



研究与开发

基于频偏矫正辅助的 Starlink 下行信号帧同步轻量化电路设计

韩煜¹, 周雪齐², 沈雷²

(1. 中国电子科技集团公司第36研究所电磁空间安全全国重点实验室, 浙江 嘉兴 314033;
2. 杭州电子科技大学通信工程学院, 浙江 杭州 310018)

摘 要: 传统正交频分复用 (orthogonal frequency division multiplexing, OFDM) 帧同步方法, 如基于前导序列、基于导频、基于深度学习的帧同步算法在面对 Starlink 高速率、高动态通信环境时局限于单路计算思路, 且具有较多的资源消耗和较高的时钟需求。而基于差分相移键控 (symmetric differential phase shift keying, SDPSK) 调制的多路并行帧同步算法虽然引入频偏矫正和本地前导序列相关使得性能上有所提升并降低了时钟需求, 但加大了计算量和硬件资源占用。针对以上情况, 基于 Starlink 公开前导结构, 提出了四路并行的基于频偏矫正辅助的 Starlink 下行信号帧同步轻量化电路设计。首先, 设计基于延时相关复用的轻量化粗帧同步和频偏估计结构使电路在频偏估计时不需要再次计算延时相关值。然后, 提出基于四路并行直接数字式频率合成器 (direct digital synthesizer, DDS) 的轻量化频偏矫正模块避免单路信号在多路结构下变频时需要缓存。最后, 设计基于符号相关和查找表复数乘法器的轻量化精帧同步结构减少了本地序列和信号相关的资源耗用, 在保证性能的前提下使得查找表 (look-up table, LUT)、查找表随机存取存储器 (look-up table random access memory, LUTRAM)、触发器 (flip-flop, FF)、块随机存取存储器 (block random access memory, BRAM) 资源分别节省了 7%、2%、5% 和 8%。对电路编写 verilog 代码进行现场可编程门阵列 (field programmable gate array, FPGA) 实现, 并通过 Xilinx 生产的 xczu47dr 芯片上板验证了其资源占用及性能表现。

关键词: Starlink; 帧同步; 轻量化

中图分类号: TN911.72

文献标志码: A

doi: 10.11959/j.issn.1000-0801.2026004

Design of Starlink downlink signal frame synchronization lightweight circuit based on frequency offset correction assistance

Han Yu¹, Zhou Xueqi², Shen Lei²

1. National Key Laboratory of Electromagnetic Sapce Security No.36 Research Institute of CETC, Jiaxing 314033, China
2. School of Communications Engineering, Hangzhou Dianzi University, Hangzhou 310018, China

Abstract: Traditional OFDM frame synchronization methods, such as preamble-sequence-based, pilot-based, and

收稿日期: 2025-04-30; 修回日期: 2025-11-09

通信作者: 沈雷, shenlei@hdu.edu.cn



deep learning-based frame synchronization algorithms, are limited to single-channel computing ideas, resource consumption, and high clock requirements in the face of Starlink's high-speed and high-dynamic communication environment. However, although the multi-channel parallel frame synchronization algorithm based on SDPSK modulation introduces frequency offset correction and local preamble sequence correlation to improve the performance and reduce the clock requirements, it increases the amount of computation and hardware resources. In view of the above situation, a four-way parallel design of Starlink downlink signal frame synchronization lightweight circuit based on frequency offset correction assistance was proposed. Firstly, a lightweight coarse frame synchronization and frequency offset estimation structure based on delay correlation multiplexing was designed, so that the circuit would not need to calculate the delay correlation value again during frequency offset estimation. Then, a lightweight frequency offset correction module based on four-channel parallel DDS was proposed to avoid the resource consumption of buffering when a single-channel signal was converted under the multi-channel structure. Finally, the lightweight fine frame synchronization structure based on symbolic correlation and look-up table complex multiplier was designed to reduce the resource consumption of local sequence and signal, and the resources of LUT, LUTRAM, FF and BRAM were saved by 7%, 2%, 5% and 8% respectively under the premise of ensuring performance. Based on the Starlink open leading structure, a lightweight circuit design of Starlink downlink signal frame synchronization based on frequency offset correction assistance was proposed, and verilog code was written for FPGA implementation, and its resource occupation and performance were verified by the xczu47dr chip-on-board produced by Xilinx.

Key words: Starlink, frame synchronization, lightweight

0 引言

近些年来,低轨道地球卫星通信系统因其传输时延低、功率衰减小、信号覆盖范围较大等优势引起各个国家和组织的关注^[1-2],其中规模最为庞大的当属Space X旗下的Starlink^[3],在其通信过程中,信号的帧同步检测是通信中重要的一环,只有准确识别到信号的开始,才能进行后续的数据处理。同时,接收机算法往往需要进行帧同步、载波同步、信道均衡、解扰解码等复杂的计算过程,这使得其在硬件实现时需要占用大量的资源,对各个部分尤其是需要大量相关运算的帧同步部分的电路设计提出了挑战。

传统的OFDM(orthogonal frequency division multiplexing,正交频分复用)帧同步方法存在2种思路,第1种基于接收序列的延时相关性。如基于接收序列延时相关特性的帧同步算法^[4-5]、基于反相前导序列延时相关特性的帧同步算法^[6]以及基于对称共轭前导序列延时相关特性的帧同步算法^[7-8]。这3种算法均利用前导序列中相同序列之

