

# 基于低硬件复杂度、高速 CORDIC 的 SVD 模块设计与实现

张晓帆, 李广军

(电子科技大学通信学院, 四川成都 611731)

**摘要:** 为降低实现高阶矩阵 SVD 时的硬件复杂度和计算延时, 本文改进了 CORDIC 迭代结构, 设计了一种用于 SVD 的低硬件复杂度、高速 CORDIC 计算单元. 本文以  $2 \times 2$  矩阵为例, 基于 Xilinx Virtex6 硬件平台设计并实现了使用优化后 CORDIC 计算单元的 SVD 模块, 在 19bit 位宽下吞吐率达 25.9Gbps. 对比 Xilinx IP core 中同类模块, 本文设计节省 27.6% 寄存器, 27.7% 查找表, 实时性提高 14%. 对高阶矩阵, 本文给出资源消耗趋势曲线, 可证明优化后 CORDIC 计算单元能降低 16 阶矩阵 SVD 模块约 40% 的硬件复杂度.

**关键词:** 奇异值分解 (SVD); 坐标旋转数字计算机 (CORDIC); 向量旋转

**中图分类号:** TN713.7 TN431.2 **文献标识码:** A **文章编号:** 0372-2112 (2015)04-0738-05

**电子学报 URL:** <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2015.04.016

## The Design and Implementation of SVD Module with Reduced Hardware Complexity and High-Speed CORDIC Processor

ZHANG Xiao-fan, LI Guang-jun

(Centre for Communication Circuits and Systems, UESTC, Chengdu, Sichuan 611731, China)

**Abstract:** In order to reduce the hardware complexity and the delay of high-order SVD processor, two improved CORDIC modules including Arc Tan and Rotation functions are designed. These two improved CORDIC modules have better performance in terms of register saving and real-time quality. In this paper, a  $2 \times 2$  SVD module using above-mentioned CORDIC modules with 19bit data width has implemented on Xilinx Virtex6 and the throughput reaches 25.9Gbps. Compared with the  $2 \times 2$  SVD module using IP core, it reduced 27.6% registers, 27.7% LUTs and improved 14% real-time performance. Moreover, the trend curves of hardware consumption are presented which have testified that these two improved CORDIC modules can reduce 40% hardware complexity of 16-order SVD processor.

**Key words:** singular value decomposition (SVD); coordinate rotation digital computer (CORDIC); vector rotation

## 1 引言

SVD (Singular Value Decomposition) 常见于信号处理、信号检测等领域. 自 1969 年 Golub 和 Kahan 提出传统 QR 迭代算法后<sup>[1]</sup>, 零位移 QR 算法<sup>[2]</sup>进一步提高了 SVD 计算精度. 而 Forsythe 提出的 Jacobi 算法<sup>[3]</sup>提高了算法并行度, 令硬件实现更方便. 自 CORDIC (Coordinate Rotation Digital Computer) 出现以来, 又吸引了众多学者发展低硬件复杂度、高速的 SVD 算法<sup>[4-6]</sup>. 文献[7]提出了一种对  $2 \times 2$  子矩阵作对角化, 实现高阶矩阵 SVD 的方法, 我们因此可以灵活解决不同阶数矩阵的 SVD 计算. 针对大规模矩阵, 文献[8]提出的位串行结构 SVD 处理器节约了资源, 但实时性差. 文献[9]改进了流水线结构

SVD 处理器中数据流向, 减少计算延时, 却付出额外的硬件开销. 而文献[10]对 Jacobi 算法在 FPGA 上实现浮点数运算作研究, 令 SVD 运算拥有更高精度和更大动态范围, 但硬件复杂度高. 因此, 本文对采用双边 Jacobi 算法的 SVD 模块作研究, 优化其中的 CORDIC 计算单元, 降低其硬件复杂度和计算延时, 为高阶矩阵 SVD 运算节省硬件开销和计算时间.

## 2 双边 Jacobi SVD 算法

对于  $2 \times 2$  矩阵  $G$ , 双边 Jacobi SVD 算法计算步骤如式(1)~(4). 其中  $\theta_L, \theta_R$  为左右旋转角, 而式(2)等号的右边是矩阵  $G$  的对角矩阵, 其对角线上元素即为矩阵  $G$  的奇异值.

