

文章编号: 1001-3806(2016)02-0161-05

少模光纤通信频域均衡中的大点数 FFT 设计

黄战华, 赵宇璐, 李桂芳, 王云立

(天津大学 精密仪器与光电子工程学院 光电信息技术教育部重点实验室, 天津 300072)

摘要: 为了减小频域均衡系统电路实现的功耗和面积, 满足长距离少模光纤通信对均衡器的要求, 对关键环节快速傅里叶变换(FFT)电路的实现进行了研究, 采用2维分解算法将大点数的FFT运算转换为小点数FFT处理器的设计, 降低了硬件复杂度。设计了基于现场可编程门阵列的高速蝶形运算核, 实现了16384点FFT的2维R2²SDF结构, 提高存储器的资源利用率, 减少了复数乘法器的使用; 进行了理论分析和实验验证, 取得了不同时钟频率下的电路结构占用资源的数据。结果表明, FFT运算器的正确性得到验证, 该FFT运算器能够适应少模光纤通信系统中优化频域均衡电路结构的要求, 能够实现200MHz数据传输速度的频域均衡实时处理。

关键词: 光通信; 快速傅里叶变换; 现场可编程门阵列; 频域均衡; 大点数

中图分类号: TN929.11 **文献标志码:** A **doi:** 10.7510/jgjs.issn.1001-3806.2016.02.003

Long point FFT design for frequency domain equalization of few-mode fiber communication

HUANG Zhanhua, ZHAO Yulu, LI Guifang, WANG Yunli

(Key Laboratory of Opto-electronics Information Technology of Ministry of Education, School of Precision Instrument and Opto-electronics Engineering, Tianjin University, Tianjin 300072, China)

Abstract: To reduce the circuit's power and size of frequency domain equalization (FDE) system, and meet the requirements of FDE in long-distance few-mode fiber communication, the circuit for fast Fourier transformation (FFT), the key of FDE, was studied. The long point FFT operation was converted to short FFT by means of 2-D decomposition algorithm, and the hardware complexity was reduced. High-speed butterfly core based on field-programmable gate array (FPGA) was designed to implement 2-D R2²SDF structure of 16384 point FFT, improve the resource utilization of memory and reduce the usage of complex multipliers. Through theoretical analysis and experiments, resource consumption information of the circuit under different clock frequencies was collected. The result show that the designed FFT can satisfy the optimization requirement of FDE circuit structure in few-mode fiber communication and realize real-time processing of 200MHz data speed in FDE.

Key words: optical communication; fast Fourier transformation; field-programmable gate array; frequency domain equalization; long points

引言

在少模光纤(few-mode fiber, FMF)通信系统中使用模分复用技术能够扩展信道, 可以突破单模光纤信道容量提升的极限, 是光纤通信领域研究的热点^[1-2]。在少模光纤通信系统中, 模式信道的增加导致传输中存在较大的色散^[3], 信道均衡对少模光纤通信尤为重要^[4]。与采用卷积的时域均衡相比, 频域均衡(fre-

quency domain equalization, FDE)具有运算简单、快速的优势^[5]。而在使用 D 个模式传输的少模光纤通信系统中, 均衡器包含 $D \times D$ 的自适应线性频域滤波器矩阵^[6]。每个模式信道都需要进行 $4 + 4D$ 次快速傅里叶变换(fast Fourier transformation, FFT), 其中2个FFT用于输入, 一对FFT/IFFT用于更新循环, $4D$ 个FFT用于梯度估计模块^[7], 频域均衡系统的复杂度主要取决于FFT的复杂度^[8]。因此FFT是少模光纤通信频域均衡系统的关键环节, 其实现方式和电路复杂度是制约FDE电路实现的主要因素。

在少模光纤中, 由模式群延时决定了均衡器的数据块长度, 也决定了FFT运算器的点数。在前期的研究中, 进行了50km的少模光纤通信实验, 使用6个模式进行传输, 模式群延时为3.15ns, 符号率为28G, 采

