

一种高速并行的 MSK 调制算法

魏刚冬, 李浩坤, 周良臣

(成都信息工程大学电子工程学院, 四川 成都 610225)

摘要: MSK 信号的相位连续与恒包络特性, 使得传统 MSK 调制算法难以打破系统时钟频率限制, 从而难以实现高速率、超宽带调制。针对该问题, 提出一种高速并行的 MSK 调制算法, 实现 MSK 并行调制, 使得 MSK 调制实现时, 在保留相位连续与恒包络特性的同时, 比特速率不再受限于系统时钟频率。该算法包含了并行差分编码、变速率并行延迟和并行 DDS 等, 不仅突破了系统时钟频率限制, 还支持比特速率连续可调。Vivado 环境中的 Verilog HDL 程序仿真验证了其有效性。

关键词: 比特率; MSK 调制; 高速调制; 并行信号调制

中图分类号: TN761

文献标志码: A

doi: 10.16836/j.cnki.jcuit.2023.04.004

0 引言

最小频移键控 (minimum shift keying, MSK) 具有恒包络、相位连续、频带利用率高和抗干扰性能好等优点, 在测控以及深空通信等领域均有广泛的应用^[1-4]。随着这些领域不断的发展, 对信息传输距离、速率和时延提出了更高的要求。信息速率由最初的几兆比特每秒, 上升到如今的几百兆甚至千兆比特每秒。同时为兼容不同的应用场景, 还需要满足速率可变的要求。

无线通信方面, 文献[5-11]均采用传统的串行 MSK 调制方式, 由于受到时钟频率的限制, 难以实现比特速率大于时钟频率的要求。光通信方面, 文献[12]和文献[13]采用专用的芯片和调制器实现了数百 Gbps 的光通信速率, 但结构复杂、成本过高, 且依然采用串行调制方式, 未突破系统时钟频率的限制。因此从 MSK 调制算法入手, 进行并行化算法设计, 以打破时钟频率限制。然而 MSK 不同于 QPSK、QAM 和 APSK 等数字调制, MSK 调制为保证符号间相位连续的限定条件, 相位是不断累加的, 累加的递归条件由前后两个码元决定^[14]。因此, 简单地把数据分发到多路的调制模式, 对 MSK 调制是不适用的。

为满足 MSK 相位连续和包络恒定等限定条件, 提出并行差分编码和变速率并行延迟等算法, 使得 MSK 在调制实现时比特速率不再受限于系统时钟频率。并行差分编码算法通过建立并行差分编码表, 实现一个时钟内完成多路并行差分编码并保持 MSK 的相位连续性。变速率并行延迟算法通过计算多路并行状态

下, 相关延迟参数的整数部分和余数部分, 整数部分采用 FIFO 结构延迟, 余数部分采用移位寄存的方式延迟, 从而保证 Q 路能正确地延迟 I 路半个符号周期。并行 DDS 算法和并行信号加权保证, MSK 基带信号各个并行支路对应的符号周期与加权信号能正确匹配, 从而实现 MSK 包络恒定的限定条件。

1 并行 MSK 调制

MSK 信号的数学表达式:

$$S(t) = \cos(\omega_c t + \frac{a_k \pi}{2T_s} t + \varphi_k) \quad kT_s \leq t < (k+1)T_s \quad (1)$$

式中: ω_c 为 MSK 信号的载波角频率, T_s 为输入码元 a_k 一个码元周期, φ_k 为相位常数。为保证 $t = kT_s$ 时相位连续, 相位常数 φ_k 需要满足:

$$\varphi_k = \begin{cases} \varphi_{k-1}, & a_k = a_{k-1} \\ \varphi_{k-1} \pm k\pi, & a_k \neq a_{k-1} \end{cases} \quad (2)$$

将 MSK 数学表达式正交展开为

$$\begin{aligned} S_{\text{MSK}(t)} &= \cos\varphi_k \cos\left(a_k \frac{\pi t}{2T_s}\right) \cos(\omega_c t) - \sin\varphi_k \sin\left(a_k \frac{\pi t}{2T_s}\right) \sin(\omega_c t) \\ &= \cos\varphi_k \cos\left(\frac{\pi t}{2T_s}\right) \cos(\omega_c t) - a_k \cos\varphi_k \sin\left(\frac{\pi t}{2T_s}\right) \sin(\omega_c t) \end{aligned} \quad (3)$$