设  $G = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ , 其中, 左右旋转角与  $G$  的元素有以下关系:

$$\begin{cases} \theta_R + \theta_L = \arctan\left(\frac{c+b}{d-a}\right) \\ \theta_R - \theta_L = \arctan\left(\frac{c-b}{d+a}\right) \end{cases} \quad (1)$$

$$\begin{pmatrix} \cos\theta_L & -\sin\theta_L \\ \sin\theta_L & \cos\theta_L \end{pmatrix} \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} \cos\theta_R & \sin\theta_R \\ -\sin\theta_R & \cos\theta_R \end{pmatrix} \quad (2)$$

$$= \frac{1}{2} \begin{pmatrix} A+C & B+D \\ B-D & C-A \end{pmatrix}$$

$$\begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} \cos\alpha - \sin\alpha \\ \sin\alpha \cos\alpha \end{pmatrix} \begin{pmatrix} a-d \\ c+b \end{pmatrix} \quad (3)$$

$$\begin{pmatrix} C \\ D \end{pmatrix} = \begin{pmatrix} \cos\beta - \sin\beta \\ \sin\beta \cos\beta \end{pmatrix} \begin{pmatrix} a+d \\ b-c \end{pmatrix} \quad (4)$$

并且  $\alpha = \theta_R + \theta_L, \beta = \theta_R - \theta_L$ . 综上,  $2 \times 2$  矩阵 SVD 运算可方便地利用 CORDIC 实现反正切函数, 如式(1)和向量旋转运算, 如式(3)、(4)实现<sup>[11]</sup>.

### 3 CORDIC 优化算法设计与实现

由 CORDIC 迭代式可知, 每次迭代都需要计算  $x, y, z$  三路数据<sup>[12]</sup>, 故其最基本运算单元, 如图 1.

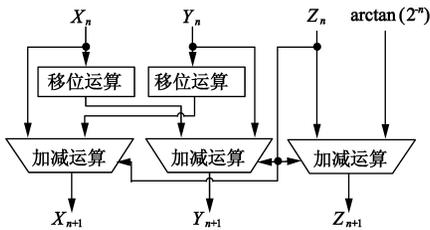


图1 CORDIC基本运算单元

对有高精度高吞吐率要求的应用场合, CORDIC 必须使用流水线结构, 基本运算单元数随迭代次数增加而上升. 为减少 CORDIC 的硬件复杂度, 提高模块实时性, 本文将分别对 CORDIC 向量模式和旋转模式作优化.

#### 3.1 CORDIC 向量模式优化算法

以 19 比特为例, 我们在 Matlab 上进行 CORDIC 向量模式仿真, 如图 2. 我们发现幅度值收敛速度比反正切值快.  $x$  路表示的幅度值经过 8 次迭代已经完全收敛, 而  $z$  路表示的反正切值约需 16 次迭代. 根据此现象, 本文对不同数据通路采用不同的迭代次数, CORDIC 迭代关系由文献<sup>[12]</sup>中的迭代式改为式(5).

$$\begin{cases} x_{m+1} = x_m \\ y_{m+1} = (y_m + \delta_m x_m 2^{-m}) \\ z_{m+1} = z_m - \delta_m \arctan 2^{-m} \end{cases} \quad (5)$$

硬件实现上, 从图 3 可见, 在  $n+1$  次迭代后,  $x$  数

据收敛, 可在第  $n+2$  次及以后的迭代关闭  $x$  数据通路, 故 CORDIC 基本计算单元可节省一个移位运算单元和一个加减法运算单元, 降低了硬件实现时复杂度.

#### 3.2 FPGA 实现 CORDIC 求反正切模块

我们对 CORDIC 求反正切模块的不同位宽作 Matlab 仿真, 并在 FPGA 上实现. 由图 2、4 和 5 可知, CORDIC 求反正切模式在位宽为 19、27 和 35 比特时, 幅度值分别在第 8、第 14 和第 17 次迭代后收敛. 此后的迭代, 将关闭  $x$  数据通路以降低硬件复杂度.