基金项目: 国家九七三重点基础研究发展计划资助项目(2014CB340100)

作者简介: 黄战华(1965-), 男, 博士, 教授, 主要从事光电图像和光电子信息技术等方面的教学和科研工作。

E-mail: zhanhua@tju.edu.cn

收稿日期: 2015-01-30; 收到修改稿日期: 2015-04-01

用 256 点的 FFT 就可以满足均衡器数据块的长度要求^[9-10]。而随着传输距离的增大,模式群延时增加,需要的数据块长度也增大,当传输距离增大到 2000km 时,需要对长度为 16384 的数据块进行 FFT 运算。而传统的通信系统中,由于采用分时传输,数据块长度小,通常采用短点数 FFT,如用于正交频分复用^[11] (orthogonal frequency division multiplexing, OFDM) 系统的 256 点 FFT^[12], 1024 点以下可配置的 FFT^[13] 等,无法满足少模光纤通信系统对均衡器长度的需求,同时传统的 FFT 设计方法占用了大量的逻辑资源,不利于频域均衡的电路实现。针对这一问题,本文中介绍了一种用于少模光纤通信频域均衡的 16384 点 FFT 运算器,利用 2 维分解算法降低了 FFT 硬件复杂度,基于现场可编程门阵列 (field-programmable gate array, FPGA)^[14] 设计了高速蝶形运算核实现基为 2² 的单路延迟反馈 (radix-2² single-path delay feedback, R2²SDF) 结构,减少了复数乘法器的使用,提高了存储器的资源利用率,适应少模光纤通信频域均衡系统中减少硬件资源使用的要求。

1 FFT 的 2 维分解算法和总体设计

设序列 $x(n)$ 的长度为 N , 且 N 为 2 的自然数次幂,其离散傅里叶变换 $X(k)$ 的定义式^[15] 为:

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, (k = 0, 1, \dots, N-1) \quad (1)$$

式中, $W_N^{nk} = \exp\left(-j \frac{2\pi}{N} nk\right)$, 称其为旋转因子。若 $N = L \times M$, 将 $x(n)$ 重新排序为 L 行 M 列的矩阵。设:

$$\begin{cases} n = Ml + m, \\ (m = 0, 1, \dots, M-1; l = 0, 1, \dots, L-1) \\ k = Lr + s, \\ (s = 0, 1, \dots, L-1; r = 0, 1, \dots, M-1) \end{cases} \quad (2)$$

将 n, k 代入 $X(k)$ 的表达式, 根据旋转因子 W_N^{nk} (nk 是幂次方, 下同) 的三角函数性质, 可将原式写为:

$$X(Lr + s) = \sum_{m=0}^{M-1} \left\{ \underbrace{\left[\sum_{l=0}^{L-1} x(Ml + m) W_N^{sl} \right]}_{X_1} W_N^{ms} \right\} W_M^{mr} \quad (3)$$

由上式可知, X_1 是一个 L 点的 FFT, 一共 M 个。而 X_2 是一个 M 点的 FFT, 一共 L 个。这样就把一个基于 1 维处理的 FFT 运算转换为两个较短点数 FFT 的 2 维处理, N 点 FFT 2 维分解的硬件复杂度 C_{2-D} 可表示为行列变换复杂度之和:

$$C_{2-D} = \underbrace{C_M}_1 + \underbrace{C_L}_2 + \underbrace{N}_3 = \frac{M}{2} \log_2 M + \frac{L}{2} \log_2 L + N =$$

$$\frac{1}{2} \left(M - \frac{N}{M} \right) \log_2 M + \frac{N}{2M} \log_2 N + N \quad (4)$$

