

现场可编程门阵列参数化多标准高吞吐率基4 Viterbi 译码器*

夏飞¹, 聂晶¹, 李荣春², 王文涛³

(1. 海军工程大学 电子工程学院, 湖北 武汉 430033; 2. 国防科技大学 计算机学院, 湖南 长沙 410073;

3. 中国人民解放军 91033 部队, 山东 青岛 266034)

摘要:为了同时达到高性能和灵活性的目标,提出一种基于现场可编程门阵列的参数化多标准自适应基4 Viterbi 译码器。译码器采用3~9可变约束长度,1/2、1/3可变码率,支持任意截断长度的纠错译码,并采用码字无符号量化、加比选单元设计优化和归一化判断逻辑分离策略优化关键路径设计,提高译码器工作频率。实验结果表明,该译码器能根据用户设定的参数改变结构,在多种通信标准之间实现动态切换;性能达到了541 Mbps,明显优于相关工作;对GPRS, WiMAX, LTE, CDMA, 3G等通信标准都取得了良好的误码性能,可满足多种通信标准的译码需求。

关键词:现场可编程门阵列; Viterbi 译码器; 参数化; 多标准; 基4

中图分类号: TN92 **文献标志码:** A **文章编号:** 1001-2486(2016)01-086-07

Parameterized multi-standard high-throughput radix-4 Viterbi decoder on field-programmable gate array

XIA Fei¹, NIE Jing¹, LI Rongchun², WANG Wentao³

(1. Electronic Engineering College, Naval University of Engineering, Wuhan 430033, China;

2. College of Computer Science, National University of Defense Technology, Changsha 410073, China;

3. The PLA Unit 91033, Qingdao 266034, China)

Abstract: To achieve the goal of high performance and flexibility, a parameterized multi-standard adaptive radix-4 Viterbi decoder based on the field-programmable gate array was presented. This decoder adopts constraint lengths ranging from 3 to 9, code rates of 1/2 or 1/3 and supports error-correcting decoding of arbitrary truncation lengths. The unsigned quantization, add-compare-select unit optimization and normalization judgment logic separation strategies were used to optimize the design of critical path, so that it can improve system throughput. Experiment results show that: the decoder can change the structures according to the parameters set by users and achieve dynamic switching in multiple communication standards; the throughput can reach up to 541 Mbps, apparently superior to the related works; the decoder achieves low bit error ratio in multiple standards such as GPRS, WiMax, LTE, CDMA and 3G and satisfies the decoding requirements of multiple communication standards.

Key words: field-programmable gate array; Viterbi decoder; parameterization; multi-standard; radix-4

随着通信技术日新月异的革新和各种无线设备的诞生,通信标准越来越呈现多样化的特点。该特点决定了移动终端需要具备针对多种通信标准的自适应能力,便于用户在多个网络中实现通信。因此需要开发一种可以兼容多个通信标准,具备多模式、多功能的无线通信系统。该系统可根据用户提供的参数自适应地变换系统结构,选择实现某种功能,以适应当前网络标准的通信需求。在上述方案中,系统只包含一个多功能部件,根据用户需求选择参数,动态变化结构,满足实时通信的需求。这对于移动设备和无线基站来说,

并不需要额外的结构设计,就可以用较小逻辑资源代价实现多个通信标准间的实时切换,从而有效提高无线设备的灵活性和可用度。

1 研究背景

Viterbi 算法是目前使用最广泛的卷积码信道译码算法。多标准 Viterbi 译码器可以通过实时调整参数,在多个通信标准间动态切换,实现相应的信道译码。表1列出了不同通信标准下,多标准 Viterbi 译码器需要支持的码率、约束长度、截断长度等参数。从表中可以看到,常用的通信标

* 收稿日期:2015-02-13

基金项目:国家自然科学基金资助项目(61202127);湖南省学位与研究生教育专项基金资助项目(YB2013B008)

作者简介:夏飞(1980—),男,湖南常德人,助理研究员,博士,E-mail:xcyphoenix@nudt.edu.cn;

聂晶(通信作者),女,讲师,硕士,E-mail: cindyrany222@163.com

准码率都在 $1/2$ 和 $1/3$ 之间变化,而约束长度在 $5 \sim 9$ 之间变化。

表1 不同通信标准下的典型信道参数

Tab.1 Channel parameters for different communication standards

标准	码率	约束长度	状态数
GPRS	$1/2$	5	16
WiMAX	$1/2$	7	64
802.11a/g	$1/2$	7	64
DVB	$1/2$	7	64
CDMA IS-95A	$1/2$	7	64
LTE	$1/3$	7	64
CDMA2000	$1/2, 1/3$	9	256

