

# 自适应定阶的快速 Burg 算法设计与 FPGA 实现 \*

郭鸣晗<sup>1</sup>, 陈立平<sup>2</sup>, 张 浩<sup>2</sup>, 赵 坤<sup>2</sup>, 柏 伟<sup>1</sup>

(1. 中国科学院大学, 北京 100049; 2. 中国科学院微电子研究所, 北京 100029)

**摘要:** 针对信号频谱分析的实时性要求, 设计了一种适用于短序列的自适应定阶的快速 Burg 算法硬件加速电路。以 FPGA 为平台进行实验, 将快速 Burg 算法与最终预测误差(Final Prediction Error, FPE)准则结合可做到自回归(Auto-Regressive, AR)参数自适应定阶。实现了灵活控制的并行二级流水线结构和并行化计算单元, 同时优化了存储单元, 达到速度与面积的平衡。实验结果表明, 该算法对短序列也能准确地估计信号频率, 与 Burg 算法硬件实现方案的计算时间对比, 该算法将运算时间降低了 75%, 确实起到了加速作用, 并且节省了内存空间, 符合设计要求。

**关键词:** AR 参数模型; Burg 算法; 快速 Burg 算法; FPGA; 硬件加速

中图分类号: TN911.72; TN4

文献标识码: A

DOI: 10.16157/j.issn.0258-7998.211411

中文引用格式: 郭鸣晗, 陈立平, 张浩, 等. 自适应定阶的快速 Burg 算法设计与 FPGA 实现[J]. 电子技术应用, 2021, 47(11): 62-67, 72.

英文引用格式: Guo Minghan, Chen Liping, Zhang Hao, et al. Design and FPGA implementation of fast Burg algorithm of adaptive order determination[J]. Application of Electronic Technique, 2021, 47(11): 62-67, 72.

## Design and FPGA implementation of fast Burg algorithm of adaptive order determination

Guo Minghan<sup>1</sup>, Chen Liping<sup>2</sup>, Zhang Hao<sup>2</sup>, Zhao Kun<sup>2</sup>, Bai Wei<sup>1</sup>

(1. University of Chinese Academy of Sciences, Beijing 100049, China;

2. Institute of Microelectronics of Chinese Academy of Sciences, Beijing 100029, China)

**Abstract:** Aiming to real-time requirement of signal spectrum analysis, an adaptive ordering of fast Burg algorithm hardware acceleration circuit for short sequence based on FPGA is designed. The fast Burg algorithm combined with FPE criterion can be used to determine the order of AR parameters. The parallel two-stage pipeline structure with flexible control is realized, and the parallel computing unit is parallelized. At the same time, the storage unit is optimized to achieve the balance between speed and area. The test show that the algorithm can accurately estimate the signal frequency for short sequences. Compared with the calculation time of Burg algorithm hardware implementation scheme, this algorithm reduces the calculation time by 75%, which does play a role of acceleration, and saves memory space. So, this design meets the design requirements.

**Key words:** AR parameter model; Burg algorithm; fast Burg algorithm; FPGA; hardware acceleration

## 0 引言

现代功率谱估计的 AR 模型法使用有限长的数据序列来估计假设模型的参数, 再将参数带入功率谱密度模型中, 可获得较好的功率谱估计结果<sup>[1-3]</sup>。

Burg 算法是一种常见的 AR 模型求功率谱的方法, 其主导思想是利用前后向预测误差功率之和最小的方法来计算反射系数  $k$ , 然后带入 Levinson 递推, 求解 AR 模型参数<sup>[4-6]</sup>。此方法在处理短数据时具有较高的频率分辨率<sup>[7-8]</sup>, 但求解反射系数计算量较大。为了改进这一问题, Vos 提出一种快速 Burg 算法<sup>[9]</sup>, 通过一系列矩阵变换降低了反射系数求解时的计算量, 但是不能确定 AR

模型的阶数, 并且串行算法的执行耗时较长。针对上述问题, 本文将快速 Burg 算法与 FPE 准则<sup>[10]</sup>结合, 对短序列的功率谱估计实现自适应定阶的功能, 达到较高频率分辨率, 并使用 Verilog 硬件描述语言设计电路, 达到硬件加速功能<sup>[11]</sup>。电路结构在二级流水线的基础上<sup>[12]</sup>, 结合自适应定阶方案, 提出一种新的流水线结构, 并设置状态机灵活控制。本文对计算单元进行并行化处理加速计算。考虑速度与面积的折中, 针对算法特点设计内存读写方案, 减少数据存储长度, 从而减小了存储单元的面积。

## 1 自适应定阶的快速 Burg 算法

快速 Burg 算法过一系列矩阵变换将反射系数计算过程化简<sup>[9]</sup>, 避免了 Burg 算法<sup>[4]</sup>中计算反射系数前需计算出前、后向预测误差  $e_f$ 、 $e_b$  而带来的较大计算量。在本

\* 基金项目: 国家重点研发计划项目(2020YFC2003304)

算法中,通过自适应定阶的方式,对长度为  $N$  的输入序列  $x_n$  的递归计算步骤如下:

(1)计算自相关  $c_i$ 。(本实验中自适应定阶在 10 阶以内,在此只计算 10 个自相关值)。

$$c_i = [x_0 \ x_1 \ \cdots \ x_{N-i-1}] \begin{bmatrix} x_i \\ x_{i+1} \\ \vdots \\ x_{N-1} \end{bmatrix} \quad (i=0, 1, \dots, 9) \quad (1)$$

(2)初始化。初始为第一阶,此时迭代次数  $i=0$ ;用于计算反射系数的列向量  $g_0 = [2c_0 - |x_0|^2 - |x_{N-1}|^2; 2c_1]^T$ ; 用于计算列向量  $g_i$  的列向量  $r_i = c_i$ ; 预测系数  $a_0 = 1$ ; 反射系数  $k = \frac{2c_1}{2c_0 - |x_0|^2 - |x_{N-1}|^2}$ 。

(3)计算初始预测误差功率  $P_{\min}(0)$ , 预测误差 FPE 值和初始阈值 TH:

$$P_{\min}(0) = \frac{c_0}{N} \quad (2)$$

$$TH = FPE(i) = \frac{N+i+1}{N-i-1} P_{\min}(i) \quad (3)$$

(4)判断迭代条件( $TH < FPE(i)$ ),即若满足阈值 TH 小于预测误差 FPE 值,则将当前阈值 TH 更新为预测误差值  $TH = FPE(i)$ ,之后  $i+1$ ,继续执行步骤(5);若不满足,则计算阶数,输出预测系数  $a_i$ 。

(5)更新列向量  $r_{i+1}$ :

$$r_{i+1} = \begin{bmatrix} 2c_{i+1} \\ r_i - \begin{bmatrix} x_0 \\ \vdots \\ x_{i-1} \end{bmatrix} x_i - \begin{bmatrix} x_{N-1} \\ \vdots \\ x_{N-i} \end{bmatrix} x_{N-i-1} \end{bmatrix} \quad (4)$$

(6)更新用于计算列向量  $g_i$  的矩阵:

$$\Delta R_{i+1} = - \begin{bmatrix} x_i \\ \vdots \\ x_0 \end{bmatrix} \begin{bmatrix} x_i & \cdots & x_0 \end{bmatrix} \begin{bmatrix} x_{N-i-1} \\ \vdots \\ x_{N-i} \end{bmatrix} \begin{bmatrix} x_{N-i-1} & \cdots & x_{N-i} \end{bmatrix} \quad (5)$$

(7)更新列向量  $g_i$ :

$$g_i = \begin{bmatrix} g_{i-1} + k_{i-1} J g_{i-1} + \Delta R_{i+1} a_i \\ r_{i+1}^T a_i \end{bmatrix} \quad (6)$$

其中,  $J$  是  $i+1$  行的方阵:

$$J = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ \cdots & \cdots & 1 & \cdots \\ 0 & \cdots & \cdots & \cdots \\ 1 & 0 & \cdots & 0 \end{bmatrix} \quad (7)$$

(8)计算初始反射系数:

$$k = - \frac{[a_i^T \ 0]^T J g_i}{[a_i^T \ 0] g_i} \quad (8)$$

(9)计算最小预测误差功率  $P_{\min}(i)$  与最终预测误差准则中的 FPE( $i$ )值:

$$P_{\min}(i) = P_{\min}(i-1)(1 - |k|^2) \quad (9)$$

$$FPE(i) = \frac{N+i+1}{N-i-1} P_{\min}(i) \quad (10)$$

(10)更新 AR 模型参数  $a_i$ , 跳转到步骤(3)。

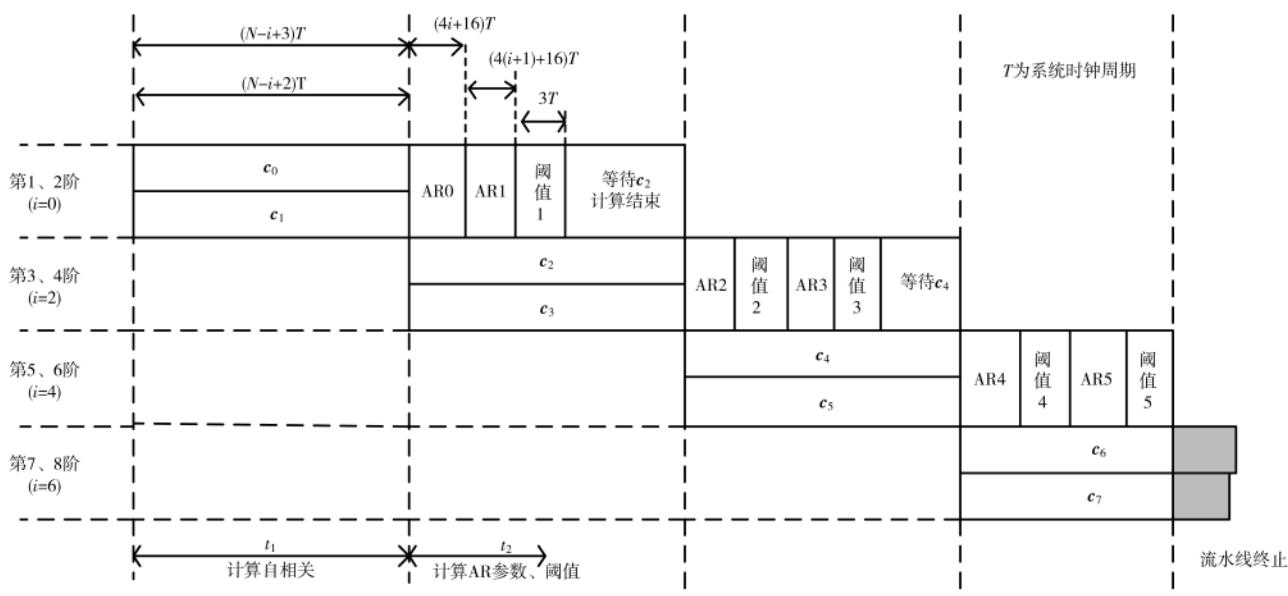
$$a_{i+1} = \begin{bmatrix} a_i \\ 0 \end{bmatrix} + k J \begin{bmatrix} a_i \\ 0 \end{bmatrix} \quad (11)$$