间的相关性进行帧同步检测。第一种思路的定时同步算法抗频偏能力强,但由于仅依赖被噪声干扰的接收信号进行帧同步,偏差较大^[9]。第2种思路则基于接收前导序列与本地前导序列的互相关性^[10],通过接收前导序列和本地前导序列互相关得到相关曲线,并判断曲线的峰值位置以实现帧同步。第2种思路引入未被噪声干扰的本地序列,使得其帧同步算法的偏差较小,但是接收信号往往带有频偏,使得其与本地序列的相关性降低,频偏较大时性能较差^[7]。文献[11]提出了一种基于零自相关(const amplitude zero auto-correlation, CAZAC)序列的算法,该算法将2段CAZAC序列组合作为训练符号,并且对后一段CAZAC序列进行加权处理,在低信噪比时依旧可以保持性能。文献[12]利用伪噪声序列加权恒包络导频,提出一种时频同步算法,在大频偏情况时算法仍有较好的估计结果。文献[13]提出了基于度量学习的轻量化神经网络定时同步方法,利用深度学习进行帧同步。文献[14]采用零相关区(zero-correlation zone, ZCZ)序列进行时频同步,利

用 ZCZ 序列在某个相关区内具有完美的自相关和互相关特性的特点, 提高了同步率。以上算法均需要进行大量的运算, 在电路设计上需要较多的乘法器, 会消耗大量的查找表 (look-up table, LUT)、块随机存取存储器 (block random access memory, BRAM) 资源。且以上算法在面对 Starlink 的高速率场景时, 其单路的计算结构需要高时钟与之对应, 从而造成较大的时钟需求。

文献[9]将 2 种思路进行了结合, 提出了基于对称差分相移键控 (symmetric differential phase shift keying, SDPSK) 调制的多路并行帧同步算法, 该算法首先利用前导序列的延时相关性进行粗帧同步, 得到前导序列的粗略起始点, 再利用前导序列进行频偏估计和频偏校正。其中, 粗帧同步是利用前导序列进行频偏估计和校正的前提。最后利用本地前导序列和频偏校正后信号前导序列的互相关性进行精帧同步, 由于频偏校正后的信号和本地前导序列相关性相较于频偏校正前更强, 达到了更好的性能。同时算法采用四路并行结构, 通过增大资源占用量的方式应对 Starlink 的高时钟需求, 在低时钟的情况下即可完成帧同步功能。然而, 该算法粗帧同步和频偏估计相互独立, 使其对应的现场可编程门阵列 (field programmable gate array, FPGA) 实现方案模块复用度低、资源占用多, 同时其精帧同步部分因为相关运算也需要大量的乘法器资源, 挤占了接收机后续算法的空间资源。

针对上述算法存在的问题, 本文提出了一种四路并行的、基于频偏校正辅助的 Starlink 下行信号帧同步轻量化电路设计。本电路设计分为 3 步, 首先, 提出了基于延时相关复用的轻量化粗帧同步和频偏估计结构, 区别于传统算法中将粗帧同步和频偏估计由于相关长度不同完全分开的做法, 本文提出的电路设计将粗帧同步中和频偏估计中均需要进行的相关运算的相关长度进行了统一, 从而实现了复用设计, 减少了资源消耗; 其次,

提出了基于四路并行直接数字合成器 (direct digital synthesizer, DDS) 的轻量化频偏校正结构, 使得电路在多路结构下不需要像传统单路 DDS 再对需要变频的信号进行缓存; 最后, 提出了基于符号相关和查找表复数乘法器的轻量化精帧同步结构, 通过将本地前导序列的信号进行符号量化并利用其符号的相关性进行精帧同步, 通过符号量化减少乘法器位宽从而降低资源消耗, 提出基于查找表的复数乘法器, 由于位宽较低, 基于查找表的复数乘法器相较传统的并行乘法器结构不需要进行逻辑运算, 消耗了更少的资源。此外, 本文提出的电路设计在精帧同步前引入了频偏校正辅助, 有效消除了频偏, 使得电路的帧同步性能也得到了保证。

对轻量化的帧同步电路进行设计并进行了 FPGA 实现, 最后结果显示本文提出的帧同步电路在保证性能的前提下节省了 LUT、查找表随机存取存储器 (look-up table random access memory, LUTRAM)、触发器 (flip-flop, FF)、BRAM 资源, 可为其他接收机算法留下更多富余资源。

1 基于频偏校正辅助的 Starlink 下行信号帧同步轻量化电路

根据现有公开文献^[9,15], Starlink 的前导序列由 1 个长度为 127 的最大长度线性反馈移位寄存器 (linear-feedback shift register, LFSR) 产生。本文采用的前导序列 S 如式 (1), 前导序列 S 结构如图 1 所示, 此处的 T 为经过调制的复数序列, 采样率为 240×10^6 时 T 长 128, 图 1 中 N 为 128×8 。后续电路设计基于此前导序列。

