位 + 01111010 左移 2 位。结果为 0101010111001000 完全一样。仿真时序见图 5。完成乘法所需时间仅为 35 ns,消耗器件的逻辑单元 LCs 53 个。用移位加代替乘法器是完全可行的,既节约器件资源又可以提高系统运算速度。

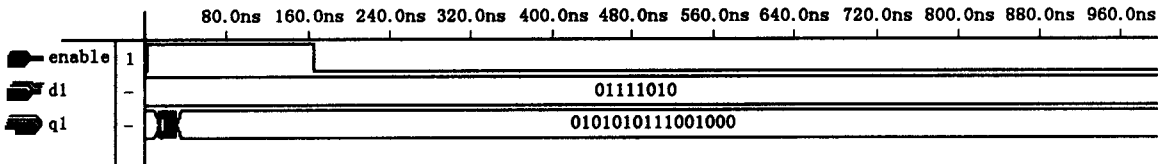


图5 二进制乘法仿真时序

3 结论

本文直接利用离散傅立叶变换系数,实现了离散傅立叶变换在 CPLD 器件的快速实现,实现时间在 200 ns 左右,满足实时要求,对工程运用有很好的实际指导意义。另外这种方法是 以 2 为基的实现,对于非 2 的整数倍的数据点,可以通过在数据点后补零的方法完成。更高位数的离散傅立叶变换可以在此基础上,利用离散傅立叶变换系数对称的性质来实现。

参考文献:

[1] 宋万杰,罗 丰,吴顺君. CPLD 技术及其应用[M]. 西安:西安电子科技大学出版社,1999.
 [2] 夏宇闻. 复杂数字电路与系统的 Verilog Hdl 设计技术[M]. 北京:北京航空航天大学出版社,1999.

[3] 梁虹. 信号与系统分析及 MATLAB 实现[M]. 北京:电子工业出版社,2002.

(编辑:姚树峰)

A Method of DFT Fast Hardware Realization with CPLD

HE Wei¹, YANG Bao - qiang², TANG Bin¹

(1. University of Electronic Science and Technology of China, Chengdu, Sichuan 610054, China; 2. The training Department of Air Force Engineering University, Xi'an, Shaanxi 710051, China)

Abstract: By exploring the coefficients of the discrete Fourier transform (DFT), a method of DFT fast hardware realization by using CPLD devices is proposed in this paper. 8 - point DFT realization by using this method takes about 200ns.

Key words: DFT; CPLD; fast realization

(上接第 43 页)

[2] 姬红兵,高新波,谢维信. 一种编队目标架次检测方法[J]. 系统工程与电子技术,2000,22(6):62-65.

[3] Cohen L. Time - Frequency Analysis: Theory and Applications[M]. America:Prentice Hall,1995.

[4] 胡昌华. 基于 Matlab 的系统分析与设计—时频分析[M]. 西安:西安电子科技大学出版社,2001.

(编辑:门向生)

A Study of Methods in Identifying Targets Sortie in Formation

SHI Ding - jie, YAO Pei - yang, DU Qin - feng

(The Telecommunication Engineering Institute, Air Force Engineering University, Xi'an, Shaanxi 710077, China)

Abstract: In view of the defect that a conventional low - resolution radar fails to directly resolve targets flying in formation in both range and azimuth, the radar echo signals of multi - targets in formation are analyzed for time - frequency characteristics. A method is proposed, which is used for extracting the sortie information from the formation target radar echo on the basis of smoothed pseudo - WVD distribution, and such a method is used not only for identifying target sortie number reliably, but also for resisting noise and interference. The experimental results illustrate that the efficiency of the method is superior to that of the conventional WVD distribution.

Key words: low - resolution radar; target sortie identification; time - frequency analysis