目前 Viterbi 译码器的研究和实现主要基于数字信号处理器 (Digital Signal Processor, DSP)、专用集成电路 (Application Specific Integrated Circuit, ASIC) 和现场可编程门阵列 (Field-Programmable Gate Array, FPGA) 三类计算平台。由于 DSP 芯片只能支持固定指令集,其灵活性和并行度都受到限制,译码器的吞吐率一般都在 20 Mbps 以下。ASIC 平台拥有面积小、性能高和功耗低的优势,但是开发周期长,设计成本高,而且结构和功能固定,灵活性较差。FPGA 则具有成本低、灵活性强的优势,易将多种功能融合到单芯片逻辑结构中。而对多模式多功能无线通信系统来说,必须兼有软件的灵活性和硬件的高性能。因此,FPGA 芯片以其灵活的算法适应性、细粒度并行能力、低硬件代价和高性能功耗比成为参数化多标准 Viterbi 译码器最理想的实现平台。

Viterbi 译码算法主要有基 2 和基 4 两种。基 4 Viterbi 译码器吞吐率较高,但是硬件逻辑实现相对复杂。随着通信技术的迅速发展,通信系统呈现多样性及高带宽的特点,迫切需要吞吐率高且灵活性强的 Viterbi 译码器。本文基于上述需求,在 FPGA 平台上首次实现了参数化多标准基 4 Viterbi 译码器,该译码器可实现 $3 \sim 9$ 可变约束长度, $1/2/1/3$ 可变码率和任意截断长度的纠错译码,性能达到 541 Mbps ,明显优于相关工作。

2 相关工作

根据算法类型和灵活性,可将现有的基于 FPGA 的 Viterbi 译码器分为三类^[1-11],分别是:基 2 固定结构译码器^[1]、基 2 多标准译码器^[2]和

基 4 固定结构译码器^[3-7]。

基 2 固定结构 Viterbi 译码器功能固定,只能支持某一种通信标准的信道译码^[1]。基 2 多标准译码器与基 2 固定结构译码器实现的算法相同,但译码器中包含了多个功能模块,可以通过调整参数实现多种功能间的动态切换,从而支持不同通信标准的信道译码。Batcha 等^[2]在 FPGA 上实现了 5 和 7 两种约束长度的 Viterbi 译码器,其吞吐率为 150 Mbps ,是目前速度最快的多标准 Viterbi 译码器。基 2 多标准译码器虽然具有较好的灵活性,但是多个功能模块占据了大量逻辑资源,导致其吞吐率不够,不能满足某些通信系统,如 UWB 的 200 Mbps 译码速率要求。

相对于基 2 译码器,基 4 固定结构 Viterbi 译码器性能更高,但复杂度显著增加。在频率相同的情况下,其吞吐率是基 2 译码器的两倍。Santhi 等^[6]实现了约束长度为 7、码率为 $1/3$ 的基 4 Viterbi 译码器。该译码器采用两级流水乒乓结构,支持两帧同时译码,吞吐率为 274 Mbps ,是目前性能最高的基 4 Viterbi 译码器。但是由于其结构固定,不能满足多个通信标准的译码需求,灵活性不足。

为了兼顾高性能和灵活性,提出了参数化多标准基 4 Viterbi 译码器。译码器采用无符号度量计算、新的加比选单元设计和归一化判断逻辑分离策略优化关键路径,工作频率达到 270.5 MHz ,性能达到 541 Mbps ,是目前基于 FPGA 实现的最高性能基 2 多标准译码器^[2]的 3.60 倍,是当前最高性能的基 4 固定结构译码器^[6]的 1.97 倍。此外,现有的基 4 Viterbi 译码器只支持 6 和 7 两种^[3-7]约束长度,无法满足多种通信标准需求,而本文实现的译码器可支持 $3 \sim 9$ 的约束长度和任意截断长度的纠错译码,不但性能更高而且灵活性更强,可满足多种通信标准的译码需求。

3 Viterbi 译码基本原理

Viterbi 算法是一种用于解决有限状态离散时间马尔科夫链的状态估计问题的优化算法。由于 Viterbi 算法计算复杂度较低的特性,其被广泛用于数字通信系统中卷积码的译码,并取得了良好的译码性能。