## 2 硬件电路设计与实现

### 2.1 并行二级流水线结构

本算法完成硬件加速的关键思路是使用流水线的结构。自相关  $c_i$  的计算只和输入序列  $x_n$  有关,与 AR 参数的计算无关,而 AR 参数的计算需要利用自相关的值,故电路结构将划分为二级流水线,第一级为自相关  $c_i$  的计算,第二级为 AR 参数和阈值的计算。为了提高计算速度,本设计将采用并行的二级流水线结构,时序关系如图 1 所示。

本电路结构的顶层电路图如图 2 所示,计算单元由两个并行连接的自相关计算单元和 AR 参数、阈值计算



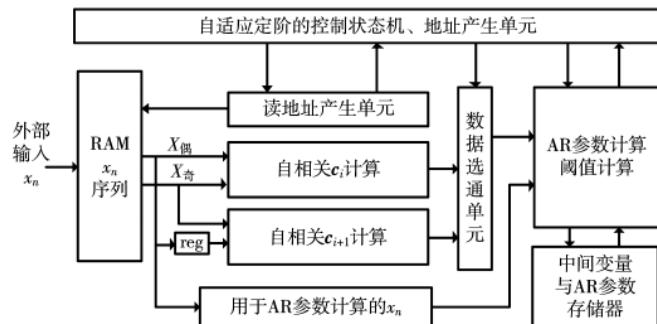


图 2 快速 Burg 算法顶层电路图

单元构成,控制单元为状态机来控制迭代次数以及计算单元。存储单元包括两个存放  $x_n$  的 RAM,宽度为 16,深度为序列长度  $N$ (在本设计中  $N=128$ ),一个用于自相关计算,一个用于 AR 参数计算;第二部分用来存储 AR 参数、阈值计算单元产生的中间结果。

## 2.2 自适应定阶与状态机控制

为了达到自适应定阶目的和提高流水线使用效率,采用状态机来控制计算单元,根据阈值 FPE 是否达到最小值来确定是否停止运算。在本算法中,AR 参数的阶数自适应得确定在 2~10 阶内。

本状态机的输入信号为外部输入的算法使能信号和计算单元的结束状态标识信号,输出为计算单元的使能控制信号。

根据并行二级流水线结构和自适应定阶算法,控制单元状态机如图 3 所示,共分为 6 个状态。

算法启动后,初始阶数为 1,迭代次数  $i=0$ ,进入 S1 状态进行初始自相关计算;之后进入 S2 启动后续自相关计算及 AR 参数和阈值计算。计算完成后若 FPE 达到最小值,跳至 S5 结束运算,最终定阶为  $i+1$  阶;否则跳转至 S4,启动下一阶 AR 参数模型和阈值计算。计算完成后若 FPE 达到最小值,则跳转至 S5 结束运算,最终定阶为  $i+2$  阶;否则跳转至 S4,等待当前的自相关计算结束后再次进入 S2 进行下一轮的迭代,直至 FPE 达到最小值,迭代停止,RAM 中的 AR 参数即为所求。

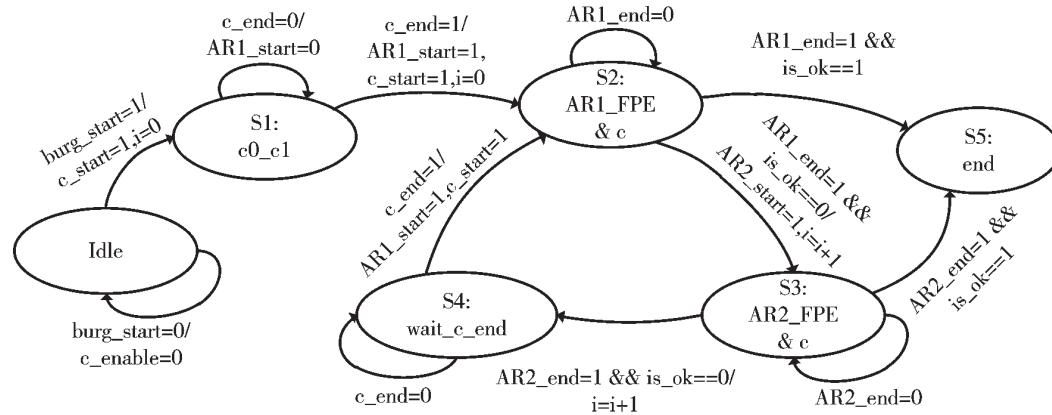


图 3 快速 Burg 算法控制单元状态机

## 2.3 计算单元的设计

### 2.3.1 自相关的计算单元

自相关的计算使用流水线结构。产生地址、读取数据、乘累加运算,三拍完成一次乘累加输出,两个自相关计算单元分别进行  $N-i$  次和  $N-i-1$  次乘累加后,输出自相关值  $c_i, c_{i+1}$  给下一级计算单元,当前阶数下消耗的总计算时长为  $N-i+3$  个时钟周期。