$$S = [\text{CP}, -T, T, T, T, T, T, T, T] \quad (1)$$

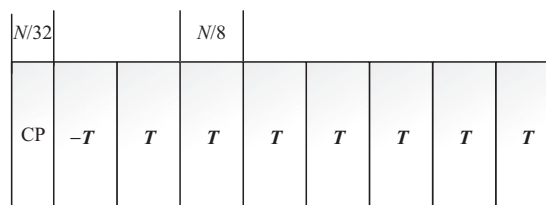


图1 前导序列 S 结构

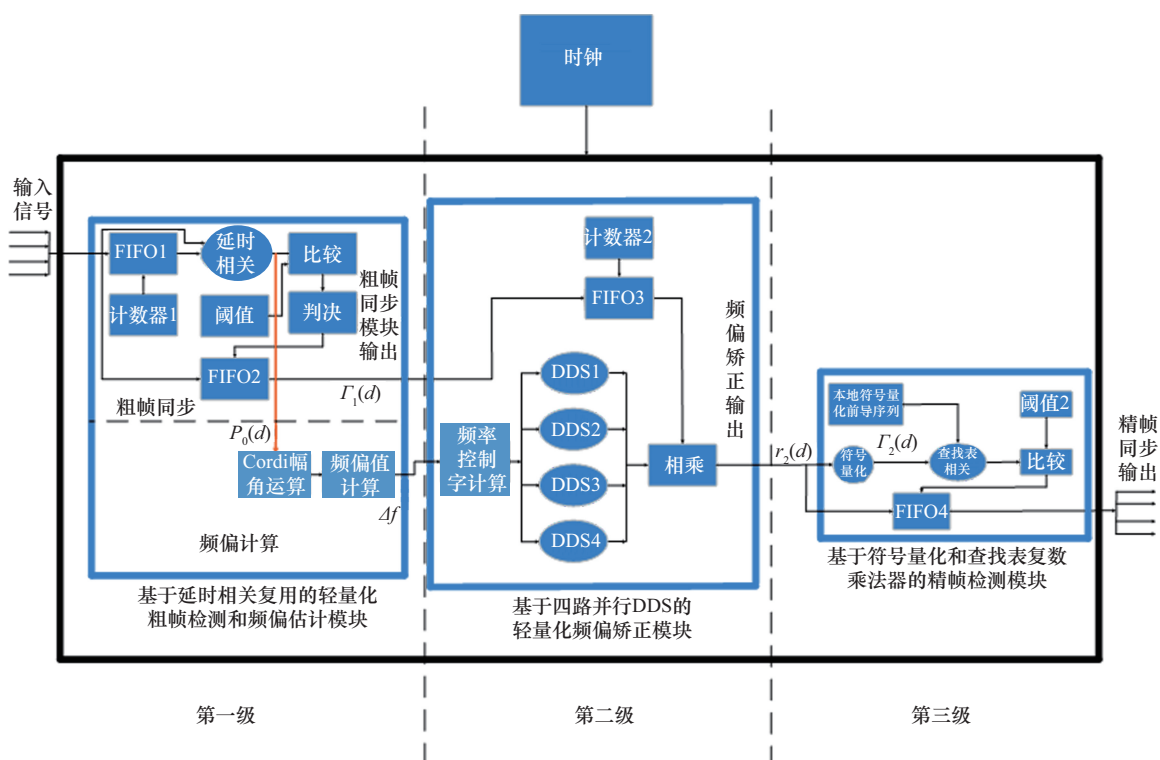


图2 帧同步轻量化电路设计

本文提出的帧同步轻量化电路设计如图2所示。

本电路设计按时间顺序由左到右分为3级，圆框部分为本文提出的轻量化设计部分。输入信号为四路并行信号，位宽共64 bit，每一路占16 bit。输入信号输入电路时，首先进入第一级基于延时相关复用的轻量化粗帧同步和频偏估计模块，本文将粗帧同步和频偏估计过程中都需要进行的延时相关的相关长度进行了统一，从而实现了延时相关的复用，将粗帧同步和频偏估计整合在同一模块中，如图2中第一级框图所示，延时相关模块计算得到的 $P_0(d)$ 会分别传输至粗帧同步（第一级虚线上半部分）过程中的比较模块和频偏估计（第一级虚线下半部分）过程中的坐标旋转数字计算方法（CORDIC）幅角运算模块，而传统算法的方案中的粗帧检测模块和频偏估计模块由于2个模块中的延时相关长度不一致而无法复用，这使得本文的电路设计相较于传统方案更加轻量

化，节省了更多的LUT、LUTRAM、FF、BRAM资源；粗帧同步后的信号 $r_1(d)$ 和频偏估计值 Δf 会输入电路的第二级基于四路并行DDS的轻量化频偏矫正模块，此结构避免了四路并行输入的情况下单路DDS需要先对信号进行缓存再进行变频的操作，从而节省了FF、BRAM资源， $r_1(d)$ 会和四路DDS产生的正、余弦波形相乘进行频偏矫正；频偏矫正后的信号 $r_2(d)$ 之后输入第三级基于符号相关和查找表复数乘法器的轻量化精帧同步模块中，本结构使用了符号量化，通过前导序列和本地序列的符号的相关性进行精帧同步，进而提出了基于查找表的复数乘法器来进行符号相关。通过符号进行相关时两路单路输入共需4 bit，而传统算法在进行相关时，使用原始数值计算，其位宽远大于4 bit，因此符号量化的做法可以节省更多资源。本文提出的基于查找表的复数运算乘法器，不需要进行逻辑运算，而是依靠查找表中存储好的结果输出，在输入数据位宽较低的时候相

较于传统的并行乘法器结构需要更少的逻辑资源,使得需要大量复数乘法器的轻量化精帧同步模块节省了更多的资源。同时,由于精帧同步前进行了频偏矫正,提高了本地前导序列和信号前导序列的符号的相关性,使得本电路的帧同步性能得到了保证。

结合文中给出的训练序列结构和电路设计图,本节将对本文提出的轻量化帧同步电路设计和其对应的原理逐级进行具体解释。

1.1 基于延时相关复用的轻量化粗帧同步和频偏估计模块

电路的第一级为基于延时相关复用的轻量化粗帧同步和频偏估计模块。电路首先对四路并行的输入信号进行粗帧同步,在进行粗帧同步的同时利用其过程中计算的延时相关值进行频偏估计。本文将2个模块中都需要进行的延时相关的长度进行了统一,实现了延时相关的复用,从而节省了更多资源。

粗帧同步利用输入信号中前导结构 S 序列的延时相关性进行。首先将输入信号分为两路,第一路输入先进先出(first in first out, FIFO) 1进行缓存并进行计时,第二路输入FIFO2中进行缓存, $N/8$ (N 为 S 的长度)个时钟之后将第一路输入FIFO1的信号输出,同时将此时FIFO1的输入和输出的信号一起传至延时相关模块,此时FIFO1的输入和输出的信号相距 $N/8$ 个采样点。

延时相关模块对两路相隔 $N/8$ 长度的输入信号执行相关计算,其相关值 $P_0(d)$ 的计算式见式(2),其中 $r(d)$ 为输入信号,输入信号为以 S 为前导的OFDM信号, d 为采样点坐标,*代表共轭。

$$P_0(d) = \left| \sum_{m=0}^{N/8-1} r(d+m) \cdot r^*\left(d+m+\frac{N}{8}\right) \right| \quad (2)$$