图 1 所示的状态迁移图显示了卷积码译码流程,图中的时间节点数为信息码字的个数。Viterbi 译码器从网格图中找到最大似然路径。在 t 时刻,网格图中每个分支都必须计算状态 i 到状态 j 的分支度量 $BM_t(i, j)$,每个状态 i 计算路径度

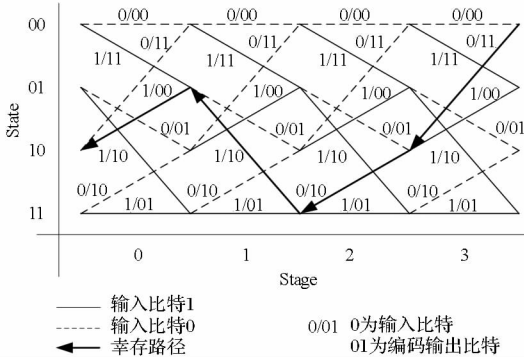


图 1 基 2 Viterbi 译码网格图

Fig. 1 Radix-2 trellis and Radix-2 Viterbi decoding algorithm

量 $PM_t(i)$ 。每个状态完成两个分支度量和两个路径度量的加比选操作,然后选择较大的路径度量作为该状态的路径度量,并将对应的路径使用幸存比特 $SB_t(i)$ 的形式存储在该状态。在正向计算生成网格之后,开始逆向回溯寻找信息比特。每个状态存储一个幸存比特,用于标识幸存路径上到达该状态的前一个状态。通过幸存路径上存储的幸存比特就可以回溯找到最大似然路径,对应的信息比特序列就是最后输出的译码信息。不同的通信标准对应着不同的网格图。图中的状态数等于 2^{K-1} ,其中 K 为约束长度;卷积码编码多项式不同,对应网格图中的编码输出也不同;分支度量的计算也会随着码率的变化而有所变化。

根据网格图构造不同,Viterbi 算法可分为基 2 Viterbi 算法和基 4 Viterbi 算法,其区别如图 2 所示。图 2(a)为基 2 结构,每个节点有两个输入,完成两个分支度量和两个路径度量的加比选操作。图 2(b)为基 4 结构,每个节点有 4 个输入,

完成 4 个分支度量和 4 个路径度量的加比选操作。其中连续两个时间节点 4 个状态的基 2 网格可以通过一个时间节点 4 个状态的基 4 网格来实现。图 2 中 $PM_t(i)$ 是指 t 时刻状态 i 的路径度量值。图 2(b)中 $BM_t(2,0)$ 是指基 4 网格中 t 时刻状态 2 到 0 的分支度量值,它等于图 2(a)中基 2 网格 $BM_{t-1}(2,0)$ 与 $BM_t(0,0)$ 之和。