同相分量 $I_k = \cos\varphi_k$ 和正交分量 $Q_k = a_k \cos\varphi_k$, $\cos(\pi t/2T_s)$ 和 $\sin(\pi t/2T_s)$ 分别为同相分量和正交分量的加权函数。MSK 调制框图如图 1 所示。

同相分量和正交分量可以通过差分编码得到, 保证了 MSK 的相位具有连续性。通过并行差分编码算法, 保证并行 MSK 调制的相位连续性。

时延 $\tau/2$ 使得 I、Q 两路符号错开半个符号周期,

以满足加权后 MSK 基带的恒包络特性。在并行结构中,由于调制速率的变化,使得时延 $\tau/2$ 不再是简单的串并转换能够解决的,因此提出了变速率并行延迟算法,以满足 I、Q 两路符号错开半个符号周期。

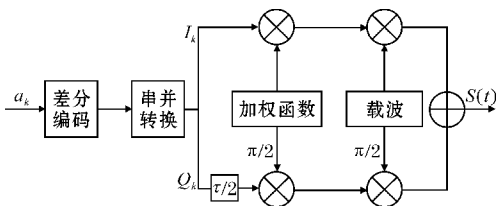


图1 MSK 正交调制框图

1.1 并行差分编码算法

差分编码是实现 MSK 相位连续的关键步骤之一。差分编码不仅仅与当前的状态有关,还与上一个状态有关,前后码元之间具有相关性。MSK 正交调制采用的差分编码为空号差分(0 变 1 不变或 -1 变 1 不变),其串行差分编码数学表达式为

$$d_n = d_{n-1} \odot a_n \quad (4)$$

其中, n 表示时间刻度, \odot 表示同或运算。差分编码的初始状态为 $d_0 = a_0$ 。差分编码运算为,当前时刻输入数据 a_n 与上一时刻差分码 d_{n-1} 之间的同或运算。对式(4)进行 N 路并行公式推导:令 $n = Nk + i, i = 0, 1, 2, \dots, N-1$, 则并行差分编码数学表达式为

$$d_{Nk+i} = d_{Nk+i-1} \odot a_{Nk+i} \quad (5)$$

其中 $d_{Nk-1} (i=0)$ 的取值为 $k-1$ 时刻,差分编码的最后一路输出,即 $d_{Nk-1} = d_{N(k-1)+(N-1)}$ 。若以式(5)进行并行差分编码,则在并行周期 k 内,只能完成第一路差分编码却无法完成剩余 $N-1$ 路差分编码。为解决冲突,采用枚举法,构建 N 路并行差分编码表。只需将当前输入和上一时刻最后一路差分值,作为查表地址,即可在一个周期内完成对输入信号的 N 路差分编码。

记并行差分编码的函数表达式为 $f(x, y)$, x 为 $d_{Nk-1} (i = N-1)$, y 表示当前时刻的 N 路输入数据即 $(a_{Nk}, a_{Nk+1}, \dots, a_{Nk+(N-1)})$ 。 $f(x, y)$ 为当前时刻差分编码的输出值。当 N 为 2 时,2 路并行差分编码表,如表 1 所示。针对不同的并行度 N ,只需替换对应的 N 路差分编码表即可实现。

表1 2路并行差分编码表

x	0				1			
y	00	01	10	11	00	01	10	11
$f(x, y)$	10	11	01	00	01	00	10	11

1.2 变速率并行延迟算法

传统 MSK 正交调制采用串并转换后,则 Q 路符号

自然会延迟 I 路符号一个码元周期,得到的符号周期正好对上加权函数的半周期,从而使得加权后的 Q 路基带信号延迟 I 路半个符号周期。若 I、Q 基带信号分别由 N 路信号并行构成,由于符号速率可变,使得半符号周期内所对应信号路数是变化的,因此传统的延迟方式是不适用的。针对上述问题,提出一种并行变速率延迟算法。

整个算法过程分为两步:延迟整数部分计算;延迟余数部分计算。描述如下:

设 MSK 的符号率为 f_{sps} , 采样率为 f_s , 则每一个符号的采样点数为 $M = f_s / f_{\text{sps}}$ 。若 MSK 的加权函数由 N 路数据并行而成,则延迟整数部分 D_{Int} 和余数部分 D_{Rem} 表达式如下:

$$D_{\text{Int}} = \left\lfloor \frac{M}{2} \right\rfloor // N \quad D_{\text{Rem}} = \left\lfloor \frac{M}{2} \right\rfloor \% N \quad (6)$$