式中, C_M 和 C_L 分别表示行列变换的复杂度, 第 3 项的 N 表示行列变换之间数列与旋转因子 W_N^{ms} 的 N 次乘法运算。 N 值确定后, $M = L = N^{1/2}$ 时, 其复杂度最小, 并且行变换和列变换可进行复用, 复杂度可降为:

$$C_{2-D} = \frac{\sqrt{N}}{4} \log_2 N + N \quad (5)$$

在少模光纤通信频域均衡系统中要求 FFT 长度为 $N = 16384$, 根据 2 维分解的思想将其分解为 128×128 , 复用 128 点 FFT 处理器电路结构如图 1 所示。

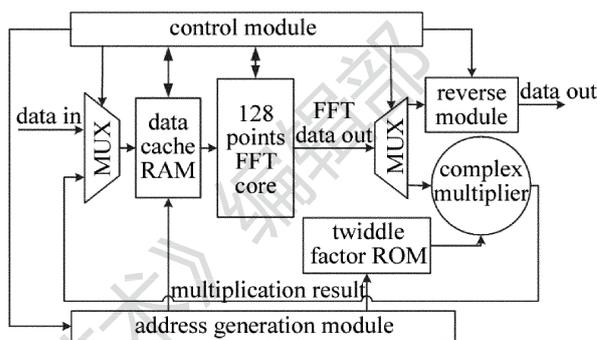


Fig. 1 FFT operator structure

输入数据按 128 行 128 列排列后, 通过数据选择器进入 128 点 FFT 变换器, 先进行 128 次列变换, 变换结果与相应的旋转因子进行复数乘法运算, 乘积进入数据缓存单元进行重新排列。所有的列变换完成后, 数据由缓存单元再次进入 128 点 FFT 变换器, 变换结果经过倒序输出即为 16k 点的 FFT 结果。旋转因子直接存放在 ROM 中, ROM 读出地址和数据缓存 RAM 读写地址由地址产生模块控制。控制信号模块产生输入输出数据选择, 倒序模块以及地址产生模块的时序信号。

2 128 点 FFT 处理器的 R2²SDF 结构

由上述讨论, 将一个 16k 点的 FFT 转换为 128 点的 FFT 设计, 针对 128 点的 FFT 处理器可以采用 R2²SDF 结构, 减少复数乘法器的使用。

N 点的数列, 将 n, k 用线性组合表示, 则:

$$\begin{cases} n = \left\langle \frac{N}{2} n_1 + \frac{N}{4} n_2 + n_3 \right\rangle \\ k = \langle k_1 + 2k_2 + 4k_3 \rangle \end{cases} \quad (6)$$

式中, n_1, n_2, k_1, k_2 的取值范围为 $[0, 1]$, n_3, k_3 的取值范围为 $[0, N/4)$ 。将 n_1 的值代入 $X(k)$ 的表达式计算, 可得:

$$X(k_1 + 2k_2 + 4k_3) =$$

$$\sum_{n_3=0}^{\frac{N}{4}-1} \sum_{n_2=0}^1 \left[B_{\frac{N}{2}} \left(\frac{N}{4} n_2 + n_3, k_1 \right) W_N^{\left(\frac{N}{4} n_2 + n_3 \right) k_1} \right] W_N^{\left(\frac{N}{4} n_2 + n_3 \right) (2k_2 + 4k_3)} \quad (7)$$

由此可得第 1 个蝶形 (butterfly, BF) 运算 BF I :

$$B_{\frac{N}{2}}\left(\frac{N}{4}n_2 + n_3, k_1\right) = x\left(\frac{N}{4}n_2 + n_3\right) + (-1)^{k_1}x\left(\frac{N}{4}n_2 + n_3 + \frac{N}{2}\right) \quad (8)$$

同样,将 n_2 的值代入计算并分解旋转因子,得到长度为 $N/4$ 的 FFT:

$$X(k_1 + 2k_2 + 4k_3) = \sum_{n_3=0}^{\frac{N}{4}-1} \left\{ H_{\frac{N}{4}}(n_3, k_1, k_2) W_N^{n_3(k_1+2k_2)} \right\} W_{\frac{N}{4}}^{n_3k_3} \quad (9)$$