### 2.3.2 AR 参数和阈值计算单元

本计算单元电路包括 4 个模块:预备参数计算模块、反射系数  $k$  值和 AR 参数  $a$  计算模块、预测误差功率  $P_{\min}$  计算阈值计算模块,以及判断阈值是否达到最小值的定阶模块。本单元顶层结构如图 4 所示。

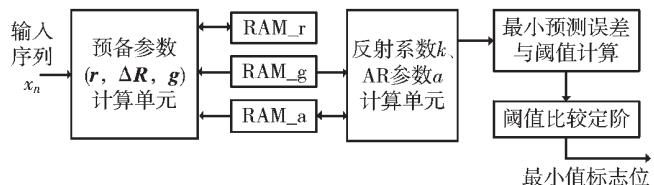
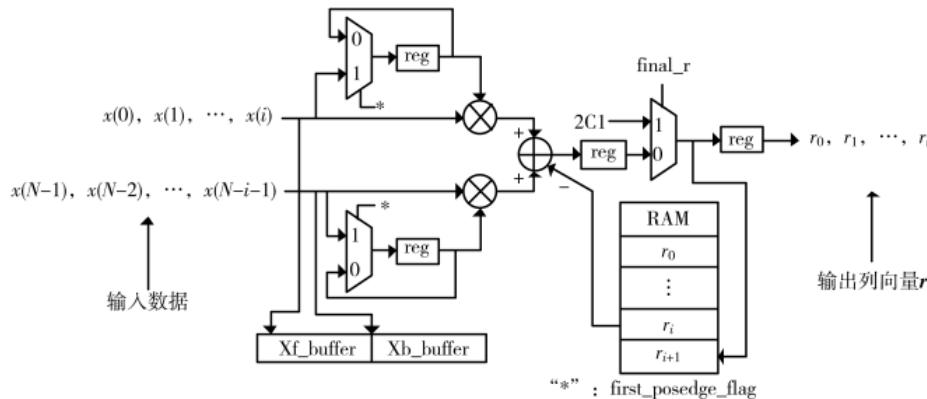
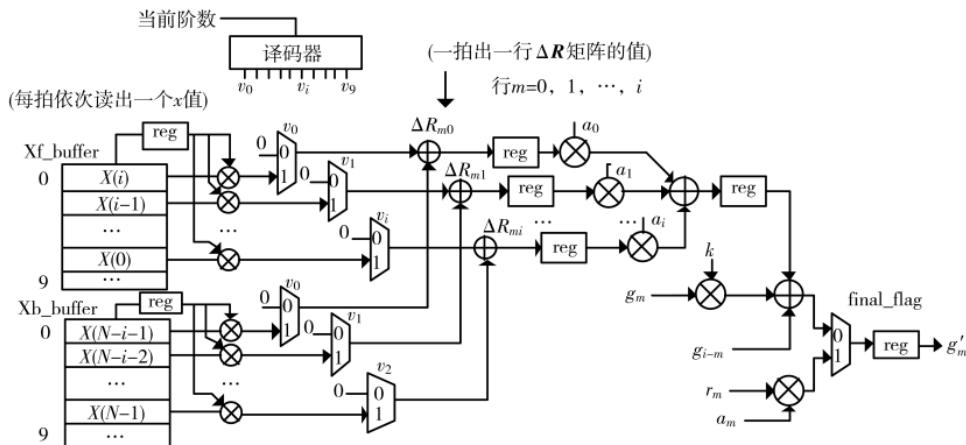


图 4 AR 参数和阈值计算单元顶层电路图

快速 Burg 算法的关键即为计算预备变量,包括  $r, \Delta R$  和  $g$  3 个变量。

$r$  列向量的计算电路为三级流水线结构,其电路如图 5 所示。第一拍锁存当前阶数下输入序列的第一个乘数,并将读入的数据写入  $X_f$ \_buffer 和  $X_b$ \_buffer;第二拍并行完成两次乘加运算;第三拍做选通输出  $c_1$  或乘加值,并倒序输出  $r$  列向量的行值。当前阶数下消耗的总计算时长为  $i+3$  个时钟周期。

中间变量矩阵  $\Delta R$  和列向量  $g$  的计算电路采用并行化处理,矩阵  $\Delta R$  和列向量  $g$  的四级流水线电路如图 6 所示。第一拍从两个 buffer 中各依次读出一个  $x$  值送入乘法器;第二拍做乘加运算,得到矩阵  $\Delta R$  的一行值;第三拍  $\Delta R$  的行值分别与上一阶求得的 AR 参数依次相乘并求和;第四拍选通输出列向量  $g$  的行值。当前阶数下消耗的总计算时长为  $i+5$  个时钟周期。