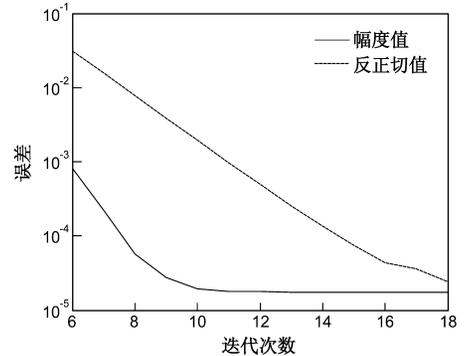


图2 CORDIC向量模式输出误差与迭代次数关系

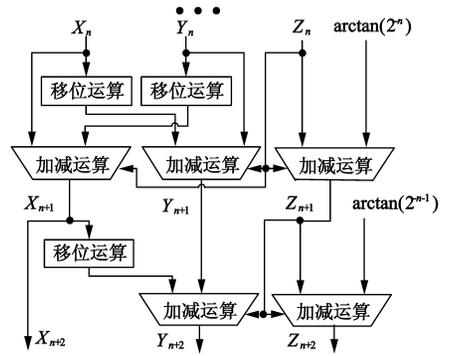


图3 优化后CORDIC基本单元结构

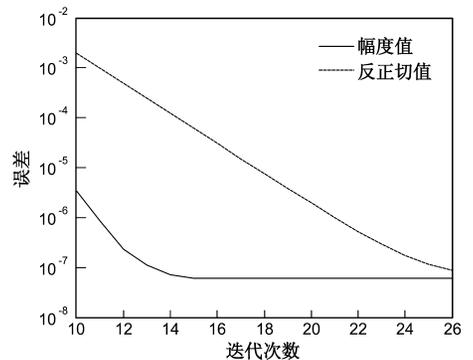


图4 27比特位宽时模块输出误差与迭代次数关系

如表 1、2 所示, 优化后的 CORDIC 求反正切模块比 Xilinx IP core 中同类型模块更节省硬件资源: 位宽分别为 19、27 和 35 比特时, 寄存器开销可分别减少 12.5%、

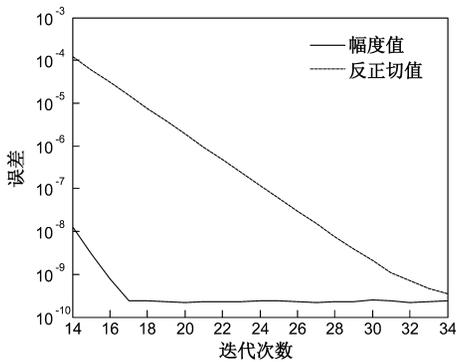


图5 35比特位宽时模块输出误差与迭代次数关系

11.8%和14.9%;查找表开销可减少11.7%、16.4%和21.4%;实时性提高5.0%、6.7%和5.2%。

表1 CORDIC求反正切模块资源消耗情况

模块名	寄存器	查找表	查找表-触发器组合
输入输出数据位宽为19比特			
本文设计模块	1114	1160	1209
IP core	1274	1315	1414
输入输出数据位宽为27比特			
本文设计模块	2190	2127	2207
IP core	2484	2545	2659
输入输出数据位宽为35比特			
本文设计模块	3549	3310	3514
IP core	4171	4215	4425

表2 CORDIC求反正切模块实时性情况

模块名	输出滞后输入时钟个数		
	19比特	27比特	35比特
本文设计模块	20	28	36
IP core	21	30	38

### 3.3 CORDIC旋转模式优化算法

本文选择优化CORDIC的迭代次数,结合Low latency算法<sup>[13]</sup>,设计出一种能有效提高模块实时性、节省大量资源的优化方案.假设CORDIC在第 $j$ 次迭代后,关系如式(6).

$$\begin{cases} x_1 = x_j \cos z_j - y_j \sin z_j \\ y_1 = y_j \cos z_j + x_j \sin z_j \end{cases} \quad (6)$$

其中, $z$ 为剩余待旋转角度值.在特定精度下

$$\cos(\theta_j) \approx 1, \sin(\theta_j) \approx \theta_j \quad (7)$$

随后的CORDIC迭代关系则由式(6)变为:

$$\begin{cases} x_1 = x_j - y_j \cdot z_j \\ y_1 = y_j + x_j \cdot z_j \end{cases} \quad (8)$$

因为式(7)在CORDIC迭代进行至一半时就能符合误差要求,如图7,所以在硬件实现上,可通过一次乘法

操作,减少近一半迭代次数,节省大量如图1所示的CORDIC基本运算单元,降低了模块硬件复杂度并提高了模块实时性.软件仿真结果如图6,在相同的输出精度要求下,使用本文算法的收敛速度可比传统算法快将近一倍.