其中,第 2 个蝶形运算 BF II 为:

$$H_{\frac{N}{4}}(n_3, k_1, k_2) =$$

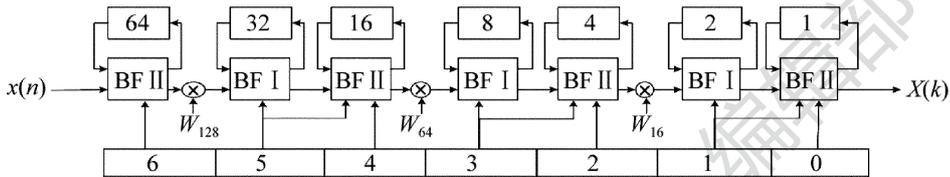


Fig. 2 R²SDF structure of 128-points FFT

128 点 FFT 可以先经过一个 R²SDF 的蝶形处理,分解为两个 64 点数据进行 R²SDF 处理, R²SDF 和 R²SDF 采用同样的 BF 单元,通过逐级递减的移位寄存器控制进行运算的数据对间隔。数据在蝶形运算核间的传递只有一条数据通路,另一条路径由移位寄存器的输出反馈到蝶形运算的输入端,因此称之为单路径延迟反馈 (single-path delay feedback, SDF)。同级 BF 之间直接相连,不同级 BF 之间需要复数乘法器乘上旋转因子,只需要一半的复数乘法器就可以完成全部的乘法运算。

3 提高高速蝶形运算核的资源利用率

在传统的 FFT 设计中,为了使输入输出分时复用,一个移位寄存器减少电路资源的使用,蝶形运算总是由组合逻辑组成,在一个时钟周期内完成。在通信系统中,对数据处理速度的要求希望处理器能够在更高的时钟频率下运行。而蝶形运算中的组合逻辑电路延时无法满足电路的时序约束条件,因此在蝶形运算中插入缓冲寄存器,打断过长的组合逻辑电路,将运算分为 3 个周期以流水线的方式进行,使其能够适用于 400MHz 的时钟频率。输出相对于输入有 3 个周期的延时,在移位寄存器的使用时间上有部分重叠。

为实现连续运算,可以不再采用单路反馈的结构,使用两组移位寄存器分别实现输出和输入流水线操作,但是这样会造成电路规模和面积的增加。为减少不必要的资源占用,可以对无重叠的部分移位寄存器进行循环使用,对蝶形运算核进行改进,将输入输出控

$$\underbrace{B_{\frac{N}{2}}(n_3, k_1)}_{\text{BF I}} + (-j)^{(k_1+2k_2)} \underbrace{B_{\frac{N}{2}}\left(n_3 + \frac{N}{4}, k_1\right)}_{\text{BF I}} \quad (10)$$

(8) 式和 (10) 式表示的两个蝶形运算中只包含了简单的固定数值的乘法,这部分乘法可以直接通过取反完成,不需要使用复数乘法器。由 (9) 式可知,在两个蝶形运算 BF I 和 BF II 之后要使用复数乘法器与分解后的旋转因子进行乘法运算。因此,在硬件实现中,每隔两个蝶形运算核就需要一个复数乘法器,这样的空间规律性能够提高复数乘法器的资源利用率,在使用流水线结构时具有很大优势。128 点 FFT 的 R²SDF 结构如图 2 所示。

制信号分离,只需增加部分移位寄存器进行缓冲,就可以达到数据连续输入输出的目的,实现高速运算核的 SDF 结构,其结构如图 3 所示。下面以第 1 级的蝶形运算设计来进行详细介绍。

