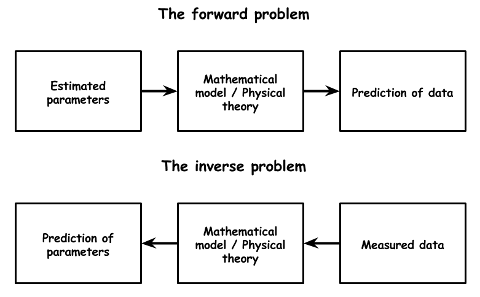
前言

实现是 forward problem，设计是 inverse problem



[What are inverse problems?](https://link.zhihu.com/?target=https%3A//cmontalto.wordpress.com/2013/03/08/what-are-inverse-problems/)

直接数字频率合成 (Direct Digital Synthesis, DDS) 技术是由美国人 J.Tiemcy 首先提出的，它是一种以数字信号处理理论为基础，从相位概念出发直接合成所需波形的一种新的全数字技术的频率合成方法。 DDS 主要出现在数字混频系统中。 在数字混频中， 通过 DDS产生正交的本地振荡信号即正、 余弦信号与输入信号相乘实现频谱搬移， 如通信系统中的调制、解调。目前，通过 FPGA 实现 DDS 有三种途径： 基于 IIR 滤波器的实现方法、 基于查找表 LUT 的实现方法以及基于 CORDIC 算法的实现方法。其中采用 LUT 的方法较为通用， 在 FPGA 设计中是较为主流的实现途径， 这缘于 FPGA 芯片中都会有丰富的 LUT 资源。

1 基于 IIR 滤波器的 DDS

利用 IIR 滤波器产生正弦波如图 1.1 所示。 图中 IIR 滤波器是一个全极点滤波器。

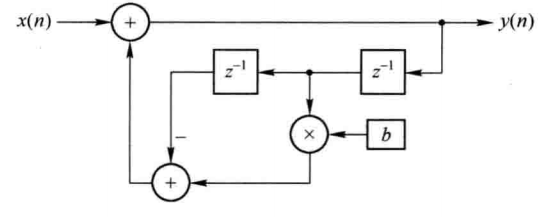


图1.1 基于 IIR 滤波器的正弦信号发生器

该滤波器输出与输入之间的关系可表示为：



在 z 域可表示为:



从而其传递函数可表示为:



式中， *p1* 和 *p2*是该滤波器的极点，且满足b= *p1+p2*， *p1p2*=1 通过计算可知



考虑到 B 是实数， 那么*p1* 和 *p2*即为复共轭，因此，可重新表示为



显然，两个极点模值均为1，且必然落在单位圆上。从频率角度描述极点即为：



对比式 (1.5) 和 (1.6) 可知



其中*f*s为采样频率。

由方程(1.7)可得滤波器的输出频率为：



从方程(9)可以看出适当调整参数b的值就可以得到不同频率的正弦曲线。例如当采样频率*f*s =100MHz时，b=时，输出频率*f*=*f*s/8=l2.5MHz。

假定输出频率*f*=*f*s/8 ，带入式（1.7）中可知对应的，从而计算得出b=。此时图1.1对应的IIR滤波器其频谱如图1.2所示，冲激响应如图1.3所示。

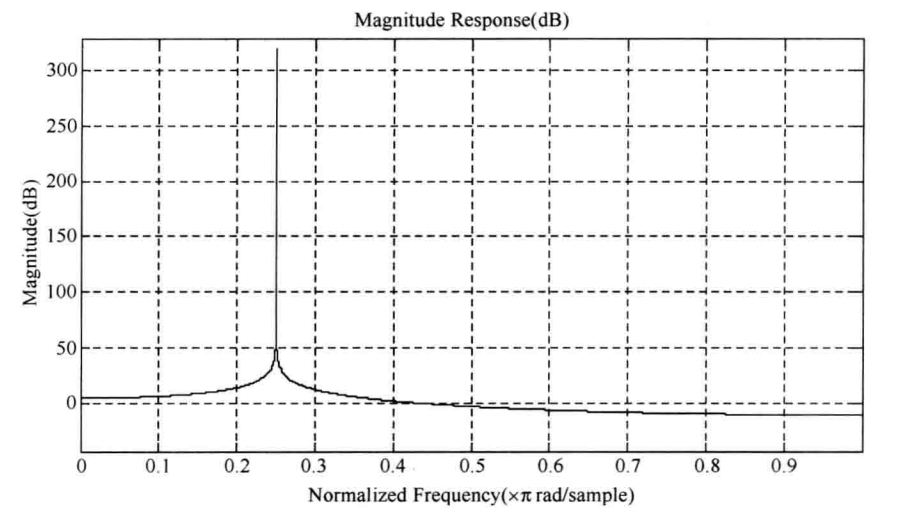


图1.2 b=时IIR滤波器幅度频谱

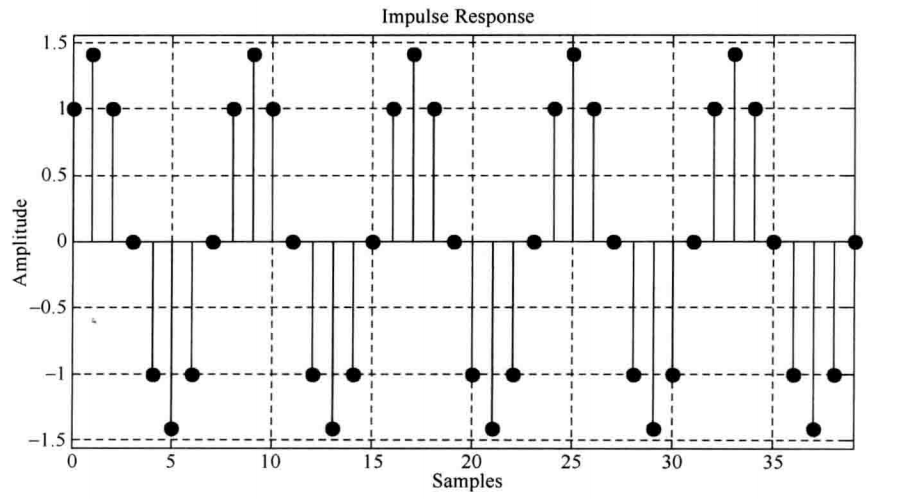


图1.3 b=时IIR滤波器冲激响应

图 1.3 中， 根据正弦信号频谱特性可知输出信号的频率为 0.25 *f*s/2= *f*s/8，从图 5.3 中可清晰地看出，此时输出信号即为一正弦波。

**利用SysGen进行仿真**

在SysGen中搭建如图1.4所示模型，并进行仿真，SysGen的使用教程可以参考：https://blog.csdn.net/Pieces\_thinking/article/details/83656686