延时相关模块内部结构如图3所示。对于输入的两路信号,其中一路进行共轭操作,然后四路信号按图3中的方式对应相乘,之后进行累加缓存。对于信号中相邻两端序列的相关值,只需要减去缓存的第1个相乘值再加上刚计算的相乘值即新的相关值。

计算得到 $P_0(d)$ 后将 $P_0(d)$ 传至同级的频偏估计部分,同时将 $P_0(d)$ 与设定的阈值进行比较,若大于阈值则判断为检测到帧起始点,向FIFO2发出指令输出信号,此时FIFO2输出的信号始端即粗帧同步的结果,定义粗帧同步的结果为 D ,粗帧同步后的信号为 $r_1(d)$,见式(3),检测完毕后将信号输入频偏矫正模块。

$$r_1(d) = r(d+D) \quad (3)$$

对于合作方通信,通信双方会在较优的信噪比情况下进行通信,因此区别于传统算法利用重复序列的相关性进行帧同步需要归一化再和阈值比较判决的做法,本文提出的设计舍去了归一化

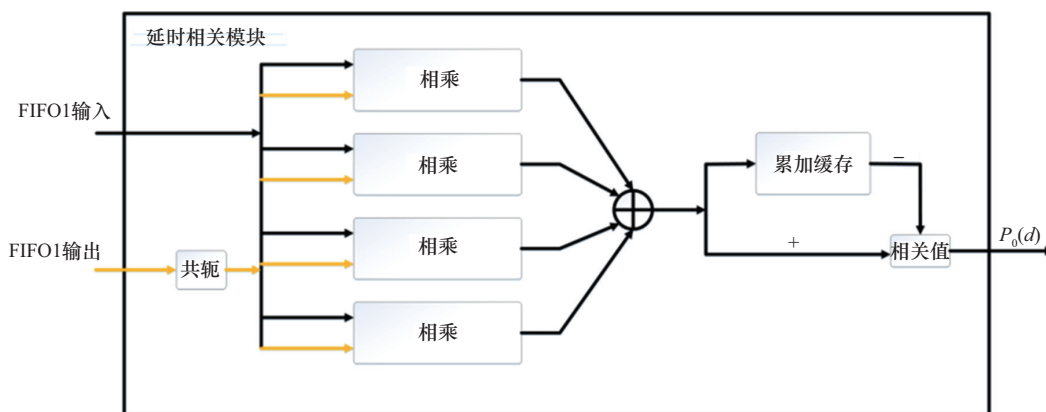


图3 延时相关模块内部结构



的步骤。同时,频偏估计部分在得到相关值 $P_0(d)$ 后可以根据式(4)、式(5)计算频偏值 Δf ,其中*代表共轭。定义 T_s 为采样周期,则有:

$$P_0(D) = e^{\left(-j2\pi\Delta f \frac{N}{8} T_s\right)} \sum_{m=0}^{N/8-1} \left| r(D+m) \cdot r^*\left(D+m+\frac{N}{8}\right) \right| \quad (4)$$

那么:

$$\Delta f = -\frac{\text{angle}(P_0(D))}{2\pi T_s N/8} \quad (5)$$

为实现式(4)代表的功能,频偏估计部分需要先求 $P_0(D)$ 的幅角,利用 Vivado 的 Cordic IP 可以实现这一功能, Cordic IP 为 Vivado 提供的用于 FPGA 中的预先设计好的电路功能模块,可进行幅角运算。对于式(5)中的 $-2\pi T_s N/8$ 可以提前计算其倒数作为参数,在得到 $P_0(D)$ 的幅角值后与其相乘得到 Δf 。基于上述分析, $P_0(d)$ 的幅角严格定义在 $[-\pi, \pi]$, 则可估计的最大频偏误差为 $\frac{1}{2T_s N/8}$, 即 937 500 Hz, 频偏估计部分计算频偏值所需要的相关值从粗帧同步处获得,而传统做法为粗帧同步完成后重新计算品频偏估计的相关值,本文的设计相较于传统做法而言节省了频偏估计中延时相关需要的信号缓存操作和乘加运算,节省了 LUT、LUTRAM、FF、BRAM 资源。得到粗帧同步后的信号和频偏值后,将其输入下一级。

1.2 基于四路并行 DDS 的轻量化频偏矫正模块

本文电路设计的第二级为基于四路并行 DDS 的轻量化频偏矫正模块(后简称为频偏矫正模块)。本文模块提出了四路并行 DDS 结构,使得电路在多路结构下不需要像传统单路 DDS 那样对需要变频的信号进行缓存。粗帧同步后的信号在到达频偏矫正模块后,由于四路 DDS 的正、余弦波形还未计算完毕,需要输入 FIFO3 进行短时间的缓存。粗帧同步的结果和频偏估

计的结果之间的时间差为固定值,因此可以用计时器 2,在固定时钟周期后输出缓存的信号,与此时输入的正、余弦波形对应相乘,其对应的计算式如下:

$$r_2(d) = r_1(d) * e^{-j2\pi\Delta f N T_s} \quad (6)$$

其中, $r_2(d)$ 为频偏矫正后的信号。

在频偏矫正模块得到 Δf 之后,需要根据 Δf 和系统时钟求出频率控制字 λ ,再利用 DDS 生成对应的正、余弦数据对粗帧同步后的信号进行频偏矫正。传统的 DDS 均为单路 DDS,即单路直接数字式频率合成器,其工作原理为利用频率控制字与累加寄存器输出的相位数据相加,相加的结果又反馈至累加寄存器的数据输入端,以使加法器在下一个时钟脉冲的作用下继续与频率控制字相加。这样,相位累加器在时钟作用下,不断对频率控制字进行线性相位累加。即在每一个时钟脉冲输入时,相位累加器便把频率控制字累加一次。相位累加器输出的数据为合成信号的相位,作为波形存储器的相位采样地址,即可把存储在波形存储器里的波形采样值经查表找出,完成相位到幅度的转换,从而输出所需频率的正、余弦波形。其中频率控制字需要通过系统时钟和所需频率计算得到。