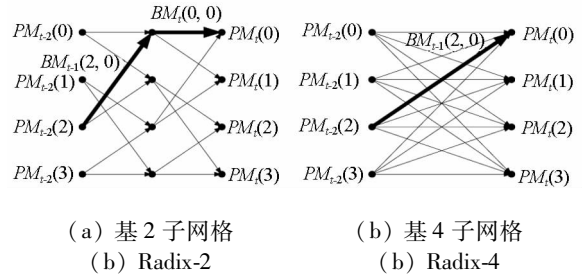


图 2 基 2 和基 4 子网格图

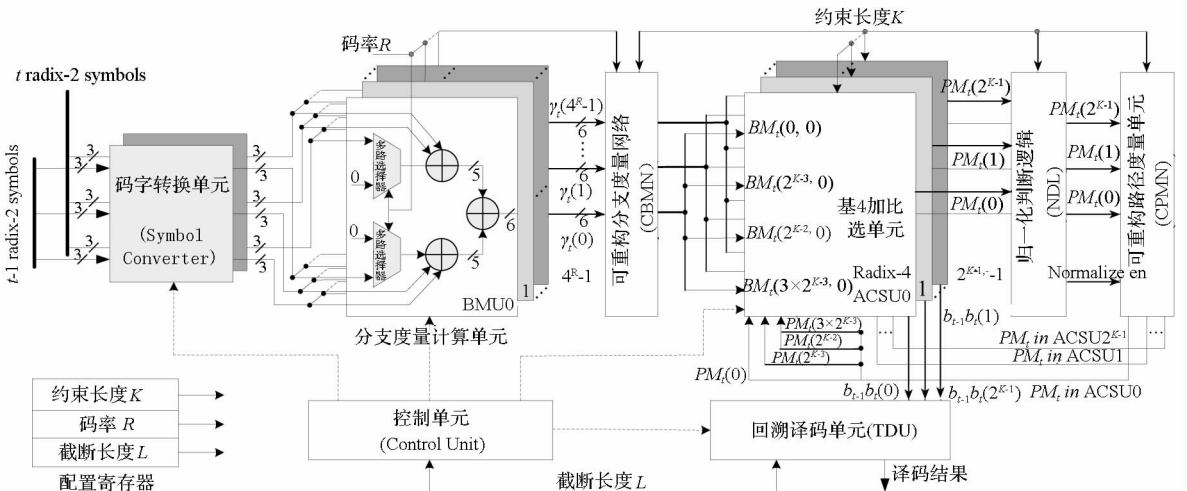
Fig. 2 Radix-2 and Radix-4 sub-trellis

4 参数化多标准 Viterbi 译码器体系结构

4.1 Viterbi 译码器总体结构

参数化多标准基 4 译码器可在工作状态下通过动态参数配置实时变换结构,以适应当前选择的通信标准,并通过自定义截断长度,自动寻找合适的译码精度和译码延迟。

如图 3 所示,译码器由控制单元 (Control Unit, CU)、码字转换单元 (Symbol Converter, SC)、分支度量计算单元 (Branch Measure Unit, BMU)、可重构分支度量网络 (Configurable Branch Measure Network, CBMN)、基 4 加比选单元 (R4 Add Compare and Select Unit, R4 ACSU)、归一化



b ——输出的比特位; $\gamma_t(i)$ ——分支度量值,括号中的数值为序号。

图 3 参数化多标准 Viterbi 译码器总体结构

Fig. 3 Architecture of the proposed parameterized multi-standard Viterbi decoder

判断逻辑 (Normalization Decision Logic, NDL)、可重构路径度量网络 (Configurable Path Measure Network, CPMN) 以及回溯译码单元 (Traceback Decode Unit, TDU) 8 个部分组成。其译码流程如下: 首先, 译码器接收输入码字, 将码字转化成无符号整数, 将其输入到 BMU。BMU 计算网格中每种可能的分支度量值, 通过 CBMN 将正确的分支度量分别送入对应的基 4 ACSU。基 4 ACSU 计算出新的路径度量, 送入 NDL, 并将幸存比特信息送入 TDU。NDL 判断路径度量是否需要归一化。归一化后的路径度量值送入 CPMN, 并根据约束长度的变化, 自适应选择合适的互联关系, 再将新的路径度量值送回基 4 ACSU 进行迭代计算。TDU 执行 3 点回溯算法, 对网格进行回溯和译码, 输出译码比特。CU 负责对整个译码算法执行过程进行流水控制。

译码器使用配置寄存器存放参数。在工作时, 通过更新参数实现译码器结构的实时重构。码率 R 用来重构 BMU 和 CBMN 单元。整个译码器共实例化 4^3 个 BMU 单元, 在 64 个 BMU 单元中选择其中的 4^R 个, 并且根据 R 选择 BMU 计算中的第 3 个码字。约束长度 K 用来重构 R4 ACSU, NDL, CBMN, CPMN 模块。译码器根据 R 和 K 值重构 CBMN 的互联结构, 根据 K 重构 CPMN 的互联结构。译码器共实例化 2^8 个 R4 ACSU 单元, 根据码率 R 在 256 个 BMU 单元中选择其中的 2^{K-1} 个。采用截断译码方式 (截断长度根据参数 L 确定), 在可重构时送入控制单元和回溯译码单元, 通过流水线实现截断长度 L 的 3 点译码算法。在约束度不同的情况下, 利用参数 L 配置, 可以有效地实现译码精度和译码延迟的折中。

4.2 参数化多标准实现

4.2.1 码率 R 的参数化

由于码率 R 的变化会引起 BMU 和 CBMN 结构的变化, 为了实现码率 R 的参数化, 上述两个模块的结构必须可根据参数 R 实时可重构。

如图 3 所示, BMU 的结构由两个选择器和三个加法器组成。当码率为 $1/2$ 时, 选择器选择 0; 当码率为 $1/3$ 时, 选择器选择增加的第三个码字。随后将三个输入相加, 得到基 2 的分支度量; 再计算相应的基 4 分支度量。 $t-1$ 和 t 时刻的两个基 2 分支度量同时计算, 并且 BMU 采用流水线结构, 避免其逻辑延迟成为译码器的瓶颈。