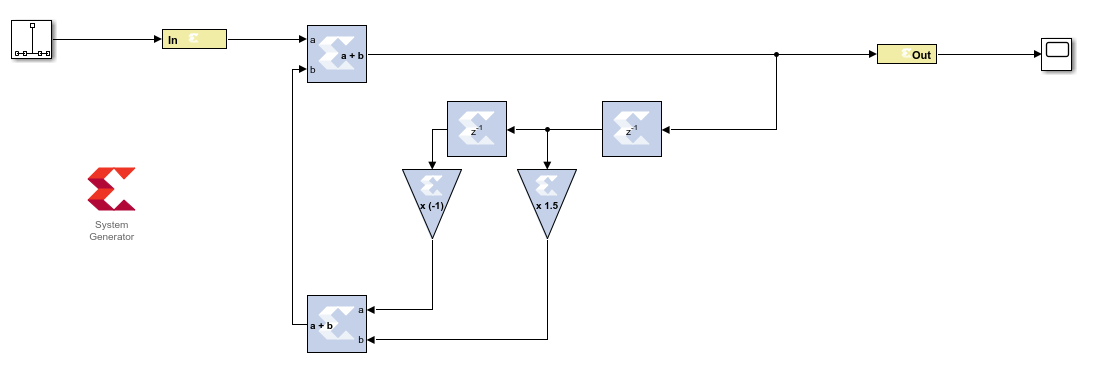


图1.4 基于SysGen系统搭建及仿真

输入信号是冲激信号，整个电路的位宽采用32位，小数位宽为30位，IIR数字滤波器系数b=≈l.5，系统采样频率*f*s=100MHz。

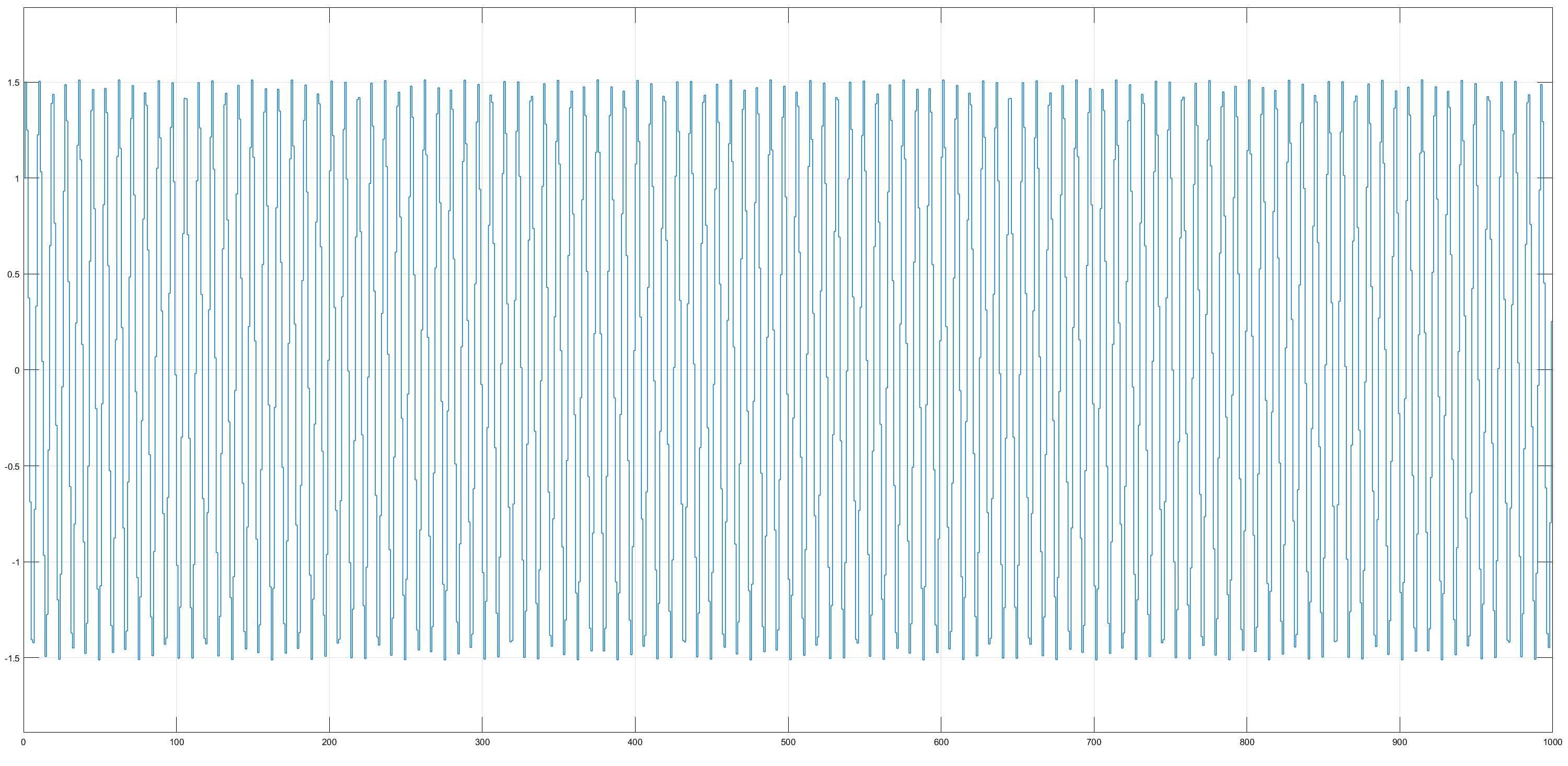
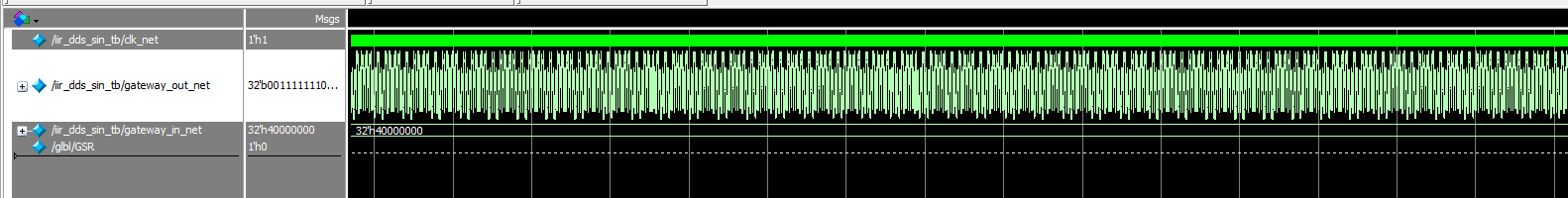


图1.5 正弦波成电路仿真

运行系统并观察其仿真波形验证结果如图1.5所示。山于极点位置处于单位吲上所以系统开始时处于振荡状态，但随后过程中不断叠加形成稳定频率的正弦信号。

整个设计使用system Gencrnor生成Vivado工程，并进行综合仿真布局布线工作。图1.6为仿真波形，仿真波形不是特别完美，不知道哪里出错了，可能是字长效应引起的。

图1.6 Vivado仿真波形

2 基于 LUT 的 DDS

2.1 常规型基于 LUT 的 DDS

一个典型的基于 LUT 的 DDS 系统由相位累加器和波形存储器两部分构成，如图 2.1 所示。图中相位累加器的位宽为 ***n*** bit，步进值为 ***μ***，LUT 的深度 ***N***为2n，宽度为 ***L*** bit。

LUT 中依相位顺序存储一个周期的波形数据。显然，此时相位累加器的输出位宽与 LUT 的地址位宽是一致的。 其中，***μ***又被称为频率控制字， 它决定了输出正弦波的中心频率， N 决定了相位分辨率， 如式 (2.1) 所示。



假定相位累加器位宽为 4bit, 那么 LUT 的深度N=16，此时其内部所存数据如表 2.1 所不。 这种对应关系完整地体现在图 2.2 中

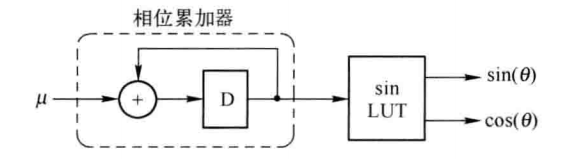
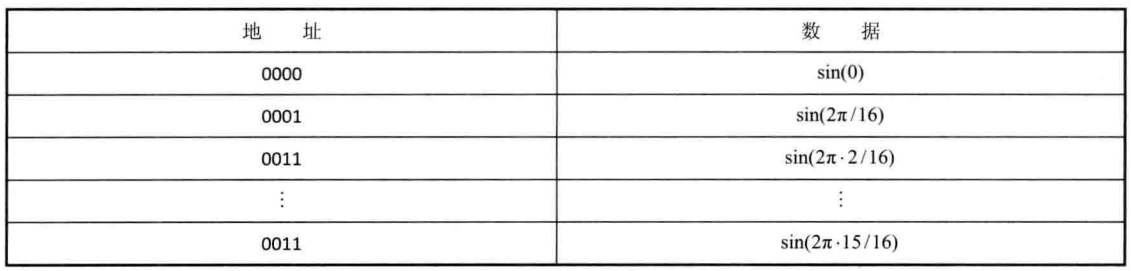


图2.1 基于LUT 的 DDS 硬件架构

表2.1 相位累加器位宽为 4bit 时 LUT 存储的数据



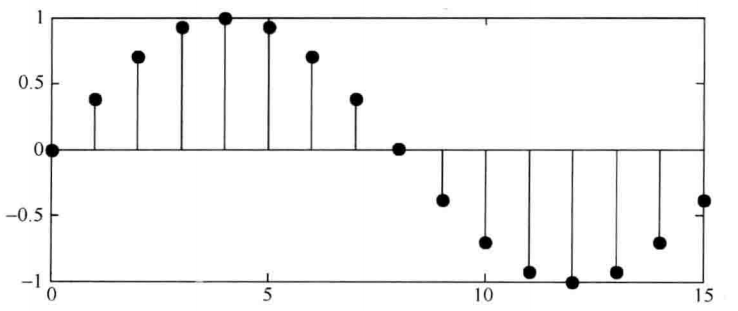


图2.2 相位累加器位宽为 4bit 时 LUT 地址与存储数据的对应关系

利用 Matlab 可生成所需的存储数据，如式 (2.2) 所示。



相位累加器的位宽决定了频率控制字***μ***的分辨率，在图1.1中，相位累加器的位宽和LUT地址位宽一致，因此要求相位累加器输入为整数，故频率控制字的分辨率为Δ***μ=***1。Δ***μ***决定了DDS输出频率的分辨率Δ*f*，如式（2.3）所示。频率控制字***μ、***LUT深度***N、***采样频率*fs*和输出频率*fd*之间的关系如式(2.4）和式(2.5)所示。







根据采样定理可知，输出频率的最大值为 *fs*/2 , 这意味着***μ***的最大值为 N/2 。 若***μ***= 3N/4，根据式 (2.4) 可知*fd*=3 *fs*/4，但实际输出频率应为 *fd*=3 *fs*/4- *fs* =- *fs*/4，这是因为时域的离散对应频域的周期，这个周期即为釆样频率。

改变***μ***即可改变输出频率，如图 2.3 所示。图中，左半部分对应的频率控制字小于右半部分对应的频率控制字，从而导致右半部分相位累加器输出波形较为“陡峭”（斜率大），LUT 输出频率大。 这是因为***μ***的增大，使得系统可以在较短时间内读完 LUT 中的所有数据(一个周期的波形）。