单路 DDS 在四路并行的结构下需要先对信号进行缓存,每计算出一个正、余弦数据时输出一路信号进行变频相乘计算,这一过程破坏了电路的流水线式结构,且需要占用大量的缓存资源。为了对电路结构进行轻量化设计,本文提出了四路并行的 DDS 结构,具体工作原理如下。

以信号的 32 个采样点值为例进行说明,单路结构和四路并行下信号波形和余弦波形如图 4 所示,其中,图 4(a)表示单路结构下信号波形和余弦波形,图 4(b)表示四路并行下信号波形和余弦波形。

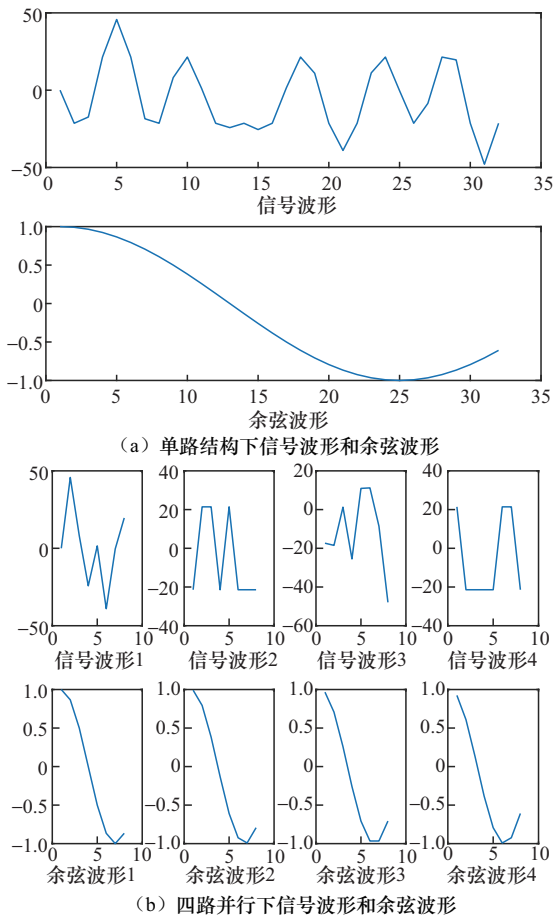


图4 单路结构和四路并行下信号波形和余弦波形

对于传统单路DDS来说,变频时信号值和余弦值逐点相乘即可,但在四路DDS的结构下,信号波形和余弦波形均需要变为4组,变频时信号波形1~4分别与余弦波形1~4逐点相乘。

余弦波形1~4在同一时刻的采样点实际上依次相隔1个采样点,单独的余弦波形中相邻时刻的采样点实际上相隔3个采样点。分组后的信号波形若要与分组后的余弦波形对应相乘,需要4个DDS模块,每个DDS模块需要设置新的频率控制字和初相以对应上述采样点顺序,DDS1~4初始相位和频率控制字见表1。

表1 DDS1~4初始相位和频率控制字

	DDS1	DDS2	DDS3	DDS4
初始相位	0	λ	2λ	3λ
频率控制字	4λ	4λ	4λ	4λ

通过表1中参数可知,四路DDS产生的正、余弦波形直接与对应的信号相乘,省去了信号缓存环节,保持了流水线设计,大大节省了FPGA的LUT和BRAM资源。

得到频偏矫正后的信号后,将其输入精帧同步模块。

1.3 基于符号量化和查找表复数乘法器的精帧同步模块

本电路设计的第三级为基于符号量化和查找表复数乘法器的精帧同步模块(后简称精帧同步模块)。该模块通过符号量化减少了计算位宽,并提出不需要像传统并行乘法器一样进行逻辑运算的查找表复数乘法器,在低位宽运算时更具优势,从而节省了硬件资源,配合前级的频偏矫正可以保持较优的性能。

精帧同步模块通过计算本地前导序列和信号中前导序列的相关值,并将相关值与阈值比较进行判决,从而得到精确的帧起始点。相关的计算式如式(7)。

$$P_{\text{exact}}(d) = \left| \sum_{m=1}^{\frac{N}{8}} r_2^*(d+m) \cdot \sum_{m=1}^{\frac{N}{8}} T(m) \right| \quad (7)$$

其中, $P_{\text{exact}}(d)$ 为本地前导序列和信号中前导序列的相关值。

精帧同步模块结构设计如图2第三级所示。假设采样率为480 Mbit/s,此时 $N/8$ 为256,若要实现流水线式设计,需要在一个时钟内计算256个信号采样点和256个本地前导序列点的相乘值,对应的FPGA结构需要256个16 bit×16 bit的乘法器,需要消耗巨大的资源。为了优化精帧同步模块的资源,本文提出了符号量化和查找表复数乘法器的电路结构,通过本地前导序列和信号中前导序列的符号的相关性来确定帧精确起始点。图2中第三级圆框部分为精帧同步模块的轻量化设计部分。

频偏矫正后的信号 $r_2(d)$ 在输入精帧同步模



块后分为两路，一路输入FIFO4进行缓存，另一路输入符号量化模块进行符号量化操作，符号量化操作可用式（8）表示。

$$r_3(d) = \begin{cases} 1, & r_2(d) \geq 0 \\ -1, & r_2(d) < 0 \end{cases} \quad (8)$$

$r_2(d)$ 通过量化后得到量化后的数据 $r_3(d)$ ，在FPGA中，符号量化操作可通过信号的最高位，即符号位和1 bit的1拼接实现。然后将 $r_3(d)$ 输入符号相关模块进行相关运算。量化后的数据仅有2 bit位宽，可减少后续运算的计算量。