图 5 列向量  $r$  的运算电路图 6 矩阵  $\Delta R$  和列向量  $g$  的运算电路

## 2.4 存储单元设计

由于参数计算过程中部分变量需要在每阶下递归运算,因此需为其分配内存空间。为了起到加速的作用,使用双口 RAM 对中间结果进行存储,提高读写数据的速度。

## 3 结果分析

### 3.1 功率谱估计的分辨率

在本节中,当输入数据采样率为 512 Hz,输入数据点数为 512 时,通过 ModelSim 实验平台对基于定点数<sup>[13-14]</sup>的算法进行仿真,并将 FFT 算法和本算法的频率分辨率

进行比较。仿真实验中,AR 参数计算结果  $a$  和最小预测误差  $P_{\min}$  仿真结果如图 7 所示。

为了更直观地观察频谱结果,将硬件电路的结果传入 MATLAB 中,经过处理打印功率谱图,并将硬件电路计算出的定点数模型的功率谱结果与 MATLAB 直接计算的浮点数功率谱结果进行比较,如图 8 所示,谱峰一致说明量化良好。

实验设置输入数据为 15 Hz 和 20 Hz 正弦波的叠加信号,采样率为 512 Hz。

当输入数据长度为 512 时,FFT 算法和自适应定阶

◆ /tb/dk	1'd0	
◆ ◆ ◆ /tb/ar_i	4'd6	
◆ ◆ ◆ /tb/f_32_Q30	32'd449548534	
◆ ◆ ◆ /tb/a_32_Q30	36'd449548534	
◆ /tb/s2_end	1'd0	
◆ ◆ ◆ /tb/P_42_Q40	42'd1126	
◆ ◆ ◆ /tb/c_end	1'd0	
◆ ◆ ◆ /tb/is_ok	1'd0	
◆ ◆ ◆ /tb/mem_a[0]	36'd1073741824	
◆ ◆ ◆ /tb/mem_a[1]	-36'd2025745600	
◆ ◆ ◆ /tb/mem_a[2]	-36'd527308200	
◆ ◆ ◆ /tb/mem_a[3]	-36'd2073443486	
◆ ◆ ◆ /tb/mem_a[4]	36'd735634897	
◆ ◆ ◆ /tb/mem_a[5]	-36'd1861178463	
◆ ◆ ◆ /tb/mem_a[6]	-36'd55321111	
◆ ◆ ◆ /tb/mem_a[7]	36'd449548534	
◆ ◆ ◆ /tb/mem_a[8]	36'd0	
◆ ◆ ◆ /tb/mem_a[9]	36'd0	
◆ ◆ ◆ /tb/mem_a[10]	36'd0	

图 7 ModelSim 仿真结果

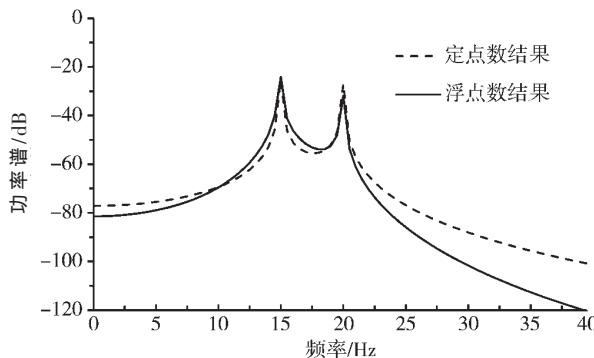


图 8 定浮点算法功率谱结果比较

的快速 Burg 算法的谱分析结果分别如图 9(a)和 9(b)所示,两种方式都可分辨出 15 Hz 和 20 Hz 两个谱峰,快速 Burg 算法的谱分析结果更为平滑,阶数为 7 阶;当输入数据长度为 128 时,上述两种算法的谱分析结果分别如图 9(c)和 9(d)所示,FFT 算法无法得到两个独立谱峰,而自适应定阶的快速 Burg 算法依然可以产生 15 Hz 和 20 Hz 两个独立谱峰,自适应定阶为 5 阶。