[] 表示向下取整, // 表示取商, % 表示取余数。由于 M 表示 MSK 的一个符号在 f_s 下的采样点数,则半符号周期的采样点数为 $\lfloor M/2 \rfloor$, 向下取整保证为一个整数(向上取整或向下取整都可,此处定义向上取整)。 D_{Int} 表示 Q 路基带信号在当前符号率下需要延迟的周期数, D_{Rem} 表示 Q 路的 N 路基带信号中,需要向后延迟一拍的路数。

例如,当 $f_s = 3 \text{ GHz}$, $f_{\text{sps}} = 300 \text{ Msps}$, $N = 16$ 时,一个符号周期的采样点为 10, 则 D_{Int} 和 D_{Rem} 分别为 0 和 5。若 $f_{\text{sps}} = 50 \text{ Msps}$, 则 D_{Int} 和 D_{Rem} 分别为 1 和 14。根据上述变速率并行延迟算法,可以实现在并行结构下变速率延迟。

1.3 并行 DDS 算法

DDS 是数字通信常用的技术之一,通过控制频率或者相位控制字控制输出信号的频率和相位^[15]。DDS 的基本构成为相位累加器和波形存储器。相位累加器功能是对输入的频率控制字或相位控制字进行累加,输出的累加值作为波形存储器的地址值。波形存储器根据累加器给出的地址值进行查表,然后输出对应的数据。通常将一个周期的波形存放在只读存储器(read-only memory, ROM)中作为波形存储器,其 ROM 的宽度 N 决定了该 DDS 的分辨率,其频率分辨率为 $\Delta f = 1/2^N$, 相位分辨率为 $\Delta \varphi = 2\pi/2^N$ 。

频率和相位控制字的计算表达式:

$$M = \frac{f_{\text{out}} \times 2^N}{f_s}$$

$$N = \frac{\varphi_{\text{out}} \times 2^N}{2\pi} \quad (6)$$

f_{out} 和 φ_{out} 分别表示输出信号的频率和相位, N 表示波

形存储器中 ROM 的位宽, f 为系统时钟频率表示 DDS 输出波形的采样率。依据 DDS 基本原理, 在满足奈奎斯特采样定理($f_s \geq 2f_c$) 条件下, 最大能输出信号频率为 $f_s/2$, 在实际使用中最大输出频率为 $f_s/4$ 。由于通常时钟频率较低, 难以产生较高频率的信号, 因此需要对传统的 DDS 进行并行实现。中心频率为 f_c 的正弦信号表达式如下:

$$\cos(2\pi f_c \frac{n}{f_s}) \quad n=0,1,2,\dots,\infty \quad (7)$$

令 $n=Nk+i$, $N=16$ 为并行度, 表明 16 路并行, $k=0,1,2,\dots,\infty$ 为时间变量, $i=0,1,2,\dots,N-1$ 表示第 i 路。由此第 i 路的数学表达式为

$$\cos\left(2\pi f_c \frac{Nk+i}{f_s}\right) = \cos\left(2\pi f_c \frac{Nk}{f_s} + 2\pi f_c \frac{i}{f_s}\right), \quad k=0,1,2,\dots,\infty \quad (8)$$

由式(8)可知, 并行 16 路 DDS 具有相同的频率, 不同的初始相位, 即可实现并行 DDS。从而打破时钟频率限制, 输出并行的高频信号。

2 变速率并行 MSK 架构设计

通常并行路数越多, 则系统运行速度越快, 但同时会消耗更多的资源, 而硬件资源是有限的。因此 MSK 高速调制器的并行结构设计, 需要考虑并行路数和资源占用的情况。如图 1 所示, MSK 正交调制过程涉及到加权函数和载波的产生, 这两种信号需要满足奈奎斯特采样定理, 才能保证 MSK 调制器的正确性。文献[10]采用对加权后的信号进行 8 倍内插, 以满足加权后的信号与载波具有相同的采样率。为了系统更便于硬件实现以及结构更简单, 将加权函数和载波都采用同一采样率。变速率并行 MSK 架构如图 2 所示。