为进一步降低资源消耗，本电路设计通过查找表的方式将相关计算过程中需要用到的乘法器进行了替换，具体实现原理如下。

由第1.3节可知，经过量化后的信号的实部和虚部只会出现4种组合：1, 1; 1, -1; -1, 1; -1, -1。本地前导序列可进行相同的量化操作，在Verilog中具体实施时可用0代表-1，使得两路输入位宽共4 bit，量化后的乘法组合仅有16种，因此符号相关模块用到的乘法器可以优化为查找表，基于查找表的复数乘法器输入和输出见表2。

表2 基于查找表的复数乘法器输入和输出

信号数据实部	信号数据虚部	本地前导序列实部	本地前导序列虚部	输出结果实部	输出结果虚部
1	1	1	1	0	2
1	1	1	-1	2	0
1	1	-1	1	-2	0
1	1	-1	-1	0	-2
1	-1	1	1	2	0
1	-1	1	-1	0	-2
1	-1	-1	1	0	2
1	-1	-1	-1	-2	0
-1	1	1	1	-2	0
-1	1	1	-1	0	2
-1	1	-1	1	0	-2
-1	1	-1	-1	2	0
-1	-1	1	1	0	-2
-1	-1	1	-1	-2	0
-1	-1	-1	1	2	0
-1	-1	-1	-1	0	2

基于查找表的复数乘法器结构如图5所示。其中，乘法器输入1为输入信号的实部和虚部符号量化后拼接的1、0或0、1，乘法器输入2为本地序列的实部和虚部符号量化后拼接的1、0或0、1，乘法器输出为乘法器输入1和乘法器输入2的相乘值。

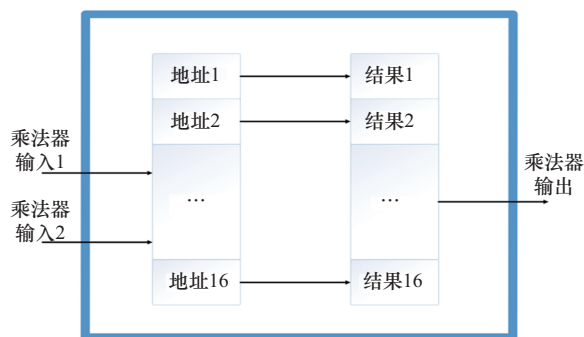


图5 基于查找表的复数乘法器结构

这样的优化方式可以在保证精度的前提下^[9]不再使用需要逻辑运算的并行乘法器，在低位宽时并行乘法器的逻辑运算相较于查找表乘法器需要消耗更多资源，并行乘法器在本文讨论中主要使用LUT资源，因此，该设计极大地降低了电路对LUT资源的消耗。

查找表相关模块在计算完两路输入的相乘值后将结果相加得到本地前导序列符号和输入信号的符号的相关值，四路结构下需要按照图3的方式对应相乘。之后将相关值和阈值进行比较，若相关值大于阈值则对FIFO4发出指令输出信号，此时输出信号的始端即精帧同步的结果。

本文电路设计提出了粗帧同步和频偏估计结合处理的设计结构，以及四路并行DDS的设计和实现，同时利用符号量化操作和基于查找表的复数乘法器进行精帧同步，通过以上轻量化设计，在保证性能的前提下极大地降低了本帧同步电路的资源消耗。

2 实验结果和分析

本部分首先以基于接收序列延时自相关特性^[4-5]、基于反相前导序列延时自相关特性^[6]、基

于对称共轭前导序列延时自相关特性的帧同步算法^[7-8]、基于SDPSK调制的多路并行帧同步算法^[9]为基准,通过MATLAB仿真对本电路设计对应的算法的性能进行比较,证明了本文提出的方案在性能上和传统最优算法持平,然后将传统算法中的最优算法和本文提出的轻量化帧同步电路设计进行了FPGA实现,通过编译得到消耗资源量,通过对比得出本文提出的电路设计在性能不损失的情况下消耗资源量更少的结论。

2.1 算法性能仿真和分析

在相同条件下,采用相同的训练序列长度和循环前缀长度。仿真了1 024个子载波、 $N/32$ 个循环前缀采样、归一化载波频偏为1.2的OFDM系统,采用2 000次蒙特卡洛实验评估在高斯白噪声信道中算法的同步性能,采用均方根误差(root mean square error, RMSE)作为性能指标来评价系统的性能。

算法性能对比如图6所示,显示了在相同条件下采用本文提出的方法和传统的算法得到的度量结果。算法性能对比见表3,给出了部分不同信噪比下RMSE的结果。

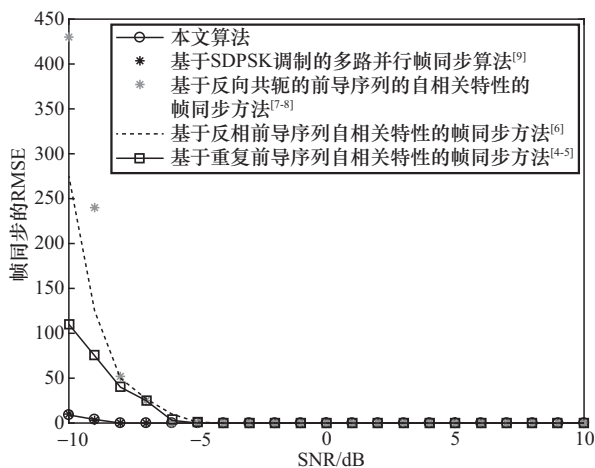


图6 算法性能对比

在高信噪比时各个算法性能差别较小,但在低信噪比(如-10 dB和-5 dB)时,在性能方案上,本文所提方法和基于SDPSK调制的多路并

行帧同步算法基本持平并优于其他算法。但由于本文算法将粗帧同步和频偏估计过程中都需要进行的延时相关的相关长度进行了统一,并降低了精帧同步中的乘法运算位宽,虽然减少了资源消耗,但造成进行频偏估计和精帧同步的数据量减少,使得算法相较于基于SDPSK调制的多路并行帧同步算法在更低信噪比的场景下可能存在鲁棒性不足的问题。