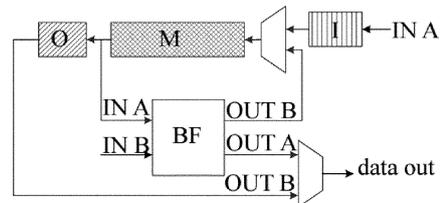


Fig. 3 Diagram of high-speed butterfly calculation

第 1 级的蝶形运算数据对间隔为 64,输入数据 IN A 和输出结果 OUT B 都需要经过长度为 64 的移位寄存器。因为输出相对输入数据有 3 个周期的延迟,设计中使用了长度为 61 的移位寄存器 M 进行输入输出的交替使用,以及两个长度为 3 的移位寄存器 O 和 I 来分别缓存重叠部分的数据。

载入时移位寄存器的使用如图 4a 所示,竖线表示前半段 64 个数据。数据先经过 I 缓存 3 个周期,再进

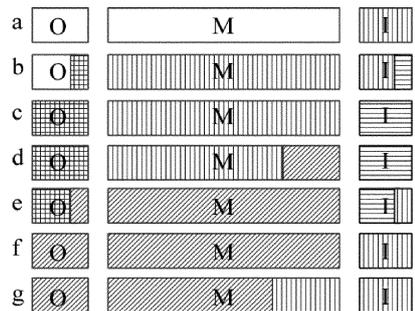


Fig. 4 Diagram of data transfer of shift register

入 M 进行移位。当第 64 个数据通过 IN B 进入运算核即后半段的数据到达时, M 开始输出前半段数据到 IN A, 对应数据对在 BF 中进行计算, 如图 4b 所示, 移位寄存器 O 中的方格表示正在进行计算的数据。

开始计算后经过 3 个周期的延迟输出计算结果 OUT A 和 OUT B。OUT A 直接输出, OUT B 直接进入 M 进行移位, 右斜线表示 OUT B 的数据, 如图 4c 和图 4d 所示, I 中缓存的是后半段输入数据用横线表示, IN A 不需要这部分数据, 此时 I 处于闲置状态。OUT B 经过 M 进行长度为 61 的移位后再进入 O 进行缓存, 如图 4e 所示, 同时 I 开始进行第 2 组数据前半段的缓存。

图 4f 和图 4g 中, OUT B 经过 M 和 O 总共 64 个周期的缓存后开始输出, 同时第 2 组前半段数据通过 I 输出端进入 M 中进行移位, 在第 1 组数据计算结果输出的同时进行第 2 组数据的缓存移位操作。这样将无使用冲突的移位寄存器合并, 只需增加少量的移位寄存器就可以实现高速蝶形运算核的连续运算, 提高了移位寄存器的利用率, 减少了重复的资源占用。

4 实验验证与性能分析

芯片选用 ALTERA 公司的 STRATIX III 系列, 在 QUARTUS II 软件上按照上述设计思路, 使用 Verilog HDL 语言完成了运算器的编写和调试。

4.1 实验结果验证

用正弦信号作为输入验证 FFT 运算器。对同一组测试数据分别用 MATLAB 和 FPGA 进行 FFT 运算, 并将两者的结果进行对比。图 5 中分别为 MATLAB 与 FPGA 的 FFT 幅频曲线与逆快速傅里叶变换 (inverse fast Fourier transform, IFFT) 恢复的采样信号, 为方便观察, 对幅频曲线的横坐标进行了对数处理。实验表明, FPGA 硬件计算结果与 MATLAB 计算出的理论值吻合, 运算器正确性得到验证。

4.2 性能分析

采用 2 维 R²SDF 结构设计的 16k 点 FFT 运算器, 资源消耗情况如表 1 所示, 并与参考文献 [16] 中采用的部分并行结构实现相同点数的 FFT 运算情况进行了对比。

Table 1 Resource consumption of different FFT structures

FFT operator	logic utilization		multipliers	equivalent gate number	block memory bits
	combination results	logic registers			
partial parallel ^[16]	8440	2464	282	253518	2269184
2-D R ² SDF	1239	510	24	31452	2074176