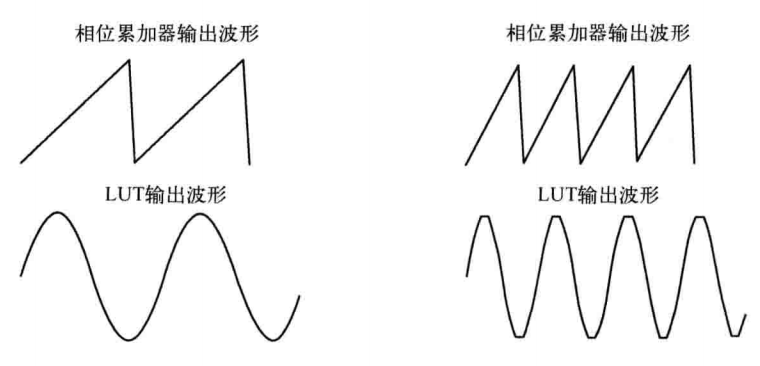
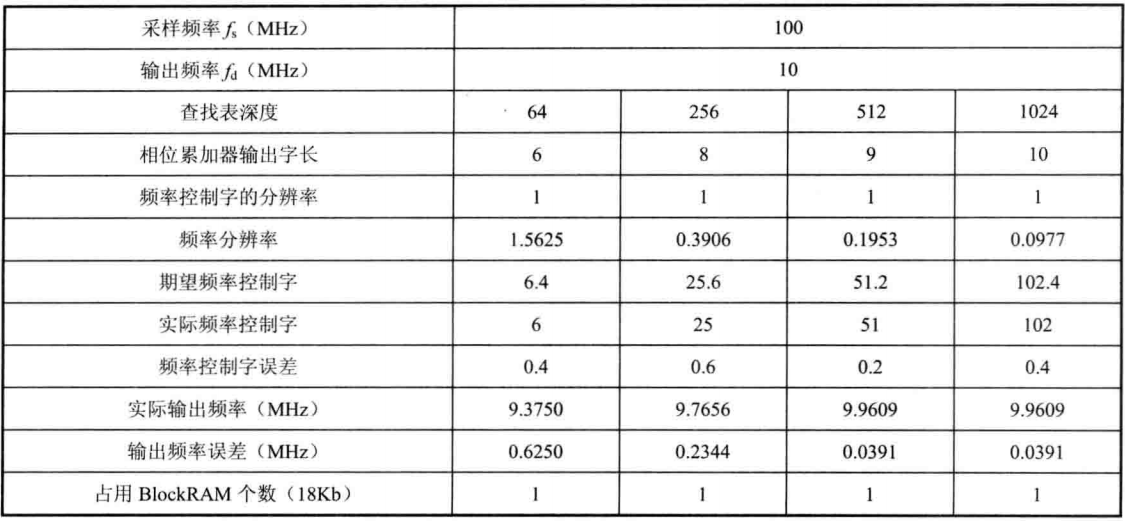


图2.3 频率控制字对相位累加器和输出波形的影响

以采样频率*fs* = 100MHz 期望输出频率*fd*= 10MHz 为例，设置不同的 LUT 深度，相应的参数如表 2.2 所示。从表中不难看出，相位累加器的输出字长 n 是决定性因素，随着它的增大， LUT 深度也增大， 频率分辨率进一步提高，输出频率精度也有所改善， 但并不是 n 越大，输出频率精度越高， 如表中n分别为 9 和 10，输出频率均为 9.9609MHz。这表明提高相位累加器的输出字长并不一定能改善输出频率精度。

表2.2 LUT 深度为 2 的整数次幂时各参数之间的关系



相位累加器输出字长与实际输出频率之间的关系如图 2.4 所示。

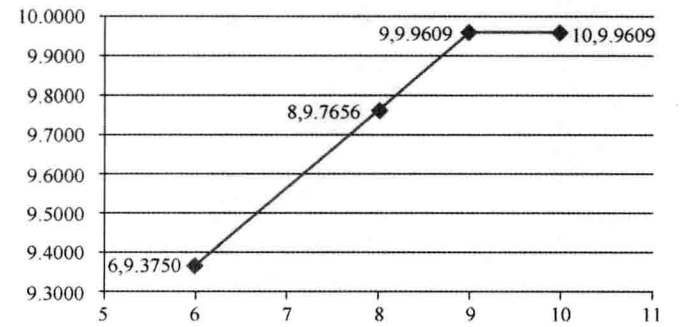


图2.4 相位累加器输出字长与实际输出频率之间的关系

由式 (2.4) 可知，实际输出频率总可表示为



式中***μ***r为实际频率控制字，可表示为



式中 floor 表示向下取整。 若***μ***本身就是整数，则可使得*fdr=fd*，如表 2.3 所示的案例。LUT深度为 1024 时输出正弦波的幅度谱如图 5.8 所示，中心频率误差为10-9.954 = 0.046MHz，SFDR ( Spurious Free Dynamic range ，无杂散动态范围）约为66.844dBm；LUT 深度为400时输出正弦波的幅度谱如图 5.9 所示，中心频率误差为10.02-10 = 0.02MHz，SFDR 约为 48.507dBm。对比之，可发现 LUT 深度为 400 时输出正弦波的中心频率更为精确。

表2.3 LUT 深度不是 2 的整数次幂时各参数之间的关系



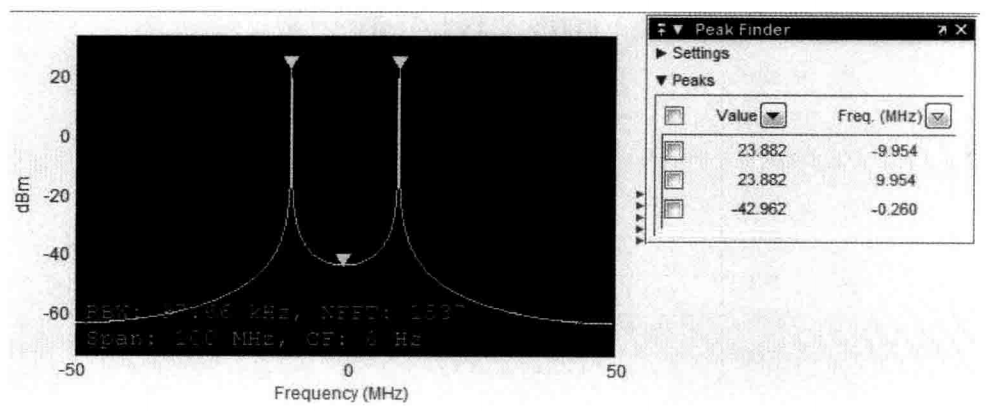


图2.5 LUT 深度为 1024 时输出正弦波的幅度谱

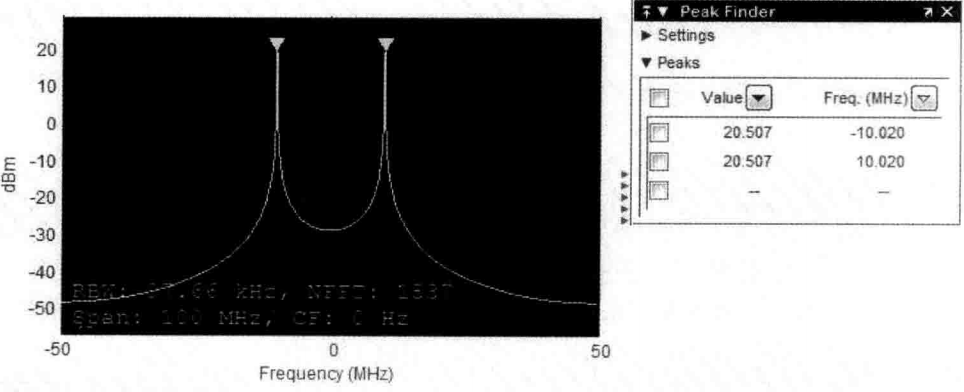


图2.6 LUT 深度为 400 时输出正弦波的幅度谱

但是，当 LUT 深度不是 2 的整数次幂时，相位累加器的设计将变得较为复杂。以 LUT 深度等于 400 为例，相应的相位累加器如 SysGen 模型 2.1 所示。其复杂性体现在需要对图中加法器的输出进行判定，当输出值大于等于模值时，相位累加器的输出为加法器输出值与模值之差。当 LUT 的深度为 2 的整数次幂时则不需要此判定，这是由二进制计数的特点决定的。以 4bit 二进制计数为例，如图 2.7 所示，同一位置的两个数相差为模值 16，且这两个数低 4 位对应的数值是相等的。鉴于此， 这里只讨论 LUT 深度为 2 的整数次幂的情形。

SysGen 模型 2.1 LUT 深度为 400 时相位累加器的硬件架构



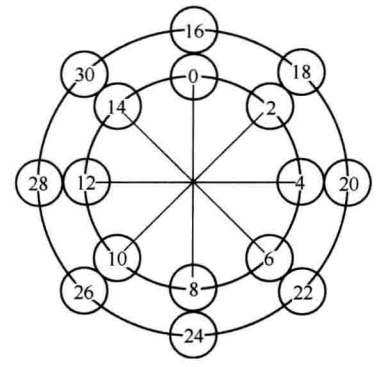


图2.6 二进制计数的特性

表 2.2 中显示，由于相位累加器执行的是整数加法，这就要求输入的频率控制字必须为整数，当期望频率控制字为小数时，需要将其取整，从而引入了误差。鉴于此，可对相位累加器进行改进，使其可以接受小数输入。此时相位累加器的结构如图 2.7 所示。相位累加器的字长为n+b，其中整数部分字长为n，小数部分字长为b。由于LUT地址必须为整数，所以需要对相位累加器的输出进行量化取整，从而形成如图 2.8 所示的 DDS 硬件架构。与图 2.1 相比，多了一个相位量化模块，其功能是取相位累加器输出的整数部分。