CBMN 主要将集中计算的分支度量值分布到 ACSU 中。CBMN 可根据约束长度的变化形成相应的映射关系。CBMN 对于每种约束长度都有固

定的互联结构, 当码率 R 分别为 $1/2$ 和 $1/3$ 时, 约束长度 5, 7, 9 都对应固定的互联结构。CBMN 利用选择器实现固定互联结构的自适应重构。当接收到约束长度参数 K 后, CBMN 利用选择器根据约束长度选择对应的互联结构。由于 CBMN 并不在迭代计算路径中, 故通过流水线设计避免其成为逻辑瓶颈。

4.2.2 约束长度 K 的参数化

与码率 R 类似, 约束长度 K 的变化会引起 ACSU、CBMN、CPMN 结构的变化。为了实现 K 的参数化, ACSU、NDL、CBMN、CPMN 的结构必须可根据参数 K 实时可重构。这里主要介绍 ACSU 和 CPMN 单元的可重构设计。

译码器中 ACSU 的数量与 K 相关, 为 2^{K-1} 个, 其支持的最大约束长度为 9, 所以共实例化了 256 个 ACSU。译码器工作时, 通过当前输入的参数 K 实时重构 ACSU 的数量。控制单元每次选择使用前 2^{K-1} 个, 并屏蔽其余的 ACSU。

CPMN 用于连接连续两个时间节点每个状态的路径度量值, 并根据约束长度 K 的变化选择合适的互联结构, 实现基 4 网格图相邻时刻状态间的互联关系。其中 $2i$ 状态和 $2i+1$ 状态的输入路径度量等于 i 状态和 $i+2^{K-2}$ 状态的输出路径度量。由于相邻时刻的基 4 子网格可以由连续两个时刻的基 2 子网格构成, 所以将 $t-2$ 时刻到 t 时刻的基 2 子网格中两个时刻的网格图合并, 可得基 4 网格中相邻时刻状态间的互联关系。 $4i$ 状态、 $4i+1$ 状态、 $4i+2$ 状态、 $4i+3$ 状态的输入路径度量等于 i 状态、 $i+2^{K-3}$ 状态、 $i+2^{K-2}$ 状态、 $i+3 \times 2^{K-3}$ 状态的输出路径度量。CPMN 通过接收约束长度参数, 实时变换相应的互联结构, 正确连接相邻时刻的路径度量值, 实现路径度量的迭代计算。

4.3 高吞吐率优化

译码器的性能受限于芯片逻辑的关键路径。为了提高系统吞吐率, 采用了无符号量化、无符号基 4 ACSU 设计和有效的归一化策略尽可能缩短译码器的关键路径, 并采用流水线实现 3 点译码算法, 有效提高了译码器的吞吐率。

4.3.1 无符号量化

无符号量化通过码字转换 (SC) 单元实现。SC 单元的主要任务是将带符号的码字及码字补码转化成无符号整数。由于译码器采用 3 比特量化, 所以码字及其补码的表示范围为 -4 到 3 , 量化振幅为 4。为了将其转化成无符号整数, 在度量转化单元中, 先计算每个码字的补码, 然后将码字和其补码同时加上量化振幅, 转化成 0 到 7 之

间的整数。其目的是在进行加比选计算时,简化逻辑,降低路径延迟,从而提高工作频率。

4.3.2 无符号运算的基 4 ACSU 设计

基 4 ACSU 用于实现加比选操作,是译码器设计的关键路径。常用的两种设计方案是采用两级级联比较结构和一级比较结构。前者采用两级比较结构来完成 4 个输入求最大值的过程,设计简单,但拉长了关键路径。为了缩短关键路径,故采用一级比较结构。如图 4 所示,利用四个加法器将输入的路径度量值和分值度量值相加,将 4 个结果同时送入 6 个比较器进行两两比较,然后将结果送入选择逻辑,得到幸存比特,利用选择逻辑在 4 个结果中选择最大值作为最后的路径度量输出。ACSU 采用组合逻辑实现,如果采用带符号数补码的比较和加法^[6-7],必须考虑符号操作。如图 4(a)所示,有符号的比较器将待比较的两个数的非符号位减法结果与两个符号位进行异或,得到比较结果,逻辑实现比较复杂。而使用无符号整数作为度量值,通过图 4(b)所示的无符号比较器即可实现比较功能。与图 4(a)相比,通过逻辑简化,ACSU 的关键路径从 8 个门延迟降低为 4 个门延迟,也就意味着在比较器设计的关键路径上降低了 4 个门延迟。