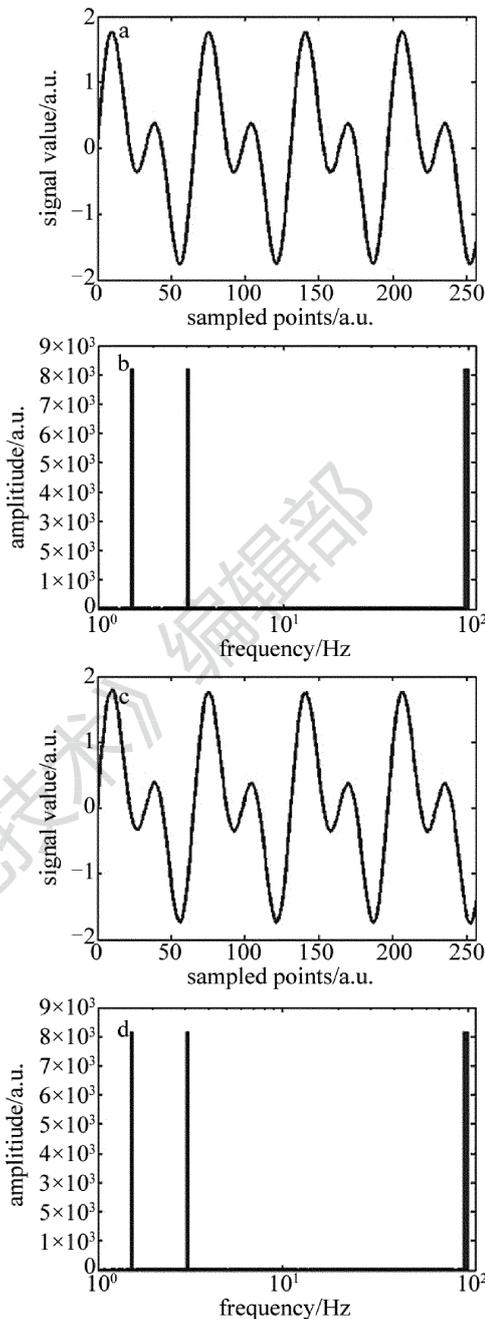


Fig. 5 Comparison between FPGA results and MATLAB values a—sampled signal b—FFT results of MATLAB c—IFFT recovery signal of FPGA d—FFT results of FPGA

2 维 R²SDF 结构与参考文献 [16] 中的部分并行结构相比, 乘法器数量和逻辑资源的使用都明显减少。由于 FFT 运算器中存储资源的使用主要由点数决定, 因此存储资源的使用量相近。而运算部分的不同电路资源可等效为门数进行比较, FPGA 中一个逻辑单元的等效门数取经典值 12, 而一个乘法器的实现至少需要 435 个门。由此可看出本设计在资源消耗方面的改善。

采用文中设计的高速蝶形运算核, 可以使运算器适应更高的工作频率, 以此提高运算速度。表 2 中列

Table 2 FFT calculator resources consumption under different clocks

clock/MHz	computing time/ μ s	equivalent gate number
50	660.740	31452
200	165.335	38136
400	82.825	70956

出了在不同工作时钟下,高速蝶形运算核的使用带来的运算速度和资源占用情况变化。

使用的资源随着工作时钟的增大而有所增加,在 400MHz 时钟下,16k 点的运算时间提高了近 8 倍,而资源占用情况增加不到 3 倍。在 FFT 运算中,每个数据都要经过列变换和行变换两个过程,一个数据的平均处理时间为两个周期,能够实现 200MHz 传输速度的频域均衡实时处理,即可以工作在较高速的信号处理中。而面积是制约 FMF 通信系统中频域均衡电路实现的关键因素,因此,采用文中设计的 FFT 能够适应 FMF 通信系统优化频域均衡电路的要求。