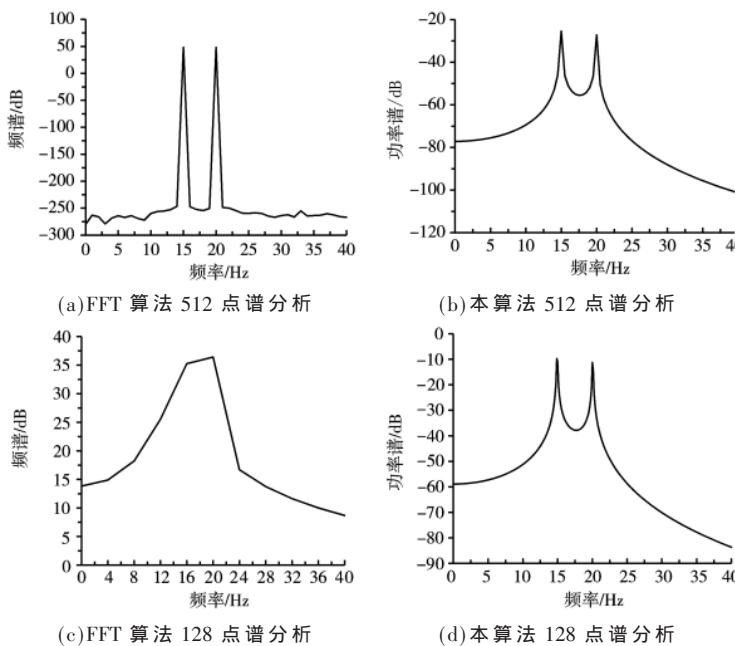


图 9 FFT 算法与自适应定阶快速 Burg 算法谱分析结果

综上,自适应定阶的快速 Burg 算法在短序列谱分析中具有更高的频率分辨率,在实际应用中,对实时性要求较高时,具有较为重要的意义。

### 3.2 计算时间

快速 Burg 算法与 Burg 算法相比,从硬件加速设计上分析,Burg 算法的前后向预测误差  $e_f, e_b$ ,反射系数  $k$ ,AR 参数  $a$ ,最小预测误差功率  $P_{min}$  的计算 4 组变量完全是串行关系,无法并行化<sup>[15]</sup>。Burg 算法计算单元在每阶下所消耗的时间如表 1 所示。

对于快速 Burg 算法,自相关的计算与其他变量无关,

表 1 Burg 算法计算单元消耗时钟数

计算单元	时钟数
前、后向预测误差 $e_f, e_b$	$N-i$
反射系数 $k$	$N-i$
AR 参数 $a$	$i+1$
最小预测误差 $P_{min}$	2

只与输入序列和当前阶数相关,它作为耗时最长的运算单元,可以提前一个流水线周期计算。在设计中采用并行的二级流水线结构,并对流水线进行灵活的启动终止控制,使该算法与加速前相比可以将整体计算速度提高约 50%。其各个计算单元所消耗的时钟数如表 2 所示。

表 2 快速 Burg 算法计算单元消耗时钟数

计算单元	时钟数
自相关	$N-i+3$
列向量 $r$	$i+3$
矩阵列向量 $g, \Delta R$	$i+5$
反射系数 $k$	$i+6$
AR 参数 $a$	$i+2$
最小预测误差 $P_{min}$ 和 FPE 值	3

最后,本文对两种算法在不同阶数下的硬件实现时计算单元所消耗的时钟数进行比较。以输入数据长度  $N=128$  为例,如表 3 所示,快速 Burg 算法的硬件加速实现方案确实可以起到更为有效的硬件加速作用,在 AR 阶数在 10 阶以内时,与 Burg 算法相比,可将计算时间降低 75%。

表 3 各算法在不同最终阶数时计算单元消耗时钟数

算法	Burg 算法硬件加速	快速 Burg 算法硬件加速
5 阶( $i=4$ )	1 285	295
6 阶( $i=5$ )	1 539	334
7 阶( $i=6$ )	1 792	430
8 阶( $i=7$ )	2 004	477
9 阶( $i=8$ )	2 295	513
10 阶( $i=9$ )	2 545	618

### 3.3 内存空间分析

快速 Burg 算法的硬件实现方案主要从反射系数  $k$  的计算中减少 RAM 的使用。其中自相关的计算由于流水线的设计无需存储器进行存储,列向量  $r$  和  $g$  需要进行存储,其存储深度等于阶数,在本设计中阶数在 2~10 之间,故深度为 10,宽度分别为 4 B 和 6 B。而 Burg 算法在计算反射系数  $k$  时,需要为前后向预测误差  $e_f, e_b$  这两个参数分配存储空间,所需 RAM 的宽度为 2 B,深度和输入数据长度  $N$  有关。因此,快速 Burg 算法的硬件实现明显更加节省面积。

## 3.4 硬件实现资源使用情况

采用如图 4 所示的自适应定阶快速 Burg 算法的实现方案在 Xilinx Virtex7 实验平台上做了仿真测试, 综合结果如表 4 所示。

表 4 资源使用情况

FPGA 资源	已用	可用	利用率/%
Number of Slice Registers	3 435	1 424 000	0.2
Number of Slice LUTs	9 725	712 000	1.3
Number of fully used LUT-FF pairs	1 924	4 989	38.6
Number of Block RAM/FIFO	6	1 880	0.3
Number of DSP48E1s	66	3 360	1.9

## 4 结论

本文以 Xilinx 公司的 Virtex7 开发板作为硬件平台, 完成了自适应定阶的快速 Burg 算法设计与实现。提出灵活控制的并行二级流水线结构和并行化计算单元结构, 达到了硬件加速的目标。同 Burg 算法相比, 其对短序列的处理在达到较高分辨率的同时, 计算时间可降低 75% 以上, 同时减小了内存空间, 并自适应定阶。实验发现, AR 模型阶数是否达到最优决定了功率谱质量的好坏, 但是此最优阶数的确定也与输入序列长度、采样率等相关, 这一关系有待进一步研究。

## 参考文献

- [1] 王世一. 数字信号处理(修订版)[M]. 北京: 北京理工大学出版社, 1997: 304–330.