表3 算法性能对比

均方根误差 (RMSE)	信噪比 (SNR) /dB				
	-10	-5	0	5	10
本文算法	9.075	1.025	0.110	0.100	0.005
文献[9]	9.072	0.975	0.090	0.050	0.005
算法 文献[7-8]	437.233	1.325	0.150	0.110	0.080
文献[6]	274.120	1.225	0.150	0.105	0.40
文献[4-5]	107.335	1.100	0.122	0.100	0.008

2.2 FPGA实现和资源分析

本文基于Xilinx的xczu47dr芯片对提出的轻量化帧同步电路和基于SDPSK调制的多路并行帧同步算法进行了实现。本文的电路设计通过集成逻辑分析仪(integrated logic analyzer, ILA) IP核将编译好的电路的中间结果进行抓取。

粗帧同步的关键在于得到延时相关曲线,粗帧同步相关结果如图7所示,通过对此相关曲线进行阈值判决得到粗帧同步结果,图7中灰框内尖峰对应位置为信号起始点。

粗帧同步完成后为频偏估计和矫正的结果,频偏矫正前信号如图8所示,频偏矫正后信号如图9所示,相较图8,图9拥有更小的频偏。

频偏矫正后为精帧同步,精帧同步的关键在于得到本地量化前导序列和信号量化后的相关曲线后进行判决,精帧检测相关结果如图10所示,为本地前导序列的符号和信号前导序列的符号的相关曲线,灰框内尖峰所在位置为信号精确起始点所在位置,相较于图7的粗帧同步相关曲线,图10的相关峰更加锐利,帧同步效果更好。

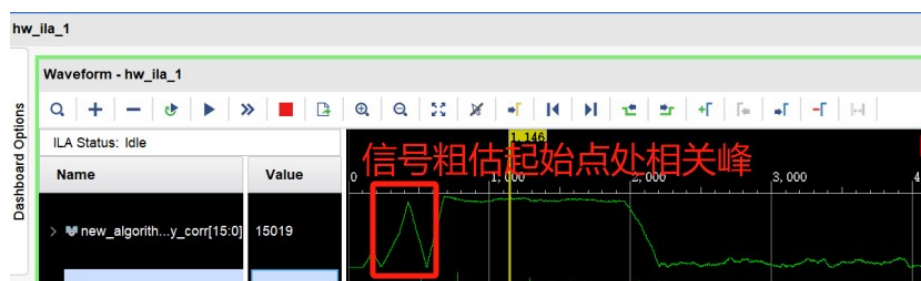


图7 粗帧同步相关结果



图8 频偏矫正前信号



图9 频偏矫正后信号

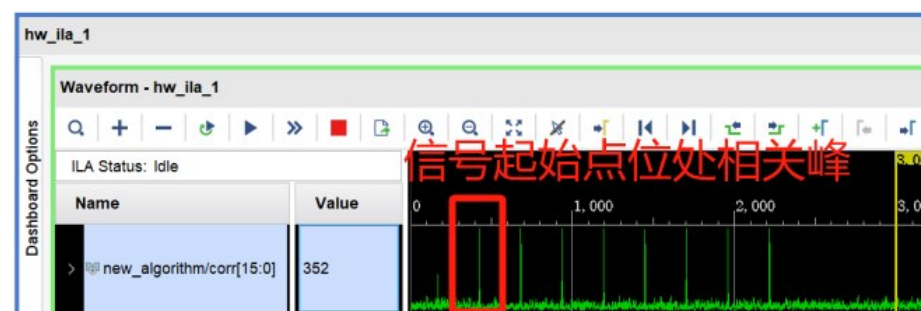


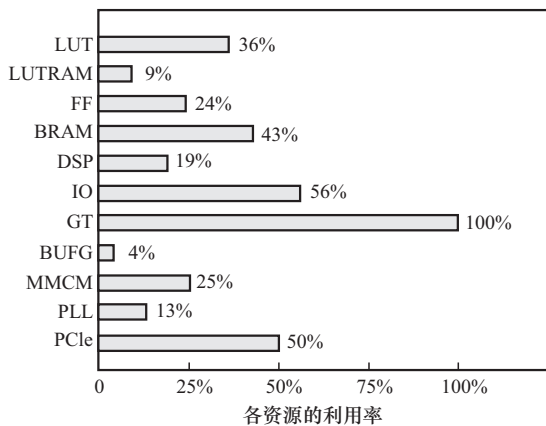
图10 精帧检测相关结果

在对本文设计的电路和性能一致的基于SDPSK调制的多路并行帧同步算法的硬件方案进行实现和编译后得到资源消耗图,电路资源消耗对比如图11所示,其中,图11(a)为本文电路的编译资源消耗,图11(b)为基于SDPSK调制的多路并行帧同步算法的硬件方案的编译资源消耗。

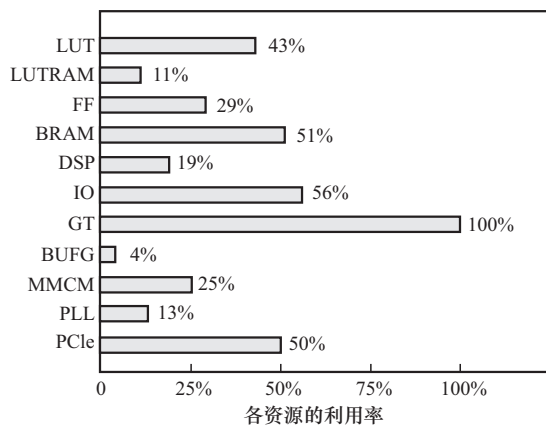
由图11(a)和图11(b)对比可知,本文提

出的轻量化帧同步电路设计在4种资源LUT、LUTRAM、FF、BRAM上分别从43%、11%、29%、51%降低至36%、9%、24%、43%,分别降低了7%、2%、5%和8%,实施后(Post-Implementation)为综合和布局布线完成之后的特定阶段,可代表电路实际占用资源。本文通过将粗帧检测模块和频偏估计模块的延时相关进行复