5 结 论

在对大点数 FFT 的 2 维分解算法及其硬件复杂度变化趋势的研究基础上,设计了基于 FPGA 的 FFT 运算器的总体结构,复用 128 点 R2²SDF 结构的运算核完成 16k 点的 FFT 运算。设计了能够适用于 400MHz 工作时钟的高速蝶形运算核,提高了存储器的资源利用率,减少了复数乘法器的使用。设计的 FFT 运算器能够适应少模光纤通信频域均衡系统中优化电路结构的要求,实现 200MHz 通信系统中数据的实时处理。可以使用多个 FFT 运算器同时处理以进一步提高频域均衡的数据处理速度。而 FPGA 的工作时钟频率仍然具有提高空间,为了满足少模光纤通信系统的符号率,在保持电路面积的基础上对运算速度的提高仍然需要进一步的研究。

参 考 文 献

[1] GRÜNER-NIELSEN L, SUN Y I, NICHOLSON J W, *et al.* Few mode transmission fiber with low DGD, low mode coupling, and low loss [J]. *Journal of Lightwave Technology*, 2012, 30(23): 3693-3698.

[2] FERREIRA F, FONSECA D, LOBATO A, *et al.* Reach improve-

ment of mode division multiplexed systems using fiber splices [J]. *IEEE Photonics Technology Letters*, 2013, 25(12): 1091-1094.

[3] CAO X. Optimization of dispersion compensation in optical fiber communication systems [J]. *Laser Technology*, 2014, 38(1): 101-104 (in Chinese).

[4] RYF R, RANDEL S, GNAUCK A H, *et al.* Mode-division multiplexing over 96km of few-mode fiber using coherent 6 × 6 MIMO processing [J]. *Journal of Lightwave Technology*, 2012, 30(4): 521-531.

[5] FARUK M S, KIKUCHI K. Adaptive frequency-domain equalization in digital coherent optical receivers [J]. *Optics Express*, 2011, 19(13): 12789-12798.

[6] BAI N. Mode-division multiplexed transmission in few-mode fibers [D]. Orlando, Florida, USA: University of Central Florida Orlando, 2013: 9-20.

[7] BAI N, IP E, LI M J, *et al.* Long-distance mode-division multiplexed transmission using normalized adaptive frequency domain equalization [C]//*IEEE Photonics Society Summer Topical Meeting Series*. New York, USA: IEEE, 2013: 135-136.

[8] BAI N, LI G F. Adaptive frequency-domain equalization for mode-division multiplexed transmission [J]. *IEEE Photonics Technology Letters*, 2012, 24(21): 1918-1921.

[9] BAI N, IP E, HUANG Y K, *et al.* Mode-division multiplexed transmission with inline few-mode fiber amplifier [J]. *Optics Express*, 2012, 20(3): 2668-2680.

[10] YAMAN F, BAI N, ZHU B Y, *et al.* Long distance transmission in few-mode fibers [J]. *Optics Express*, 2010, 18(12): 13250-13257.

[11] HE H, KE X Zh, WANG W. Particle filtering based semi-blind estimation for atmospheric laser OFDM time-varying channel [J]. *Laser Technology*, 2011, 35(6): 738-741 (in Chinese).

[12] LI S M, CHEN Y, ZENG X Y. FFT/IFFT processor for MIMO-OFDM [J]. *Computer Engineering*, 2012, 38(2): 248-249 (in Chinese).

[13] JIANG X, TANG G, WU B. Design of configurable high precision FFT/IFFT Processor [J]. *Micro Electronics and Computer*, 2010(7): 44-48 (in Chinese).

[14] WANG X Y. Implementation of wavelet denoising algorithm based on FPGA [J]. *Laser Technology*, 2013, 37(6): 786-790 (in Chinese).

[15] ZHANG L J. Research on FPGA parallel implementation of two-dimension long FFT algorithm [J]. *Radio Communications Technology*, 2013, 39(3): 86-88 (in Chinese).

[16] LI W, SUN J P, WANG J, *et al.* Implementation of 32k points ultrahigh speed FFT processor based on FPGA devices [J]. *Journal of Beijing University of Aeronautics and Astronautics*, 2008, 33(12): 1440-1443 (in Chinese).