(上接第 61 页)

Wi-Fi 的无线显示传输质量有所下降, 后续可在 Wi-Fi 标准、接收端解码等方面完善。

## 参考文献

- [1] 倪俊杰. 投屏技术及其教学应用[J]. 中国信息技术教育, 2017(9): 50–53.
- [2] 胡秋伟. 基于 Wi-Fi 网络的无线显示技术的设计与实现[J]. 电子技术与软件工程, 2016(22): 24–25.
- [3] 贺鹏. 基于 Miracast 的多媒体共享系统设计与实现[D]. 成都: 西南交通大学, 2016.
- [4] 编辑部. 拥抱 AirPlay2 让音乐和电影荡漾在家中的每个角落[J]. 家庭影院技术, 2019(11): 58–73.
- [5] 唐艳秋, 潘泓, 朱亚平, 等. 图像超分辨率重建研究综述[J]. 电子学报, 2020, 48(7): 1407–1420.
- [6] 杨悦, 谢辛, 何蕾, 等. 连分式插值结合卷积神经网络的超分辨率重建[J]. 合肥工业大学学报(自然科学版), 2021, 44(8): 1146–1152.
- [7] 刘颖, 朱丽, 林庆帆, 等. 图像超分辨率技术的回顾与展望[J]. 计算机科学与探索, 2020, 14(2): 181–199.
- [8] 刘永信, 段添添. 基于深度学习的图像超分辨率重建技术的研究[J]. 科技与创新, 2018(23): 40–43.

- [2] 邢务强, 钮金鑫. 基于 AR 模型的功率谱估计[J]. 现代电子技术, 2011, 34(7): 49–51.
- [3] 冯磊. 经典功率谱估计与现代功率谱估计的对比[J]. 商业文化(学术版), 2009(5): 239, 223.
- [4] COLLOMB C. Burg's method, algorithm and recursion[Z]. 2010.
- [5] BURG J P. Maximum entropy spectral analysis[C]//Proceeding of the 37th Annual International Meeting, Soc. of Explor Geophysics, 1967.
- [6] 张贤达. 现代信号处理(第三版)[M]. 北京: 清华大学出版社, 2015: 139–148.
- [7] 马忠强, 陈国通, 赵雪. 周期图法与 Burg 算法的对比研究[J]. 信息通信, 2017(4): 17–19.
- [8] CHAKRABORTY D, SANYAL S K. Performance analysis of different autoregressive methods for spectrum estimation along with their real time implementations[C]//2016 Second International Conference on Research in Computational Intelligence and Communication Networks(ICRCICN). IEEE, 2016: 141–146.
- [9] VOS K. A fast implementation of Burg's method[R]. OPUS Codec, 2013.
- [10] AKAIKE H. Fitting autoregressive models for prediction[J]. Annals of the Institute of Statistical Mathematics, 1969, 21(1): 243–247.
- [11] 张松, 李筠. FPGA 的模块化设计方法[J]. 电子测量与仪器学报, 2014, 28(5): 560–565.

(下转第 72 页)

- [9] 谢海平, 谢凯利, 杨海涛. 图像超分辨率方法研究进展[J]. 计算机工程与应用, 2020, 56(19): 34–41.
- [10] 李岚, 张云, 杜佳, 等. 基于改进残差亚像素卷积神经网络的超分辨率图像重建方法研究[J]. 长春师范大学学报, 2020, 39(8): 23–29.
- [11] 袁良凤. 基于 ARM 的 WiFi 无线通信终端技术探讨[J]. 数字技术应用, 2018, 36(5): 40–41.
- [12] 陈亮, 蔡声镇, 卢丽婉. 一种面向存储的高性能双向 SPI 传输机制[J]. 微电子学与计算机, 2021, 38(3): 72–76, 83.
- [13] 金海峰. TCP/IP 数据封装与传输过程分析与仿真[J]. 福建电脑, 2021, 37(7): 40–43.
- [14] 宋克. 一种高速可靠 UDP 网络传输的优化实现[J]. 数字通信世界, 2021(7): 85–86.
- [15] 顾燕卿. 一种可移植的 JPEG 解码库的实现[J]. 电子技术与软件工程, 2018(6): 68–70.

(收稿日期: 2021-08-23)