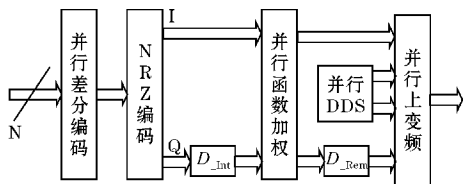


图2 两路并行 MSK 高速调制器设计框图

数据流路径为: 首先, N bit 数据输入进行 N 路差分编码, 输出 N 差分码; 然后, 进行不归零码(not return to zero, NRZ)编码(11 表示 0, 01 表示 1), 将 N 路 NRZ 码分为 I、Q 两路, N_0, N_2, \dots, N_{N-2} 为 I 路并行基带符号, N_1, N_3, \dots, N_{N-1} 为 Q 路并行基带符号; 其次, I 路基带符号和整数倍周期延迟后的 Q 路基带符号, 进行并行函数加权得到 MSK 基带信号; 最后, I 路基带信号与小数倍周期延迟后的 Q 路基带符号, 并行上变频得到 MSK 信号。

2.1 并行差分编码设计

根据并行差分编码算法, 枚举 N 路差分编码的所有组合, 建立 N 路并行差分编码表并存入 ROM 中。将 $K-1$ 时刻的最后一路差分码 x 和 K 时刻 N 路输入数据 $y=(y_0, y_1, \dots, y_{N-1})$ 拼接在一起作为表的地址, 地址为 $\text{Addr}=\{x, y\}$ 。并行差分编码具体实现结构如图 3 所示。

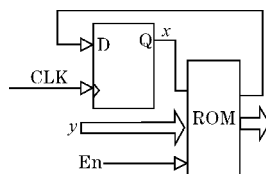


图3 并行差分编码模块

并行差分编码模块由一个 D 触发器和一个 ROM 表构成, D 触发器用于存放 x , 初始值为 0, ROM 是差分编码表。En 为输入数据 y 的使能信号, 当 En 为高电平时表示输入数据有效。ROM 表输出的同时更新 D 触发器的值, 用于下一时刻的并行差分编码。

通过枚举构建查找表的方法, 能够在时钟周期内完成 N 路差分编码, 较传统的差分编码吞吐量提升了 N 倍。ROM 表所占用的资源为 2^{N+1} 比特, 当 N 为 4、8 和 16 时对应组合数为别为 32、512 和 131072。

2.2 变速率并行延迟设计

变速率并行延迟算法分为整数延迟和余数延迟两大部分, 整数延迟表示延迟时钟周期的整数倍, 余数延迟表示延迟时钟周期的小数倍。如图 2 所示, 在具体实现过程中, 先对 Q 路符号进行整数倍周期延迟, 再对加权后的 N 路基带信号进行周期小数倍的延迟, 即完成整个 Q 路基带信号延迟半符号周期的过程。

整数延迟由 2 个同步 FIFO, 一个 D 触发器和一个计数比较器构成, 其结构如图 4 所示。

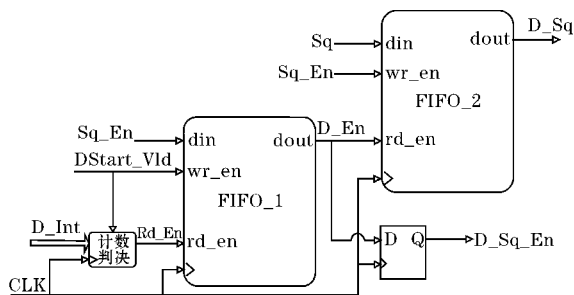


图4 整数倍延迟结构图

FIFO_1 用于缓存 Q 路符号使能, FIFO_2 用于缓存 Q 路符号。其工作原理为, 当 $D\text{Start_Vld}$ 为高时表示开始延迟, 将 Q 路符号使能 Sq_En 存入 FIFO_1, Q

路符号 S_q 存入 FIFO_2,同时计数判决器中的计数器开始工作。当计数器的值等于整数延迟值 D_Int 时,拉高 Rd_En 开始读取延迟后的使能信号 D_En ,将 D_En 作为 FIFO_2 的数据读取使能,从而得到整数倍延迟后的 Q 路符号 D_Sq 。由于 FIFO_2 数据读取过程中存在一个周期时延,因此将 D_En 通过 D 触发器缓存一个周期,得到延迟后的符号使能信号 D_Sq_En 。其时序图如图 5 所示, Q 路会固定多延迟 2 个周期,因此需要对 I 路符号打两拍,以保证延迟的正确性。