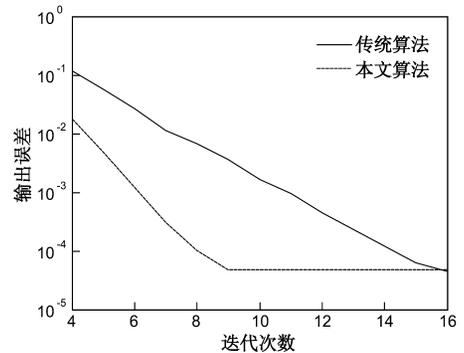


图6 CORDIC旋转模式x分量输出与迭代次数关系

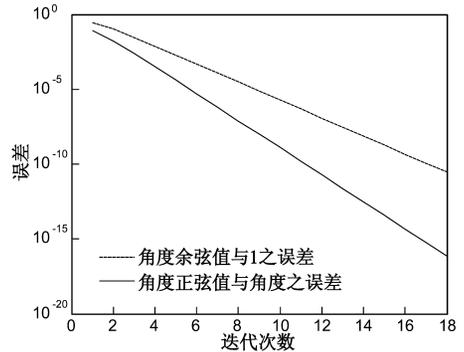


图7 不同迭代次数下近似计算误差

### 3.4 FPGA实现CORDIC向量旋转模块

考虑模块在不同位宽下的截位误差,由图7和近似关系式(8)可知,位宽为19、27和35比特时,可分别第9次、第13次和第17次传统迭代后,再使用1次乘法操作,直接得到迭代输出结果.

表3 CORDIC向量旋转模块资源消耗情况

模块名	寄存器	查找表	查找表-触发器组合	乘法器
输入输出数据位宽为19比特				
本文设计模块	1102	1096	1189	4
IP core	1870	1924	2004	0
输入输出数据位宽为27比特				
本文设计模块	2529	2506	2622	8
IP core	3553	3534	3597	0
输入输出数据位宽为35比特				
本文设计模块	3877	3715	3926	10
IP core	5760	5737	5942	0

表 4 CORDIC 向量旋转模块实时性情况

模块名	输出滞后时钟个数		
	19 比特	27 比特	35 比特
本文设计模块	20	28	33
IP core	25	34	42

表 3、4 可见,本文设计的模块拥有更少的硬件资源开销和更高的实时性. 在使用少量乘法器的情况下,位宽分别为 19、27 和 35 比特时,本文设计模块的寄存器开销可分别减少 41.0%、28.8% 和 32.6%;查找表开销可减少 43.0%、29.0% 和 35.2%;实时性提高 25.0%、17.6% 和 21.4%.

## 4 基于优化 CORDIC 计算单元的 SVD 模块设计与实现

### 4.1 SVD 模块设计

本文设计的 SVD 模块采用流水线结构,以  $2 \times 2$  SVD 为例,其硬件结构如图 8. 由 Jacobi 算法可知,  $n$  阶方阵

的 SVD 对角化模块,见图 9,完整的 SVD 迭代过程还需要进行  $\log n$  次清扫<sup>[14]</sup>,每次清扫需要  $n(n-1)$  次对角化,需要用到大量 CORDIC 计算单元.

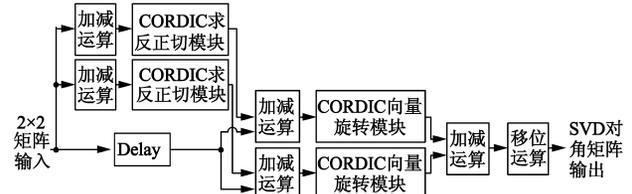


图 8  $2 \times 2$  SVD 模块结构图

### 4.2 SVD 模块 RTL 仿真

本文以 19 比特位宽的  $2 \times 2$  SVD 为例,在 Xilinx Virtex6 硬件平台实现模块. 输出首次处理延时 43 个时钟周期,在随后的每个时钟上升沿均能输出一个  $2 \times 2$  矩阵的 SVD 对角矩阵,模块最高工作频率为 348.918MHz,吞吐率达 25.9Gbps.