## 作者简介:

张银胜(1975-), 男, 博士, 副教授, 主要研究方向: 无线通信网络等。

徐文校(1997-), 男, 硕士研究生, 主要研究方向: 信号处理、深度学习等。



扫码下载电子文档

U2R 和 R2L, 本文模型仍需要进行优化改进。

### 参考文献

- [1] ANDERSON J P. Computer security threat monitoring and surveillance[R]. Fort Washington , Pennsylvania , 1980.
- [2] DENNING D E. An INTRUSION-DETECTION MODel[C]// IEEE Symposium on Security and Privacy , 1986 : 118–131.
- [3] BALABINE I , VELEDNITSKY A. Method and system for confident anomaly detection in computer network traffic : USA , 14/627 , 963[P]. 2015–02–20.
- [4] SIDDIQUI S , KHAN M S , FERENS K , et al. Detecting advanced persistent threats using fractal dimension based machine learning classification[C]// Proceedings of the 2016 ACM on International Workshop on Security And Privacy Analytics , 2016 : 64–69.
- [5] VELEA R , CIOBANU C , MARGARIT L , et al. Network traffic anomaly detection using shallow packet inspection and parallel K-means data clustering[J]. Studies in Informatics and Control , 2017 , 26(4) : 387–395.
- [6] LIU Y C , LIU S L , ZHAO X. Intrusion detection algorithm based on convolutional neural network[C]// Proceedings of 2017 4th International Conference on Engineering Technology and Application , 2017.
- [7] ZHOU H Y , WANG Y , LEI X C , et al. A method of improved CNN traffic classification[C]// Proceedings of 2017 13th International Conference on Computational Intelligence and Security , 2017 : 177–181.
- [8] JIHYUN K , HOWON K. Applying recurrent neural network to intrusion detection with hessian free optimization[C]// Proceedings of 16th International Workshop on Information Security Applications , 2015 : 357–369.
- [9] YIN C L , ZHU Y F , FEI J L , et al. A deep learning approach for intrusion detection using recurrent neural networks[J]. IEEE Access , 2017 , 5 : 954–961.
- [10] WANG H P , XU L , GU G F , et al. FloodGuard : a DoS attack prevention extension in software-defined networks[C]// Proceedings of IEEE/IFIP International Conference on Dependable Systems and Networks , 2015.
- [11] 汪洋 , 伍忠东 , 火忠彩. 基于 DBN-KELM 的入侵检测算法[J]. 计算机工程 , 2019 , 45(10) : 171–175 , 182.
- [12] 汪洋 , 伍忠东 , 朱婧. 基于深度序列加权核极限学习的入侵检测算法[J]. 计算机应用研究 , 2020 , 37(3) : 829–832.
- [13] SIMONYAN K , ZISSEMAN A. Very deep convolutional networks for large-scale image recognition[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition , 2014.
- [14] HOCHREITER S , SCHMIDHUBER J. Long short-term memory[J]. Neural Computation , 1997 , 9(8) : 1735–1780.
- [15] Li Feifei. ImageNet : crowdsourcing, benchmarking & other cool things[C]// CMU VASC Seminar , 2010 , 16 : 18–25.
- [16] Information and Computer Science , University of California. KDD Cup 1999 Data[DB/OL]. (1999–10–28)[2021–02–09]. <http://kdd.ics.uci.edu/databases/kddcup99>.
- [17] 任家东 , 刘新倩 , 王倩 , 等. 基于 KNN 离群点检测和随机森林的多层次入侵检测方法[J]. 计算机研究与发展 , 2019 , 56(3) : 566–575.
- [18] GAO N , GAO L , GAO Q L , et al. An intrusion detection model based on deep belief networks[C]// Proceedings of the IEEE Conference on Advanced Cloud and Big Data , 2014 : 247–252.
- [19] 贾凡 , 孔令智. 基于卷积神经网络的入侵检测算法[J]. 北京理工大学学报 , 2017 , 37(12) : 1271–1275.
- [20] SHEIKHAN M , JADIDI Z , FARROKHI A. Intrusion detection using rediced-size RNN based on feature grouping[J]. Neural Computing and Applications , 2012 , 21(6) : 1185–1190.

(收稿日期 : 2021–02–09)

### 作者简介 :

郭卫霞(1994–), 女, 硕士研究生, 助理工程师, 主要研究方向: 深度学习、信息安全。

张伟(1976–), 男, 硕士研究生, 高级工程师, 主要研究方向: 网络安全。

杨国玉(1980–), 男, 硕士研究生, 高级经济师, 主要研究方向: 信息化与网络安全管理。



扫码下载电子文档

(上接第 67 页)

- [12] KAJIHARA K , IZUMI S , YOSHIDA S , et al. Hardware implementation of autoregressive model estimation using Burg's method for low-energy spectral analysis[C]// 2018 IEEE International Workshop on Signal Processing Systems (SiPS). IEEE , 2018 : 199–204.
- [13] 苏丽. Matlab 定点仿真在 FPGA 验证平台中的应用[J]. 电子科技 , 2013 , 26(5) : 71–73.
- [14] 张林生. 数字信号处理系统的定点化技术研究[D]. 哈尔滨 : 哈尔滨工业大学 , 2010.

- [15] 钱莹晶 , 周群. Burg 频谱估计算法的硬件加速方法研究[J]. 电子测量与仪器学报 , 2015 , 29(9) : 1382–1390.

(收稿日期 : 2021–02–20)

### 作者简介 :

郭鸣晗(1996–), 女, 硕士, 主要研究方向: 数字信号处理、数字集成电路设计。

陈立平(1983–), 男, 硕士, 高级工程师, 主要研究方向: 生物信号处理、SoC 设计。

张浩(1975–), 通信作者, 男, 博士, 副研究员, 主要研究方向: 生物信号处理、SoC 设计, E-mail : haozhang@ime.ac.cn。



扫码下载电子文档

## 版权声明

经作者授权，本论文版权和信息网络传播权归属于《电子技术应用》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、DOAJ、美国《乌利希期刊指南》、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《电子技术应用》编辑部

中国电子信息产业集团有限公司第六研究所