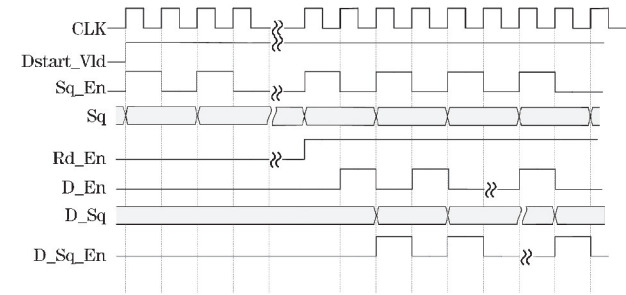


图 5 整数倍延时时序图

利用寄存器缓冲数据实现小数倍周期延迟,其工作原理如图 6 所示。

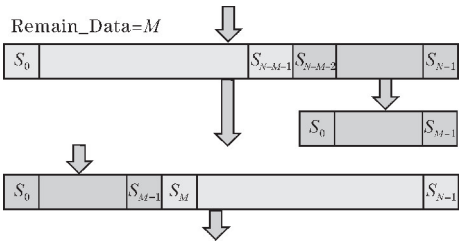


图 6 小数延迟

当 $D_Rem=M$ 时,将输入的 N 路数据的最后 M 路数据 S_{N-M-2} 到 S_{N-1} 送入缓冲寄存器。输出数据由输入数据的前 $N-M$ 路和缓冲寄存器组成,将缓存寄存器的 M 路数据作为输出数据的前 M 路数据,输入数据的前 $N-M$ 路数据作为输出数据的后 $N-M$ 数据。缓冲寄存器初始值为 0,从而实现了对于 N 路输入数据延迟 M 路。

2.3 并行 DDS 设计

并行 DDS 模块用于实现高频载波信号,16 路并行 DDS 结构如图 7 所示。其由 16 路相位累加器、查找表 (LUT) 和乘法器组成。乘法器通过移位和加法实现,以减少资源的消耗。图 7 中 $M=(f_c/f_s)\times 2^N$ 表示初始相位单元值,通过乘上对应支路上的倍数,得到各个支路的相位初始值。16 路相位累加器以 $N=16M$ 的步进,对各个支路相位初值进行累加,累加后的值作为各个支路的查表地址。累加器每一次累加都表示式(8)中 k 加 1,进而完成一次并行操作。

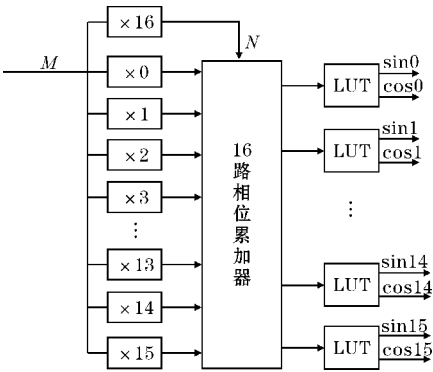


图 7 16 路并行 DDS 结构

与传统 DDS 相同,当 16 路相位累加器溢出,就表示完成一个周期的信号输出。例 $f_c=0.9\text{ GHz}$ 、 $f_s=3.6\text{ GHz}$ 代入式(8),每一路的频率为 56.25 MHz,初始相位为 $i2\pi/4$,则图 7 中 $\sin0, \sin1, \dots, \sin15$ 各支路取值如图 8 所示。

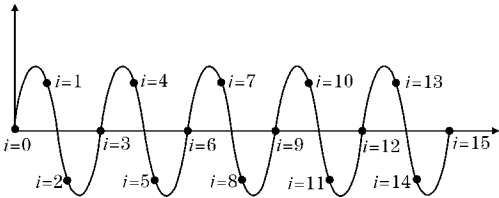


图 8 16 路并行 DDS 各支路取值

这样并行 DDS 结构解决了传统 DDS 受奈奎斯特采样定理和时钟频率限制的问题,在资源允许的条件下,可以实现任意采样率和中心频率信号的产生。

2.4 并行信号加权设计

加权函数信号生成采用与载波生成相同的并行 DDS 结构,不同的是,加权函数的 ROM 表需根据并行差分编码的并行度决定。例如,差分编码的并行度为 4,则 I 和 Q 的符号为两路并行,由于一个符号对应 $1/2$ 个加权周期,因此 2 个符号对应 1 个加权周期,此时 ROM 表需要存入 1 个加权周期。同理可知,当差分编码并行度为 6、8、10 等,ROM 表需要存入 3、4、5 个 $1/2$ 加权周期。