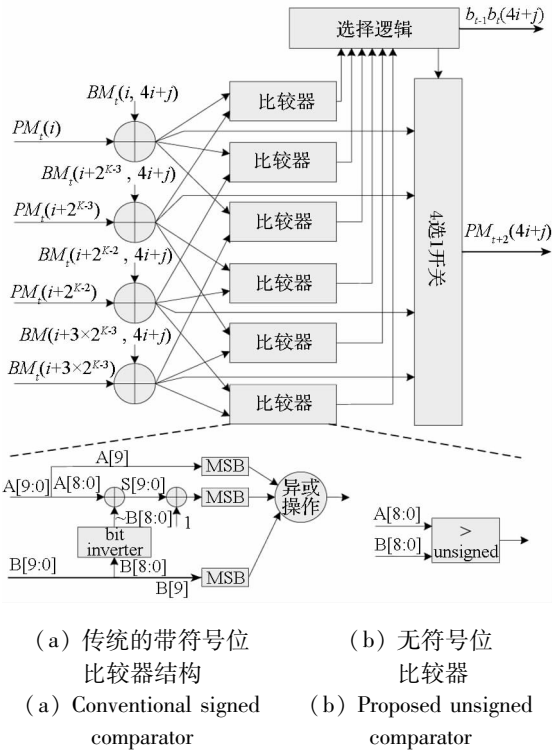


图 4 基 4 加比选计算结构图
Fig. 4 Fabric of R4 ACSU

4.3.3 归一化实现

归一化是为了防止路径度量值溢出而将所有

度量值都减去一个固定值。归一化的方法主要有以下几种^[8]:复位为状态 0 方法、减去最小值方法、MSB(最高有效位)位置零法、模归一化方法以及最高位清零方法。文献[8]对上述方法进行了模拟验证,证实了最高位清零法速度最快、功耗最低、面积最小,因此本文采用了该方法。不同的是,本文将归一化判断逻辑和加比选单元分离,利用两级流水的与操作来判断所有路径度量值最高位是否为 1。若都为 1,则在下一个时钟周期对所有度量值的最高位进行清零。其目的是将归一化判断逻辑延迟排除在关键路径之外。需要注意的是,判断逻辑只取前 2^{k-1} 最高位进行与操作。

4.3.4 三点译码算法的流水线实现

Viterbi 译码器一般有两种回溯方式:寄存器交换法和回溯法^[9-10]。前者实现简单,但是逻辑量过于庞大,其复杂的交换网络也会对设计频率造成影响。回溯法虽然占用了一定的存储资源,但是具有低复杂度和高频率的优点。

Viterbi 算法在回溯时,不论从哪个状态开始,当达到某个特定长度之后,其回溯状态将汇集到某个固定的状态,然后可以再从这个固定状态往前回溯最终得到译码比特。根据算法的上述特点,一个块的操作主要分为前向计算、回溯和译码,因此称为三点回溯算法^[11],三部分操作采用流水实现。其中前向计算包括分支度量和加比选计算。本文采用了如图 5 所示的回溯算法。每个矩形块对应一个截断长度的前向计算、回溯、译码操作。上文图 3 中的控制器(CU)完成算法的流水控制工作。截断长度 L 、约束长度 K 和码率 R 均作为参数输入,用于实现译码器重构。其原因是:不同通信标准最合适的截断长度不同。如果小于这个长度,会对译码精度造成一定的损失,而大于这个长度,则会增大译码延迟。为了找到译码精度和延迟的折中,将截断长度作为参数输入,以便对每种通信标准都能找到两者的最佳平衡点。 L 的变化通过设置控制器中计数器的上限值来实现。设计使用的存储器深度为 $4L$,经过 $4L$

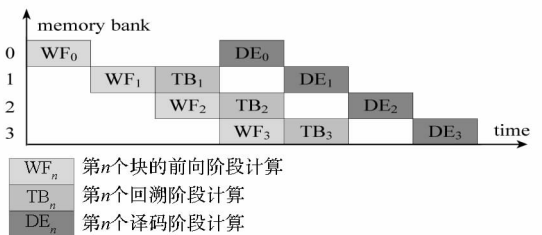


图 5 块译码流水线时空图
Fig. 5 Pipeline space-time diagram of block decoding

个时钟延迟后,开始译出第一个块的比特。图 5 中译码输出时采用移位寄存,在块末尾统一输出,不需要后进先出,从而简化了实现逻辑。

5 实验验证与性能对比

5.1 实验平台及验证方法

在 V7 系列 XC7VX485T FPGA 上实现参数化多标准高吞吐率基 4 Viterbi 译码器。如图 6 所示,系统使用了两块 FPGA 开发板,集成了无线射频装置。发射端先对二进制信号源进行卷积码编码,再进行调制、数字发射后送入无线信道;接收端对信号进行数字接收、解调后,进行 Viterbi 译码。在得到译码结果后,将结果和输入的二进制信号通过软件比较,得到译码器的误码率。用户将设定的通信标准输入到自适应控制器,将其转化成一系列参数,传送给编码器和参数化多标准 Viterbi 译码器。译码器根据输入的参数动态改变自身结构,在不需要全系统重构的情况下实时构建针对当前通信标准的信道译码器。