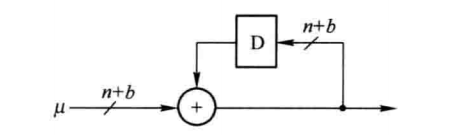


图2.7 带有小数位的相位累加器

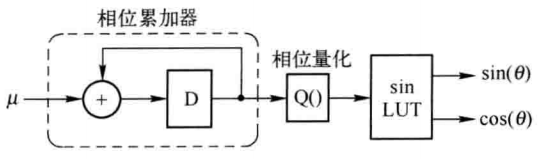


图2.8 带有小数位的相位累加器构成的 DDS 硬件架构

仍以采样频率*fs* = 100MHz、期望输出频率*fd*= 10MHz 为例，当n固定为6时，随着b的改变各参数之间的变化关系如表2.4所示，此时实际输出频率与b的关系曲线如图2.9所示；当b固定为6时，随着n的改变各参数之间的变化关系如表2.5所示，此时实际输出的频率与n的关系曲线如图2.10所示。

表2.4 n固定时不同的 b 值与各参数之间的关系

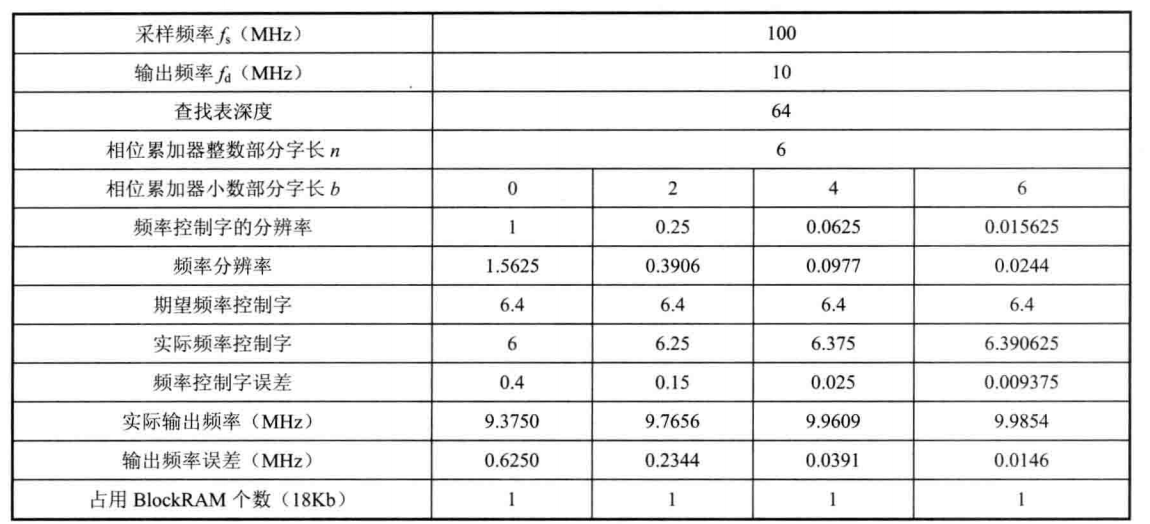
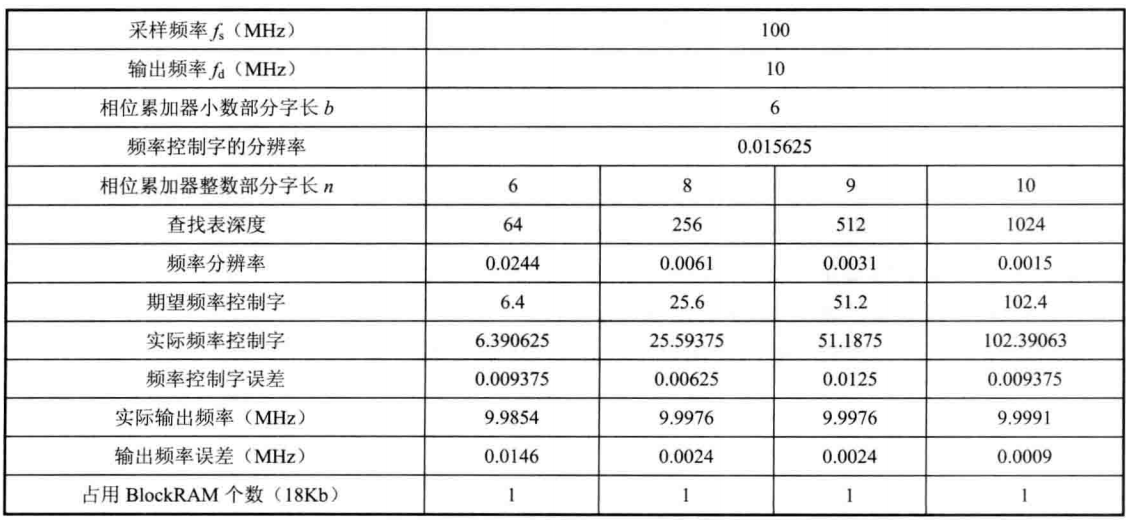


表2.5 b固定时不同的 n 值与各参数之间的关系



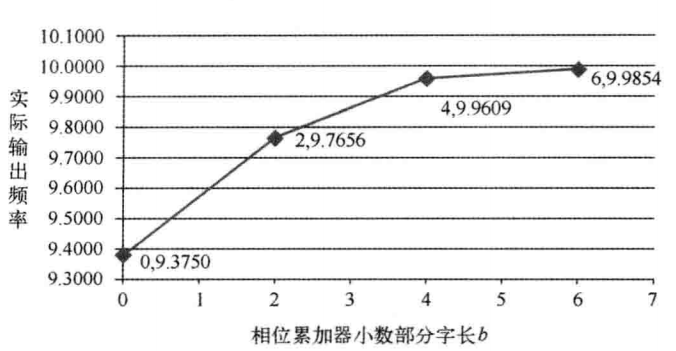


图2.9 n固定时实际输出频率与b的关系曲线

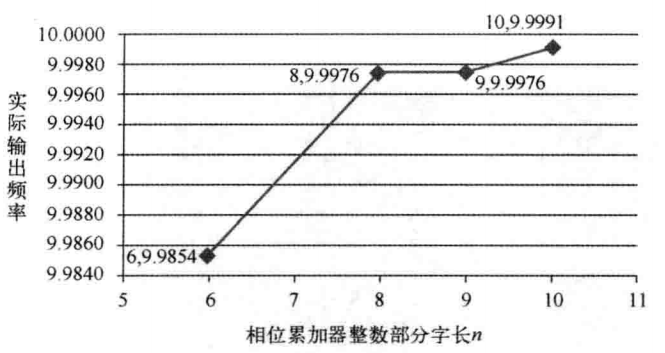
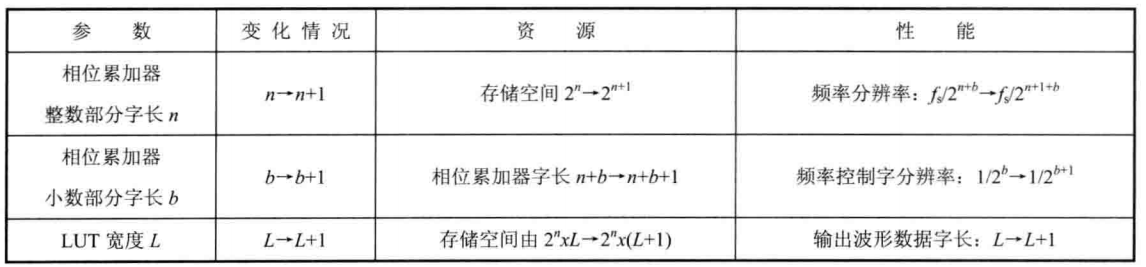


图2.10 b固定时实际输出频率与n的关系曲线

相位累加器整数部分字长n、小数部分字长b和LUT宽度对 DDS 性能的影响如表2.6所示。

表2.5 各参数对 DDS 性能的影响



在 SysGen 中搭建基于 LUT 的 DDS 如 SysGen 模型 2.2 所示。图中相位累加器整数部分字长n为 10，小数部分字长b为6，频谱分析仪的输出结果如图 2.11 所示。与图 2.4 相比，输出频率精度已有所改善，但 SFDR 为 59.688dBm，有所下降。SysGen 模型 2.2 中累加器的输出数据格式为 UFix\_16\_6，而 ROM 的地址端数据格式为UFix\_l0\_0，这意味着相位的截断(Phase Truncation)因此，常规型基于 LUT 的 DDS也被称为相位截断型 DDS。