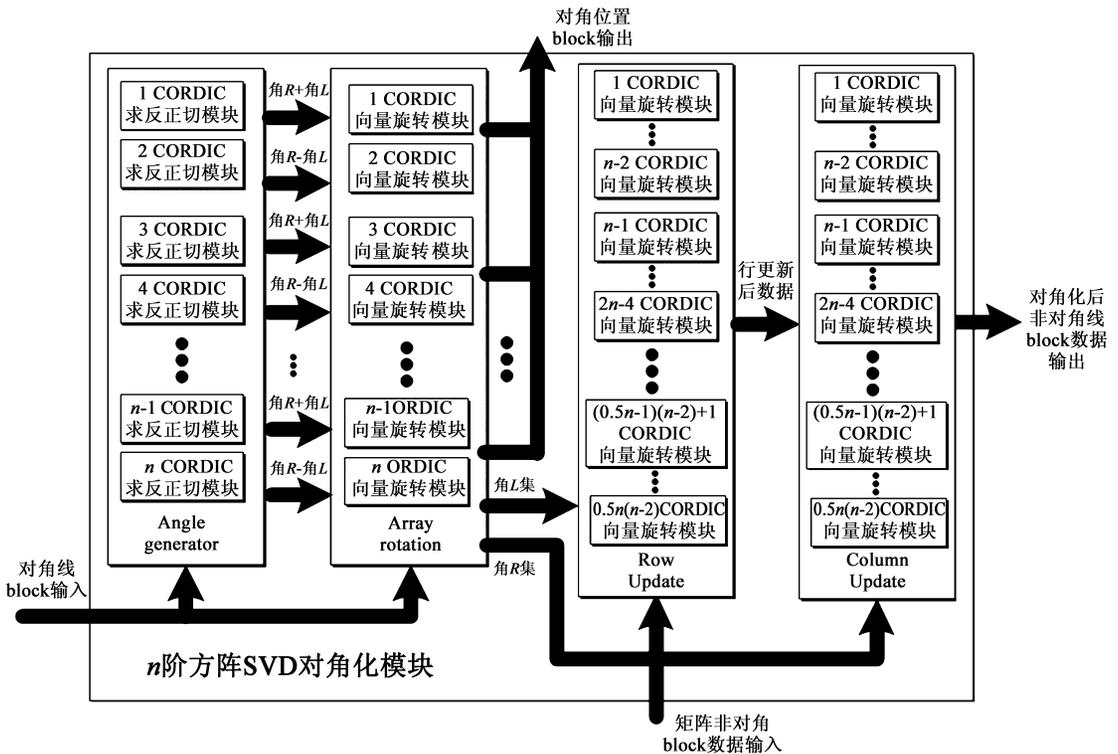


图 9  $n$  阶 SVD 对角化模块结构图

表 5 SVD 模块资源消耗和实时性情况

模块名	寄存器	查找表	查找表-触发器组合	乘法器	输出滞后时钟
使用优化后 CORDIC 计算单元的 SVD 模块	4802	4890	5065	8	43
使用后 IP core 中 CORDIC 计算单元的 SVD 模块	6630	6765	7092	0	50

表 5 可知,在同一硬件平台下,本文设计的 SVD 模块比使用 Xilinx IP core 构造的同类模块节省了 27.6% 寄存器、27.7% 查找表,实时性提高 14.0%.

### 4.3 高阶 SVD 模块资源消耗分析

图 11 和 12 列举了 19 比特位宽的不同阶数 SVD 模块需要使用的硬件资源. 可以看到,在硬件实现 16 阶矩阵 SVD 时,可节省 39.8% 寄存器和 41.7% 查找表资源.

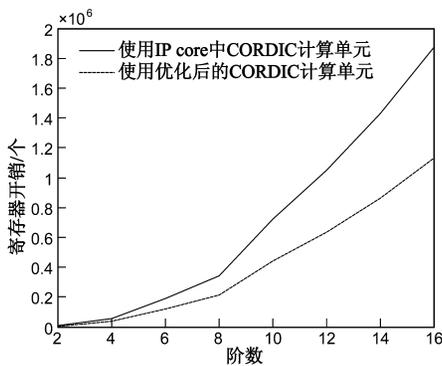


图11 不同阶数下SVD模块寄存器开销趋势

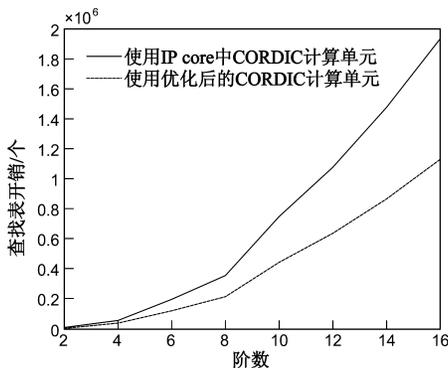


图12 不同阶数下SVD模块查找表开销趋势

## 5 结束语

本文对CORDIC计算单元作优化,改进其迭代结构,降低其硬件复杂度,提高其计算实时性,为高阶矩阵SVD提供一种节省资源和减少延时的方案.本文使用上述CORDIC计算单元设计了位宽为19比特的 $2 \times 2$ SVD模块.在Xilinx Virtex6硬件平台上实现时,吞吐率达25.9Gbps,对比使用Xilinx IP core中同类模块,节省了27.6%寄存器,27.7%查找表,实时性提高14%.最后,本文给出高阶矩阵情况下SVD资源消耗趋势曲线,证明本文设计的CORDIC计算单元能降低16阶矩阵SVD模块约40%的硬件复杂度.