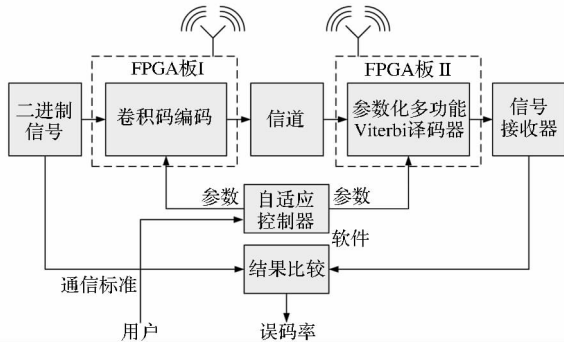


图 6 自适应无线通信系统结构框图

Fig. 6 Adaptive wireless communication system

5.2 FPGA 实现结果

使用 ISE10.1 工具链进行综合、布局、布线,表 2 列出了设计实现结果。为了便于对比,同时实现了约束长度 K 为 9 的基 2 和基 4 固定结构,以及基 2 和基 4 参数化多标准 Viterbi 译码器。从表 2 可以看出,基 2 多标准结构与固定结构相比,资源开销增加 23.6%,而基 4 多标准结构与固定结构相比,资源开销增加 30.1%,与基 2 相当。增加的部分主要用于加比选单元的逻辑实现。基 4 参数化多标准结构使用了 7823 个逻辑单元 (Slices),占芯片总逻辑资源的 15.1%,其中 14% 为 CPMM 和 CBMM 单元占据。由于采用了无符号比较和归一化判断逻辑流水线策略,关键路径 ACSU 的频率有了显著提升。实验结果表明,译码器频率达到 270.5 MHz,吞吐

率达到 541.0 Mbps,与目前最高性能的基 2 多标准译码器^[2]相比,性能提升了 2.6 倍,与最高性能基 4 固定结构 Viterbi 译码器^[6]相比,性能提升了 97%。

表 2 基 2 和基 4 Viterbi 译码器 FPGA 实现结果

Tab. 2 Device utilization summary of radix-2 and radix-4 Viterbi decoders on FPGA

基数	结构	资源利用/ Slices	资源 开销	频率/ MHz	吞吐量/ Mbps
Radix-2	固定 ($K=9$)	2744 (5.3%)		290.32	290.32
	多标准	3393 (6.5%)	23.6%	285.66	285.66
Radix-4	固定 ($K=9$)	6013 (11.6%)		280.91	561.82
	多标准	7823 (15.1%)	30.1%	270.50	541.00

5.3 误码性能

为了验证译码器的误码性能,本文实现了 GPRS, WiMAX, LTE (3GPP-Long Term Evolution), CDMA IS-95A 和 3G 五种通信标准的信道译码。信道采用高斯信道,信噪比变化为 0 到 5 dB。针对每种通信标准,分别在其约束长度的 5, 6, 7 倍截断长度下测试其误码率。实验结果证明实现的定点译码器误码率同软件模拟的浮点译码器相当。这也进一步证明提出的译码器具有良好的灵活性和实用性。

5.4 性能比较

表 3 列出了基于 FPGA 平台的基 2 多标准和基 4 固定结构 Viterbi 译码器的典型参数和性能,主要包括基数、约束长度、码率、频率和吞吐量。目前只有基于 FPGA 的基 4 固定结构的 Viterbi 译码器,实现约束长度只有 6 和 7 两种,而没有支持多标准的基 4 Viterbi 译码器,因此无法与相关工作进行直接对比。表 3 说明,本文实现的译码器不但性能更高,而且灵活性更强,可以支持约束长度在 3~9 间变化,并支持 GPRS, WiMAX, LTE, CDMA IS-95A, 3G 等多种通信标准。

6 结论

基于 Xilinx XC7VX485T FPGA 芯片首次提出并实现了基 4 参数化多标准 Viterbi 译码器。该译码器可在工作的同时,通过动态参数配置实时变换结构,在多种功能间切换,实现对通信标准的