SysGen 模型 2.2 在 SysGen 中搭建基于 LUT 的 DDS

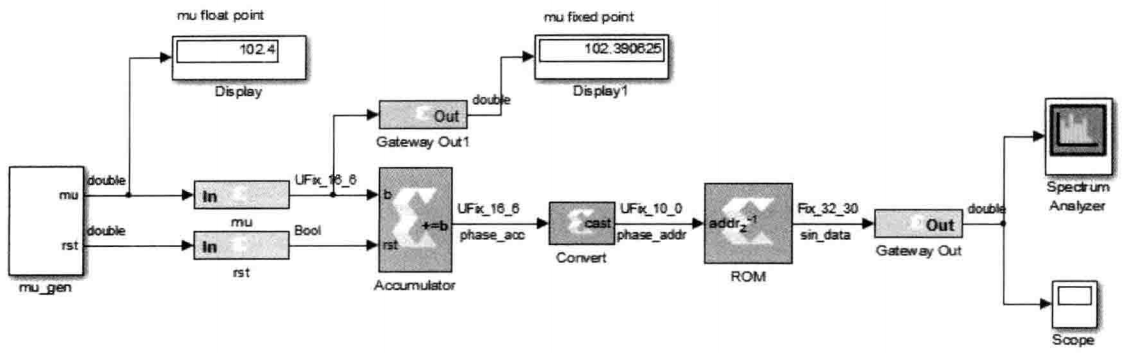


图2.11 SysGen 模型 2.2 中频谱分析仪的输出结果

2.2 通过 LFSR 改善 SFDR

SFDR 定义了目标频谱幅度与杂散频谱的最大幅度之差。 对于相位截断型 DDS 输出信号频谱的 SFDR 可表示为



式中 D 为 LUT 地址位宽。据此，我们通常粗略地认为 LUT 地址的每一位对 SFDR 贡献6dB。相位截断型 DDS 由于引入相位噪声而导致 SFDR 的下降。以相位累加器整数部分字长 n=6，小数部分字长 b=6、采样频率*fs* = 100MHz、期望输出频率 *fd*=10MHz 为例，图 2.8 中，相位累加器前 10 个输出相位点和相位量化模块前 10 个输出相位点如图 2.12 所示。可以看出，相位累加器的输出是等间隔步进，而相位量化模块的输出则是不等间隔的，在 6 和 7 之间跳变，从而引入相位噪声。

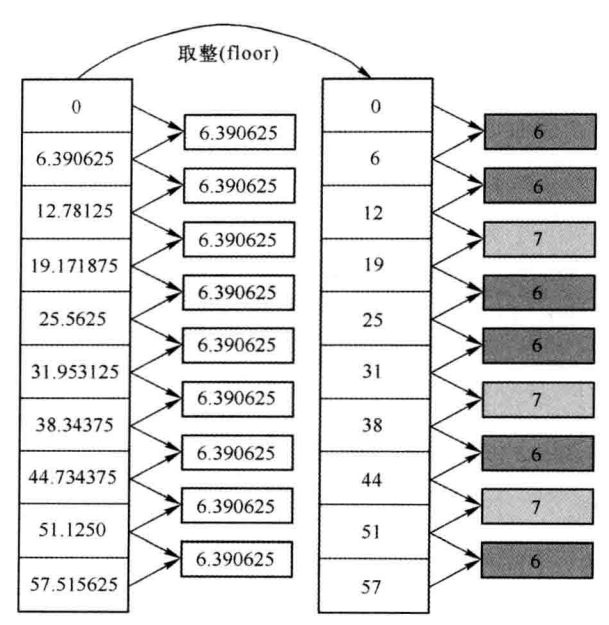


图2.12 相位累加器和相位量化模块前 10 个输出点的比较

为改善输出频谱的 SFDR，结合式 (2.8) 可知，提高 LUT 的地址位宽是一种方式，但这也会增大存储空间以及相位累加器的位宽，并不是一种经济的方式。我们可以在相位累加器的输出端引入相位抖动，破坏相位噪声的结构，如图 2.13 所示。图中 D(n) 即为相位抖动模块，其输出数据格式应为 UFix\_X\_b，即小数部分字长与相位累加器小数部分字长一致。实验表明，当X= b 时，对 SFDR 的改善最为明显。此时，D(n)输出值介于 0 和 1 之间。

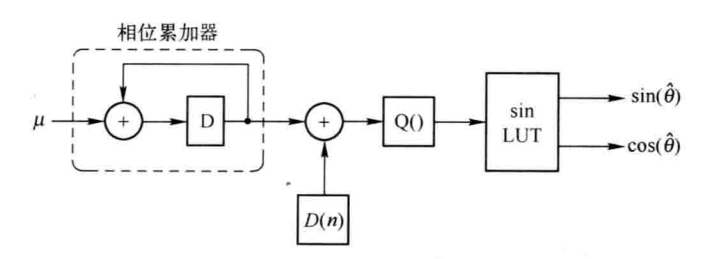
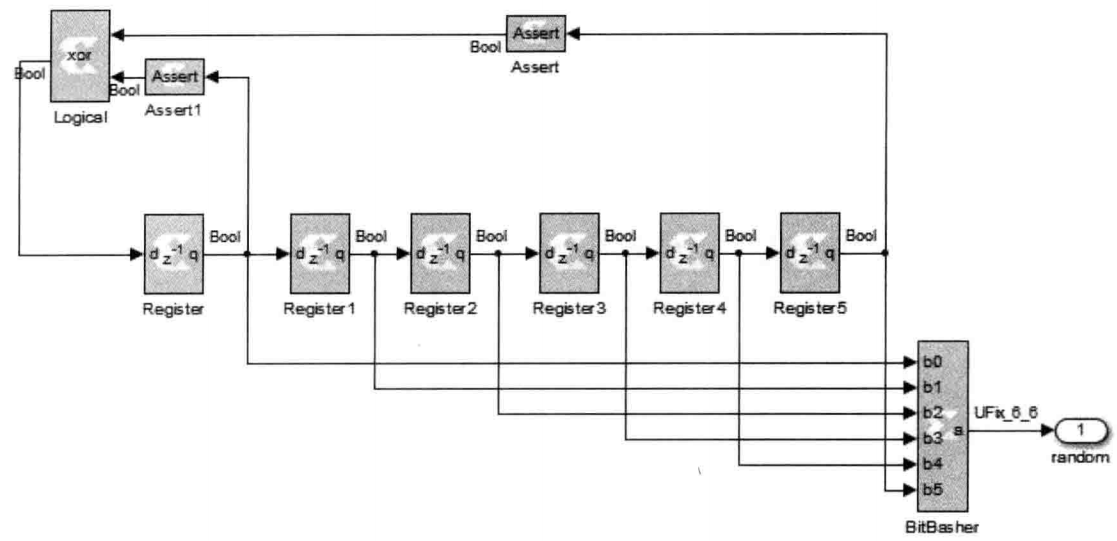


图2.13 带有相位抖动的 DDS 架构

D(n)可通过线性反馈移位寄存器LFSR(Shift Register SysGen)生成。对于M阶LFSR，产生的伪随机序列长度为2M-1。当b=6时，LFSR硬件架构如SysGen模型2.3所示，相应的DDS硬件架构如SysGen模型2.4所示，频谱分析仪的输出结果如图2.14所示。此时，SFDR 为 66.774dBm，与图 2.11 相比，提高了 7dBm。

SysGen 模型 2.3 b=6 时 LFSR 硬件架构



SysGen 模型 5.4 加入相位抖动的 DDS 硬件架构



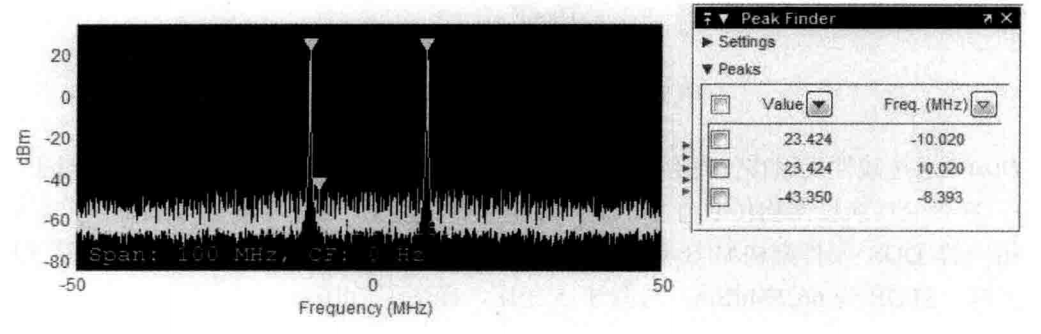


图2.14 SysGen 模型 5.4 频谱分析仪的输出结果

2.3 通过 Taylor 级数改善 SFDR

函数*f*(x)在 x0的 Taylor 级数可定义为:



函数 sin(x)和cos(n)根据式 (2.9) 展开并只取前两项可得:





由此可得相应的硬件架构如图 2.15 所示。与图 2.8 相比，多了相位转换模块和 Taylor 级数计算模块。相位转换模块的功能是完成相位累加器的输出到相位值的转换。事实上，相位累加器的输出是相位的索引。以***p***表示相位累加器的输出，表示相位量化的输出，phase 表示相位转换的结果，则三者的关系可表示为



式中n为LUT地址位宽，2π/2n确定了相位分辨率。硬件实现时，Taylor级数计算模块可以由DSP48E1完成。为了同时输出正弦波和余弦波，波形存储模块可由双端口ROM实现，只需要一套相位累加器产生正弦波所对应的地址S，余弦波所对应的地址C通过对S偏移实现，即S=C+A，其中A=2n/4。