## 参考文献

- [1] Businger P A, Golub G H. Algorithm 358: singular value decomposition of a complex matrix [F1, 4, 5] [J]. Communications of the ACM, 1969, 12(10): 564 - 565.
- [2] Demmel J, Kahan W. Accurate singular values of bidiagonal matrices [J]. SIAM Journal on Scientific and Statistical Computing, 1990, 11(5): 873 - 912.
- [3] Forsythe G E, Henrici P. The cyclic Jacobi method for computing the principal values of a complex matrix [J]. Transactions of the American Mathematical Society, 1960, 94(1): 1 - 23.
- [4] Cavallaro J R, et al. VLSI implementation of a CORDIC SVD

processor [A]. University Government Industry Microelectronics Symposium [C]. IEEE, 1989. 256 - 260.

- [5] Ma W, et al. An FPGA-based singular value decomposition processor [A]. Canadian Conference on Electrical and Computer Engineering [C]. Ottawa: IEEE, 2006. 1047 - 1050.
- [6] Szczówka P M, Malinowski P. CORDIC and SVD implementation in digital hardware [A]. Mixed Design of Integrated Circuits and Systems [C]. Wroclaw: IEEE, 2010. 237 - 242.
- [7] Brent R P, et al. The solution of singular-value and symmetric eigenvalue problems on multiprocessor arrays [J]. SIAM Journal on Scientific and Statistical Computing, 1985, 6(1): 69 - 84.
- [8] 谭曼琼, 等. 位串行SVD处理器的设计 [J]. 小型微型计算机系统, 2012, 33(006): 1358 - 1362.  
Tan Man-qiong, et al. Design of bit-serial singular value decomposition processor [J]. 2012, 33(006): 1358 - 1362. (in Chinese)
- [9] Huang Kuan-Ju. A pipeline VLSI design of fast singular value decomposition processor for real-time EEG system based on on-line recursive independent component analysis [A]. Engineering in Medicine and Biology Society [C]. Osaka: IEEE, 2013. 1944 - 1947.
- [10] 陈刚, 等. 基于CORDIC算法的高精度浮点对称矩阵特征值分解的FPGA实现 [J]. 计算机科学, 2013, 40(5): 35 - 37.  
Chen Gang, et al. Floating-point CORDIC-based real-valued symmetric matrix eigenvalue decomposition on FPGA [J]. Computer Science, 2013, 40(5): 35 - 37. (in Chinese)
- [11] 毕卓, 戴益君. 全定制CORDIC运算器设计 [J]. 计算机工程与科学, 2011, 33(10): 64 - 69.  
Bi Zhuo, Dai Yi-jun. Full custom CORDIC arithmetic unit design [J]. Computer Engineering and Science, 2011, 33(10): 64 - 69. (in Chinese)
- [12] J E Volder. The CORDIC trigonometric computing technique [J]. IRE Transactions on Electronic Computers, 1959, 8(33): 330 - 334.
- [13] E Antelo, J Villalba, E L Zapata. A low-latency pipelined 2D and 3D CORDIC processors [J]. IEEE Transactions on Computers, 2008, 57(3): 404 - 417.
- [14] Ahmedsaid A, Amira A, Bouridane A. Improved SVD systolic array and implementation on FPGA [A]. Field-Programmable Technology [C]. Tokyo: IEEE, 2003. 35 - 42.

## 作者简介

张晓帆 男, 1990年5月出生, 广东广州人. 2013年毕业于电子科技大学通信与信息工程学院, 现为电子科技大学通信与信息工程学院硕士研究生, 从事VLSI数字信号处理实现技术方面的研究。

E-mail: zhangxd218@qq.com

李广军 男, 1950年9月出生, 河北保定人. 现为电子科技大学通信与信息工程学院教授、博士生导师、中国通信学会通信专用集成电路委员会委员、四川省通信学会副理事长. 主要从事VLSI数字信号处理实现技术、移动通信系统和嵌入式系统方面的研究工作。

E-mail: gjli@uestc.edu.cn