表 3 FPGA 平台上不同 Viterbi 译码器的相关参数及性能比较

Tab. 3 Parameters and performance of the different Viterbi decoders on FPGA platform

相关工作	基数	约束长度	码率	FPGA 型号	频率/MHz	吞吐量/Mbps
R2decoder ^[2]	2	5,7	1/2,1/3	Altera APEXII 20K200E	150	150
R4 decoder ^[3]	4	7	1/3		119	238
R4 Butterfly ^[4]	4	6	1/4		41.7	83.4
R4Tail-biting ^[5]	4	7	1/3		100	120
R4 Two-stage ^[6]	4	7	1/3	Altera EP3SE80F1152C2	68.5	274
Our Multi-standard R4 Decoder	2	3~9	1/2,1/3	Altera APEXII 20K200E	163.53	163.53
				Altera EP3SE80F1152C2	145.37	290.74
	4	3~9	1/2,1/3	Xilinx XC7VX485T	270.50	541.00

自适应,并通过自定义截断长度达到最佳译码精度和延迟。此外,译码器实现了 3~9 可变约束长度,1/2 与 1/3 可变码率和任意截断长度的纠错译码,并采用无符号量化、加比选单元设计优化和归一化判断逻辑分离策略优化了关键路径设计,提高了系统吞吐率。实验结果表明,实现的参数化多标准基 4 译码器吞吐率可达 541 Mbps,是目前基于 FPGA 的最高性能基 2 多标准译码器^[2]的 3.60 倍,是目前最高性能的基 4 固定结构 Viterbi 译码器^[6]的 1.97 倍,其对多种通信标准都取得了良好的误码性能,无论是吞吐率还是灵活性都明显优于相关工作。

参考文献 (References)

- [1] Wang L O, Li Z Y. Design and implementation of a parallel processing Viterbi decoder using FPGA [C]//Proceedings of International Conference on Artificial Intelligence and Education, Hangzhou, China, 2010: 77-80.
- [2] Batcha M F N, Sha' Ameri A Z. Configurable adaptive Viterbi decoder for GPRS, EDGE and WiMAX [C]//Proceedings of IEEE International Conference on Telecommunications and International Conference on Communications, Malaysia, 2007: 237-241.
- [3] Byun S, Lee S, Park S C. Implementation of the modified state mapping Viterbi decoder with radix-4 [C]//Proceedings of International Conference on Communication Technology, China, 2006: 1-4.
- [4] Hsu Y H, Hsu C Y, Kuo T S. Low complexity radix-4 butterfly design for the Viterbi decoder [C]//Proceedings of

IEEE 64th Vehicular Technology Conference, Montreal, Canada, 2006: 1-5.

- [5] Abdallah R A, Lee S J, Goel M, et al. Low-power pre-decoding based Viterbi decoder for tail-biting convolutional codes [C]//Proceedings of IEEE Workshop on Signal Processing Systems, Finland, 2009: 185-190.
- [6] Santhi M, Lakshminarayanan G, Sundaram R, et al. Synchronous pipelined two-stage radix-4 200Mbps MB-OFDM UWB Viterbi decoder on FPGA [C]//Proceedings of International SoC Design Conference, Busan, Korea, 2009: 468-471.
- [7] Choi S W, Kang K M, Sang-Sung S S. A two-stage radix-4 Viterbi decoder for multiband OFDM UWB systems [J]. ETRI Journal, 2008, 30(6): 850-852.
- [8] Lai K Y T. A high-speed low-power pipelined Viterbi decoder: breaking the ACS-bottleneck [C]//Proceedings of 2010 International Conference on Green Circuits and Systems, Shanghai, China, 2010: 334-337.
- [9] Haridas S L, Choudhari N K. Design of Viterbi decoder with minimum transition hybrid register exchange processing [C]//Proceedings of International Conference and Workshop on Emerging Trends in Technology, Mumbai, India, 2010: 432-434.
- [10] Khatri D M, Haridas S L. Soft output Viterbi decoder using hybrid register exchange [C]//Proceedings of International Conference and Workshop on Emerging Trends in Technology, Bangalore, India, 2011: 942-945.
- [11] 王建新, 于贵智. Viterbi 译码器回溯算法实现研究 [J]. 电子与信息学报, 2007, 29(2): 278-282.
WANG Jianxin, YU Guizhi. Study on implementation of traceback algorithm in Viterbi decoders [J]. Journal of Electronics & Information Technology, 2007, 29(2): 278-282. (in Chinese)