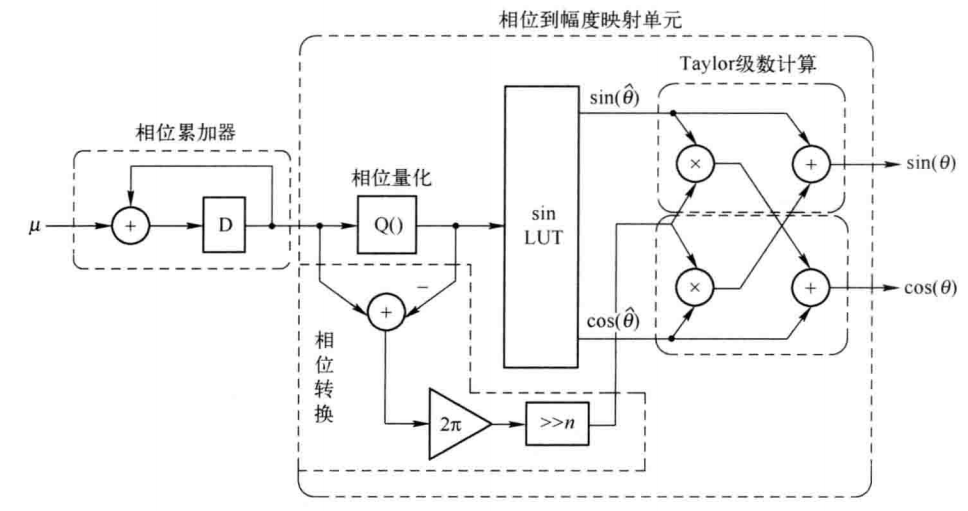


图2.15 通过 Taylor 级数改善 SFDR 的硬件架构

对于图 2.15 所示架构，仍以相位累加器整数部分字长n=10、小数部分字长 b=6 采样频率*fs* = 100MHz，期望输出频率*fd*= 10MHz 为例，频谱仪输出结果如图 2.16 所示，此时SFDR 为 101.722dBm，与图 2.11 相比提高了 42dBm。

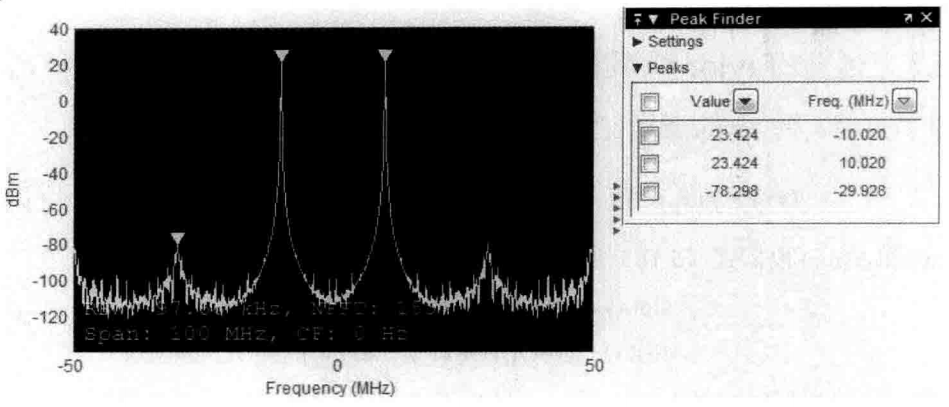


图2.16 引入 Taylor 级数计算模块之后频谱仪输出结果

**2.3 利用对称性压缩存储波形**

利用正弦波的对称性可只存储 1/2 周期或者 1/4 周期的波形，从而达到缩减存储空间的目的， 这就需要添加相位转换和幅度转换模块，如图 2.17 所示。这里以相位累加器整数部分字长为例。

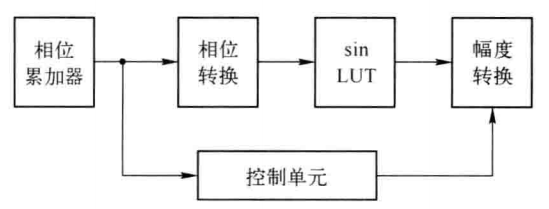


图2.17 压缩存储波形后的 DDS 架构

**2.3.1 存储 1/2 周期的情形**

n=4 时，如果存储一个完整周期的波形， LUT 的地址与存储数据的对应关系如图 2.18 所示。如果存储 1/2 周期的波形，LUT 地址的取整范围就变为[0,7]。当相位累加器输出值介于区间[8,15]时，就需要对其转换，这由相位转换模块完成，转换结果如图 2.19 所示。但此时，应对输出波形幅度进行调整使其与图 2.18 一致，这由幅度转换模块完成。

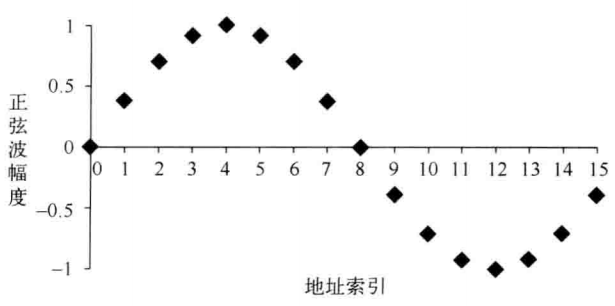


图2.18 完整周期波形时 LUT 地址与存储数据的对应关系

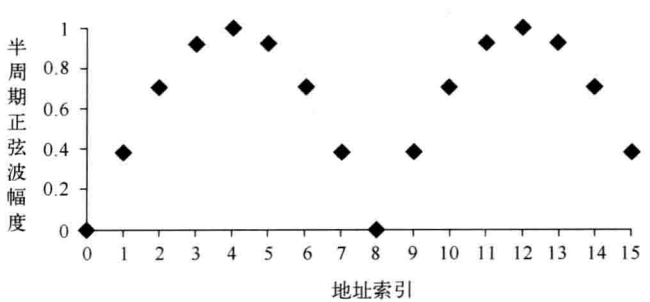


图2.19 压缩存储波形后的 DDS 架构

相位转换模块的功能如图 2.20 所示。当相位累加器输出值介于区间[8,15]时，取其低 3 位即可获得正弦波所需的地址，且此时需要对输出波形乘以-1；对正弦波地址加 4并取结果的低 3 位即可获得余弦波所需的地址。还可看出，相位累加器输出地址的高两位决定了输出波形所在的象限，如图 2.21 所示，从而也就确定了何时启动幅度转换模块。

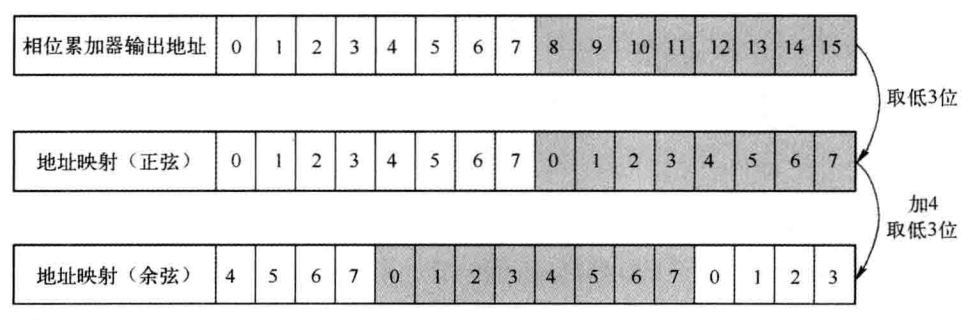


图2.20 地址之间的映射关系

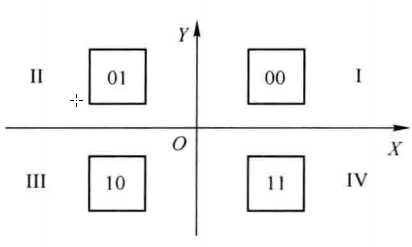
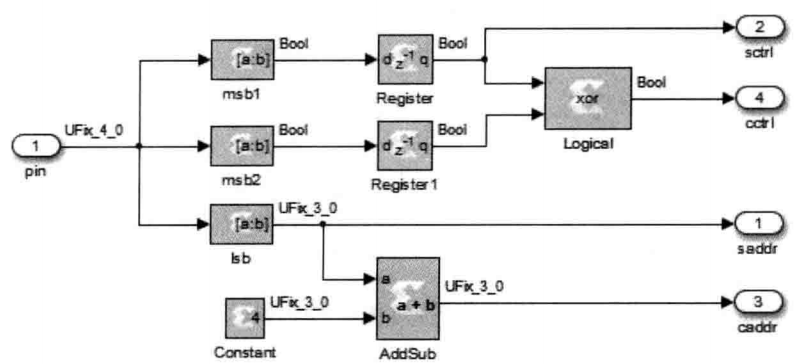


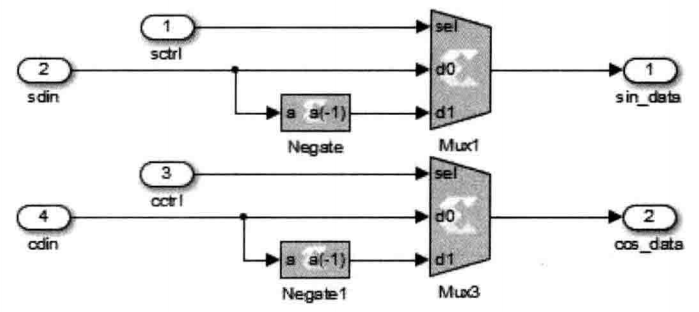
图2.21 相位累加器输出地址高两位与输出波形所在象限的关系

采用 SysGen 实现，相位转换模块如 SysGen 模型 2.5 所示。pin 为相位累加器输出地址，msb1 取相位累加器输出的最高位，msb2 取相位累加器输出的次高位。 sctrl 和 Ctrl 分别为正弦波和余弦波幅度转换控制信号；saddr 和 caddr 分别为正弦波和余弦波所需地址。幅度转换模块如 SysGen 模型 2.6 所示， 乘以-1 通过 Negate 模块实现。