并行信号加权最大的难点为,如何实现 N 路并行的加权函数各支路正确地匹配上对应的符号。由于变速率,使得采样点数不是固定的,从而导致加权信号支路与符号支路的对应关系是变化的。

加权信号支路的符号匹配采用地址匹配的方式,其工作原理为,若某支路地址加上频率控制字大于查找表的最大地址,则表明该支路之后的支路对应下一个并行符号周期。在一个并行符号周期内,各支路同样根据地址确定该支路对应符号。例如 ROM 表存放了一个加权周期的 2048 个数据,当支路的地址小于

1024,对应第一个符号;当支路的地址大于 1024,对应第二个符号;当支路地址加上频率控制字大于 2048,则更新并行符号。其时序示意图如图 9 所示,BS0 表示第一个基带符号,BS1 表示第二个基带符号。

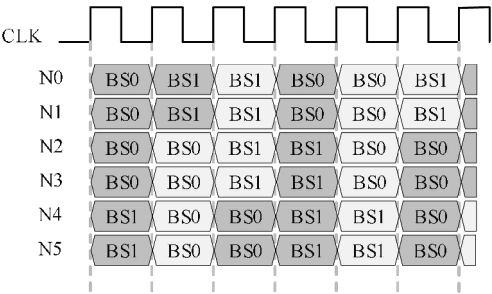


图 9 并行加权时序示意图

通过地址匹配法,正确的将加权信号支路与符号支路相匹配。基带符号在加权之前需要进入同步 FIFO 缓存,当某支路地址满足符号更新要求后,再从同步 FIFO 中读取。

3 仿真与结果分析

通过 vivado2017.4 平台和 Verilog HDL 语言,仿真验证高速并行 MSK 调制算法的正确性。由于根据并行数据难以判断各个模块功能的正确性,因此仿真通过高频时钟对并行数据合成,以验证 MSK 高速调制器的正确性。当系统时钟为 187.5 MHz、高频时钟为 3 GHz、数据速率为 600 Mbps 时,4 路并行的 MSK 调制器,仿真结果如图 10 所示。

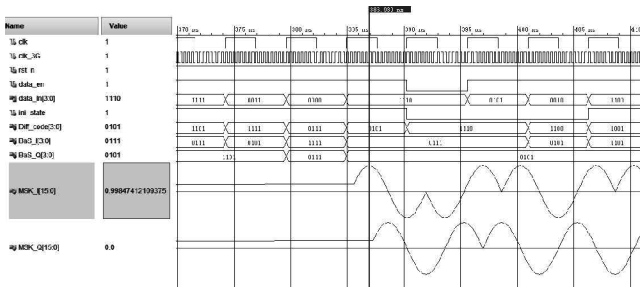


图 10 MSK 功能仿真图

图 10 中 clk 为系统时钟,clk_3G 为用于数据合成的高频时钟。msk_I、msk_Q 均为 16 路基带信号,在 clk_3G 高频下合成后的数据。如图 10 所示,输入信号 data_in 经过并行差分编码、NRZ 编码后以及 Q 路延迟整数周期,得到 BaS_I 和 BaS_Q 两路基带符号。基带符号加权和 Q 路延迟小数倍周期,得到基带信号 msk_I 和 msk_Q。将图 10 中的 msk_baseI、msk_baseQ 数据与 Matlab 数据(理论数据)进行比较,对比结果如图 11 所示。