用,使得本文的电路设计相较于传统方案节省了更多的LUT、LUTRAM、FF、BRAM资源。本文设计的基于四路并行DDS的轻量化频偏估计模块,避免了四路并行输入的情况下单路DDS需要先对信号进行缓存再进行变频的操作,从而节省了FF、BRAM资源。在基于符号相关和查找表复数乘法器的轻量化精帧同步模块中,本文使用了符号量化和基于查找表的复数乘法器,进行相关时两路单路输入仅需4 bit且不需要逻辑运算,节省了更多LUT资源。



(a) 本文算法



(b) 基于SDPSK调制的多路并行帧同步算法

图11 电路资源消耗对比

3 结束语

本文提出了一种基于频偏矫正辅助的Starlink下行信号帧同步轻量化电路设计。该设计通过将

粗帧检测和频偏估计并联,增加了模块的复用度,减少了FPGA的LUT、LUTRAM、FF、BRAM资源消耗;通过设计四路DDS结构,省去了四路并行结构下的信号在变频时需要缓存的操作,减少了LUT和BRAM资源的消耗;通过量化相关的设计,避免了精帧检测相关运算对乘法器资源的消耗,从而减少了LUT资源的消耗。实验结果表明,本文提出的电路设计在具有较高的帧同步性能的同时降低了资源消耗。

参考文献:

- [1] Deng R Q, Di B Y, Zhang H L, et al. Ultra-dense LEO satellite constellations: how many LEO satellites do we need[J]. IEEE Transactions on Wireless Communications, 2021, 20(8): 4843-4857.
- [2] 张更新,王运峰,丁晓进,等. 卫星互联网若干关键技术研究[J]. 通信学报, 2021, 42(8): 1-14.
Zhang G X, Wang Y F, Ding X J, et al. Research on several key technologies of satellite Internet[J]. Journal on Communications, 2021, 42(8): 1-14.
- [3] Osoro O B, Oughton E J. A techno-economic framework for satellite networks applied to low earth orbit constellations: assessing Starlink, oneweb and kuiper[J]. IEEE Access, 2021, 9: 141611-141625.
- [4] Schmidl T M, Cox D C. Robust frequency and timing synchronization for OFDM[J]. IEEE Transactions on Communications, 1997, 45(12): 1613-1621.
- [5] 李子雄, 鄢秋荣, 汪奥, 等. OFDM系统物理层关键技术与FPGA实现[J]. 南昌大学学报(工科版), 2024, 46(2): 148-155.
Li Z X, Yan Q R, Wang A, et al. Research on key technologies of OFDM system physical layer and FPGA implementation[J]. Journal of Nanchang University (Engineering & Technology), 2024, 46(2): 148-155.
- [6] Minn H, Zeng M, Bhargava V K. On timing offset estimation for OFDM systems[J]. IEEE Communications Letters, 2000, 4(7): 242-244.
- [7] Park B, Cheon H, Kang C, et al. A novel timing estimation method for OFDM systems[J]. IEEE Communications Letters, 2003, 7(5): 239-241.
- [8] Guo J, Liu L W, Ling C, et al. A time frequency synchronization approach for OFDM short burst signals[C]//Proceedings of



- the 2024 3rd International Joint Conference on Information and Communication Engineering (JCICE). Piscataway: IEEE Press, 2024: 46-51.
- [9] Han Y, Jin L F. A synchronization algorithm and FPGA implementation scheme for LEO satellite communication[C]//Proceedings of the 2024 4th International Symposium on Computer Technology and Information Science (ISCTIS). Piscataway: IEEE Press, 2024: 804-808.
- [10] Kang Y, Kim S, Ahn D, et al. Timing estimation for OFDM systems by using a correlation sequence of preamble[J]. IEEE Transactions on Consumer Electronics, 2008, 54(4): 1600-1608.
- [11] 朱彦, 张会生, 骆艳卜. 基于CAZAC序列的OFDM时频同步算法[J]. 计算机仿真, 2009, 26(11): 130-133.
- Zhu Y, Zhang H S, Luo Y B. An OFDM timing and frequency synchronization algorithm based on CAZAC sequence[J]. Computer Simulation, 2009, 26(11): 130-133.
- [12] Ren G L, Chang Y L, Zhang H, et al. Synchronization method based on a new constant envelop preamble for OFDM systems[J]. IEEE Transactions on Broadcasting, 2005, 51(1): 139-143.
- [13] Li Y H, Minn H. Robust and consistent pilot designs for frequency offset estimation in MIMO OFDM systems[J]. IEEE Transactions on Communications, 2008, 56(10): 1737-1747.
- [14] Li Z, Tong J, Ma Y, et al. Timing and frequency synchronization using ZCZ sequences for multi-cell OFDM systems[C]//Proceedings of the 2017 3rd IEEE International Conference on Computer and Communications (ICCC). Piscataway: IEEE Press, 2017: 6-11.
- [15] Humphreys T E, Iannucci P A, Komodromos Z M, et al. Signal structure of the Starlink ku-band downlink[J]. IEEE Transactions on Aerospace and Electronic Systems, 2023, 59(5): 6016-6030.

[作者简介]



韩煜 (1978-), 男, 中国电子科技集团公司第36研究所电磁空间安全全国重点实验室副研究员, 主要研究方向为自适应信号处理、通信与语音信号处理。



周雪齐 (1999-), 男, 杭州电子科技大学硕士生, 主要研究方向为数字信号处理。



沈雷 (1979-), 男, 博士, 杭州电子科技大学通信工程学院教授、博士生导师, 主要研究方向为数字图像处理、模式识别。