SysGen 模型 2.5 存储 1/2 周期时 SysGen 下相位转换模块的实现



SysGen 模型 2.6 SysGen 下幅度转换模块的实现



**2.3.2 存储 1/4 周期的情形**

存储 1/4 周期波形时，相位累加器输出地址与正、余弦地址对应关系如图 2.22 所示。对正弦地址，当相位累加器输出地址的次高位为 0 时，取低 2 位赋给正弦地址；当为 1 时，将低 2 位取反赋给正弦地址。对余弦地址，当相位累加器输出地址次高位为 0 时，将低 2 位取反赋给余弦地址；当为 1 时，取低 2 为赋给余弦地址。在 SysGen 下实现方式如 SysGen 模型 2.7。

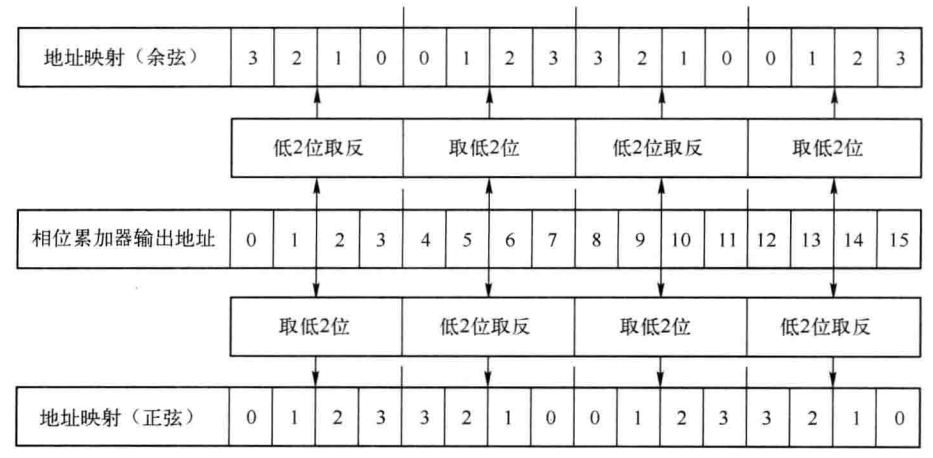
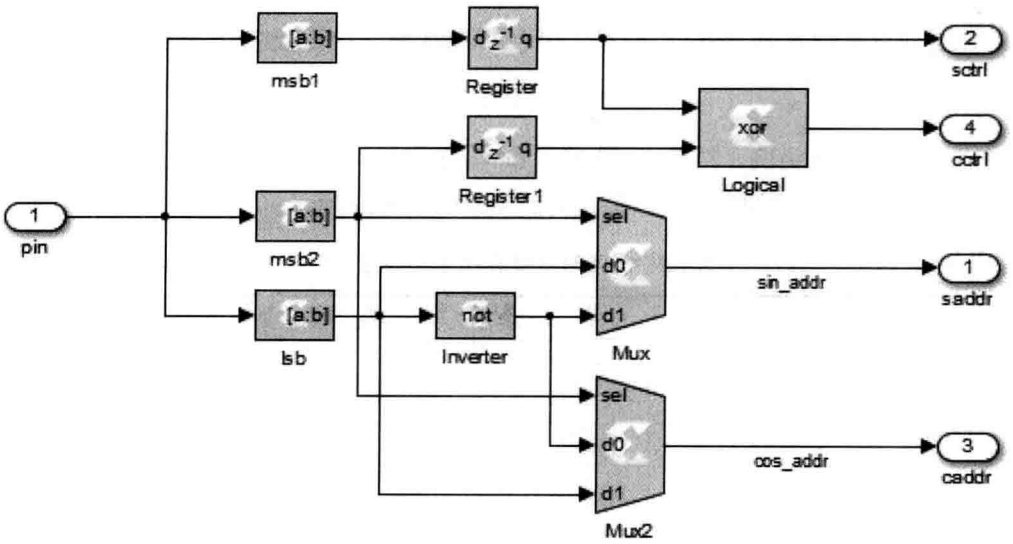


图2.21存储 1/4 周期波形时相位累加器输出地址与正、余弦地址的对应关系

SysGen 模型 2.7 存储 1/4 周期时 SysGen 下相位转换模块的实现



存储 1/4 周期时存在的问题是当相位累加器输出为 4 时，对应的正弦值应为



但实际输出值为



这是由相位转换导致的， 进一步会造成 SFDR 的损失， 而 1/2 周期存储则不存在这个问题。

3 基于双模互质算法的 DDS

**3.1 双模互质算法的基本原理**

双模互质算法是通过两个 DDS 合成目标频率。其中，两个 DDS 中的相位累加器的模值分别为N和N-1。考虑位宽为 n 的整数型相位累加器， 其模值为 N=2n，频率分辨率为：



频率控制字分辨率为 1，故可产生 N 个不同的频率，如式 (3.2) 所示。



对于模值为 N-1 的整数型相位累加器，其位宽仍为n，频率分辨率为:



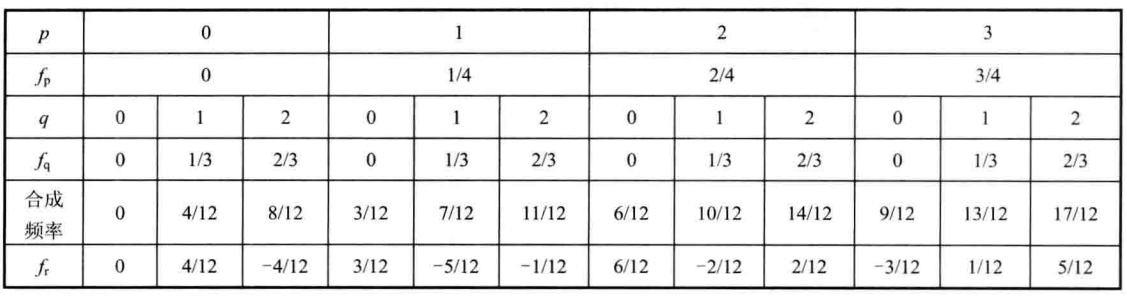
频率控制字分辨率为1，故可产生 N-1 个不同的频率，如式 (3.4) 所示。



以N=4，*fs*=l为例，结合式(3.2) 和式（3.4），两者可合成的频率如表 3.1所示。

此时，可合成 12 个不同的频率，它们之间的关系如图 3.1 所示。

表3.1 N=4 时双模值 DDS 可合成的频率



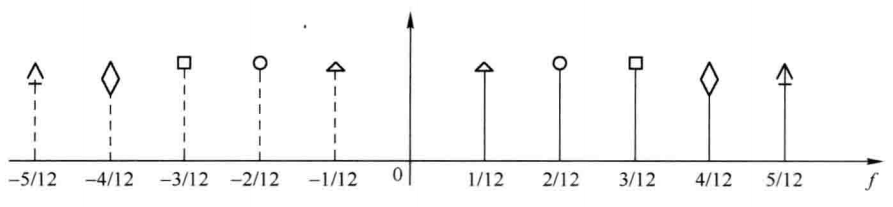


图3.1 相位累加器输出地址高两位与输出波形所在象限的关系

这种合成是基于三角函数公式：



其中，sinα和 cosα 由模值为 n 的相位累加器构成的 DDS 生成；sinβ和 cosβ由模值为N-1的相位累加器构成的 DDS 生成。由于N和N-1互质，因此， 可合成的频率个数为N(N-1)。

从表 3.1 还可看出，此时频率分辨率为



与式 (3.1) 和式 (3.3) 相比，分辨率有了明显地提高，这也正是该算法中相位累加器无须小数部分的原因。基于式（3.6)，两者合成的频率可表示为



*fr*可进一步分解为:



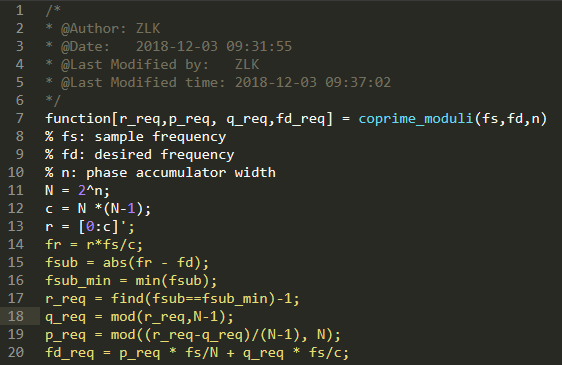
其中：





mod 表示取余。式(3.8)表明合成频率*fr*可由两部分构成，对应两个DDS。给定相位累加器位宽n、采样频率*fs*和目标频率*fd，*可通过 Matlab 代码3-1获取*p’*和*q’*。

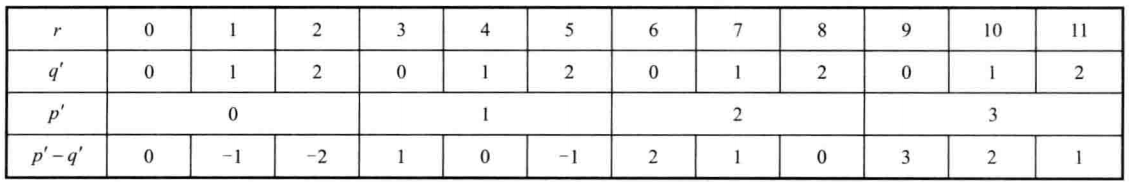
Matlab 代码 3-1 function: coprime\_moduli



Matlab 代码 3.1 表明，由于合成频率的个数有限，受制于N，要获取目标频率*fd*，事实上是寻找与*fd*最接近的人*fr*。N决定了频率分辨率，因此，在已知*fs*和*fd*的情况下，选取合适的N值尤为重要。