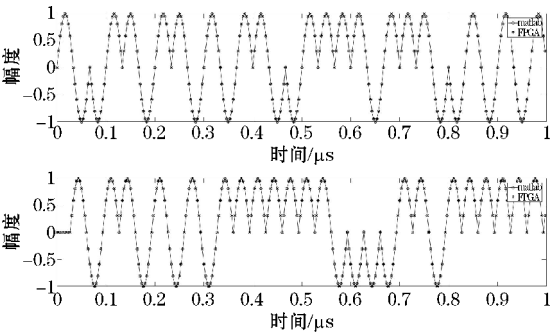


图 11 Matlab 和 FPGA 数据对比图

由对比结果可知,4 路并行 MSK 调制器得到的 MSK 基带波形与 Matlab 数据波形是重合的,验证了图 10 中 MSK 基带波形的正确性。

4 结束语

针对传统 MSK 调制算法受到 MSK 信号的相位连续与恒包络特性限制,难以打破时钟频率实现高速率调制的问题,提出一种高速并行的 MSK 调制算法。该算法由并行差分编码、变速率并行延迟和并行 DDS 等算法构成。并且根据该算法给出了具体的并行实现架构,以 FPGA 为验证平台,进行该算法正确性的验证。验证结果表明,该算法能打破硬件平台的时钟频率限制,最高比特率能达到 N 倍时钟频率(N 为并行度),使得比特速率大于时钟频率,并且支持 $(0 \sim N)$ 倍时钟频率之间的任意调制频率。该算法对于实现调制速率连续可调的超高速率、超宽带的 MSK 调制具有积极意义。

参考文献:

[1] ZANG G Z, HUANG B, CHEN L H. Performance analysis of the MSK modulation system over wireless fading channels[C]. 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC). IEEE, 2019: 1803–1806.

[2] LIU S, MA T, GANG Q, et al. Bionic communication by dolphin whistle with continuous-phase based on MSK modulation[C]. 2016 IEEE International Conference on Signal Processing, Communications and Computing(ICSPCC). IEEE, 2016: 1–5.

[3] CHEN Z, ZHAI R, Li D, et al. Performance Evaluation of a Tactical Data-Link System Based on MSK and 16QAM[J]. IEEE Access, 2021, 9: 84316–84326.

- [4] 闫涛,张大鹏,陶祁.采用延时积分的MSK软判决解调方法[J].电讯技术,2015,55(3):275-278.
- [5] PTAPKIR R, SINGH S B, Thune N N. Analysis and implementation of minimum shift keying (MSK) modulation on FPGA platform[C]. 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT). IEEE, 2016:349-354.
- [6] XIE W B, DOU H. Design and verification of MSK based on FPGA[C]. Proceedings of 2013 2nd International Conference on Measurement, Information and Control. IEEE, 2013:486-489.
- [7] MAGHAWRY A, ELDIWANY E. FPGA-based MSK DS-SS modulator for digital satellite communications[C]. 2009 National Radio Science Conference. IEEE, 2009:1-9.
- [8] 孙晔,尹立言,向新. MSK 信号数字化调制解调的工程实现方法[J]. 计算机应用与软件, 2019, 36(4):130-134.
- [9] 朱爱红,杨伟军,程旗,等. MSK 调制直接射频合成算法的设计与 FPGA 实现[J]. 电子世界, 2021(18):110-114.
- [10] 江海,蒋宇中,徐双全,等. 基于FPGA的MSK调制器设计与实现[J]. 舰船电子工程, 2012, 32(3):53-55.
- [11] 刘志武,刘志凌,江传民,等. 基于FPGA的IFF-MSK调制系统设计[J]. 航天电子对抗, 2019, 35(6):35-37.
- [12] SARKAR D, METYA S K. Performance investigation of 110-Gbps optical MSK system using a Quad Mach-Zehnder IQ modulator[J]. Optical and Quantum Electronics, 2018, 50(1):1-9.
- [13] CHEN Z, DONG S, CHEN Z, et al. Generation of high data rate MSK-modulated 180-GHz signals[J]. IEEE Microwave and Wireless Components Letters, 2019, 29(11):757-760.
- [14] 李然,王建新. 高速并行调制器的设计与实现[J]. 电子设计工程, 2017, 25(13):99-103.
- [15] DONG C C, KIM, YE E. High-Resolution Digital Beamforming Receiver Using DDS-PLL Signal Generator for 5G Mobile Communication[J]. Ieee Transactions on Antennas and Propagation, 2022, 70(2):1428-1439.

A High-speed Parallel MSK Modulation Algorithm

WEI Gangdong, LI Haokun, ZHOU Liangchen

(College of Electronic Engineering, Chengdu University of Information Technology, Chengdu 610225, China)

Abstract: For MSK modulation with phase continuity limitation, a high-speed parallel modulation algorithm is proposed, which realizes MSK parallel modulation, so that the bit rate of MSK is no longer limited to the system clock frequency when the modulation is implemented. The algorithm includes parallel differential coding, variable rate parallel delay, and parallel DDS, which not only breaks through the system clock frequency limit, but also supports continuously adjustable bit rates. Simulations of the Verilog HDL program in the Vivado environment confirmed its effectiveness.

Keywords: Bit rate; MSK modulation; high speed modulation; parallel signal modulation