N为 4 时，*p’*和*q’*随r 的变化规律如表 3.2 所示。*p’*的取值范围为[0，N-1]，*q’*的取值范围为[0，N-2]，这也可由式(3.9)和式(3.10)印证。*p’*和*q’*可能为负值，此时代入式(3.9)中的频率控制字尽管为负值，但实际频率控制字为*p’-q’+N。* 若将式(3.9)中的*p’-q’* 替换为*p’-q’+N*，合成频率*f’r=fr+fs*，是*fr*的周期延拓。

表3.2 N=4 时*p’*和*q’*随r 的变化规律



根据以上分析，为获得*fr*，模值为N的相位累加器构成的 DDS 其频率控制字为:



模值为 N-1 的相位累加器构成的 DDS 其频率控制字为



**3.1 双模互质算法的硬件实现**

双模互质算法的硬件架构如图 3.2 所示，它由四部分构成：频率控制字合成单元、相位累加器模值为 N 的 DDS 相位累加器模值为 N-1 的 DDS 和频率合成单元。

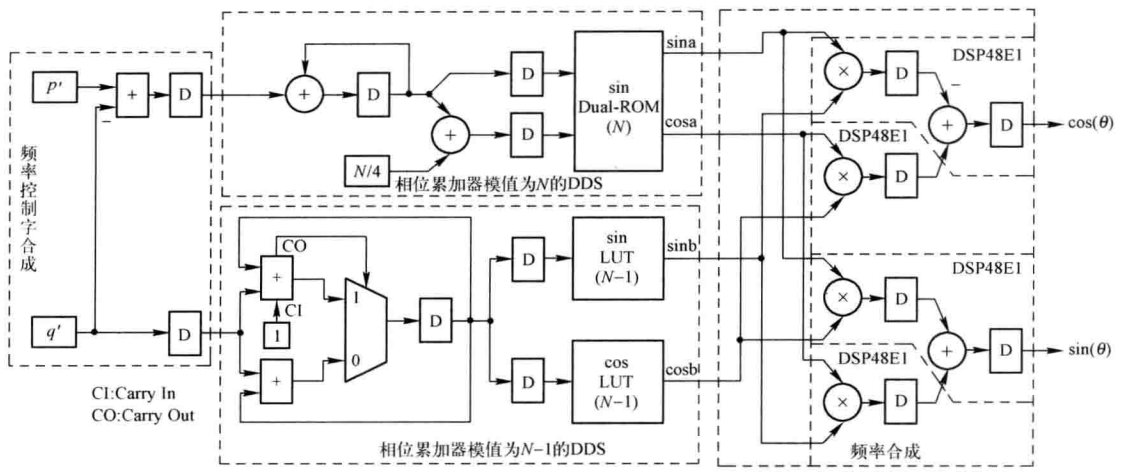
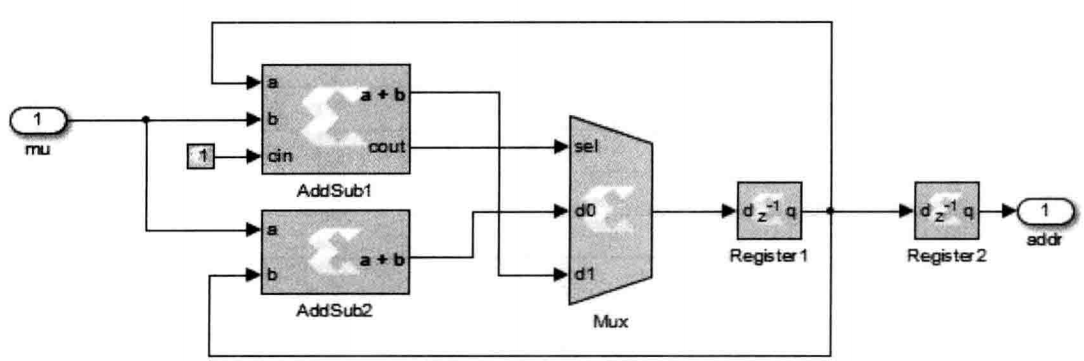


图3.2 双模互质算法的硬件架构

频率控制字合成单元执行的是*。*这里将*p’*和*q’*视为无符号数，其数据格式为UFix\_n\_0，两者相减的结果仍为无符号数，数据格式仍为UFix\_n\_0。这是因为若*p’*-*q’<0，*其有符号数二进制补码与*p’*-*q’+N* 的无符号二进制原码是一致的。两个 DDS单元与 2.2 节所述是一致的。模值为 N-1 的相位累加器在 SysGen 中如 SysGen 模型3-1 所示。频率合成单元执行的是式 (3.5) 的功能，需要占用 4 个 DSP48E1。

SysGen 模型 2-1 模值为 N-1 的相位累加器硬件架构



当n=10、*fs* = 100MHz 期望输出频率*fd*= 10MHz 时输出频谱如图 3.3 所示。此时， SFDR 可以达到 98.66dBm，与图 2.14 相比，提高了38.99dBm。

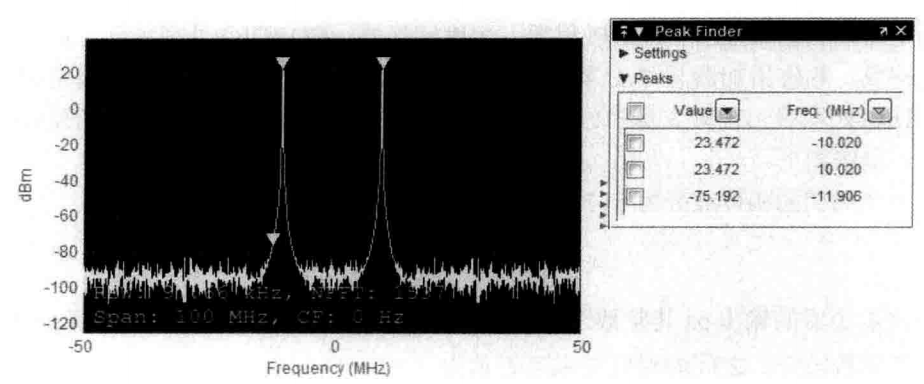


图3.2 n=10、*fs* = 100MHz、*fd*= 10MHz时双模值算法输出频谱

4 基于 CORDIC 算法的 DDS

基于 LUT 的 DDS 是静态获取正、余弦值（事先将正、余弦值存储在 ROM 中），而CORDIC 算法则是动态获取正、余弦值：根据给定的相位值，实时计算对应的正、余弦值。

在https://blog.csdn.net/Pieces\_thinking/article/details/83512820已经阐述，采用 CORDIC 算法可以获得给定角度的正、余弦函数值，重写其数学公式如图 4.1 所示。在该图中要求目标角度 z 的范围为[-99.9°，99.9°]，而 DDS 中输入角度为[0°，360°]中的任意值。这就需要将 DDS 输入角度所在区间映射到 CORDIC 算法所要求的区间。事实上，[-99.9°，99.9°]包含了[-90°，90°], 从而可以很方便地利用三角函数公式完成区间映射。于是，问题的关键即转换为如何获取目标角度。

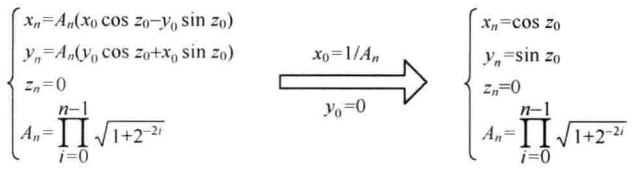


图4.1 利用 CORDIC 算法计算正、余弦函数值

联想上一节中基于 LUT 的 DDS:把相位累加器的输出作为 LUT 的地址， LUT 内存储的是与地址一一对应的正、弦函数值。本质上，该地址是与相位一一对应的。因此，只要得到地址即可得到相位，也就是相应的目标角度。从而，基于 CORDIC 算法的 DDS 系统框图如图 4.2 所示。

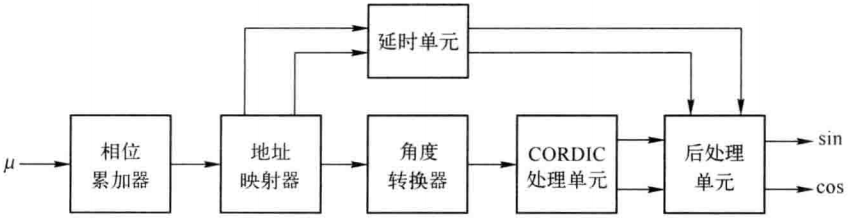


图4.2 基于 CORDIC 算法的 DDS 系统框图

整个电路由相位累加器、地址映射器、角度转换器、CORDIC 处理单元、后处理单元和延时单元构成。相位累加器与 2 节中完全一致，其输入是根据式 (2.8) 确定的相位步进值。CORDIC 处理单元在https://blog.csdn.net/Pieces\_thinking/article/details/83512820中已详细阐述，这里重点介绍地址映射器和角度转换器。为便于说明，举例如下。

假定一个周期的相位被分为 512 份， 则相位分辨率 pr 为

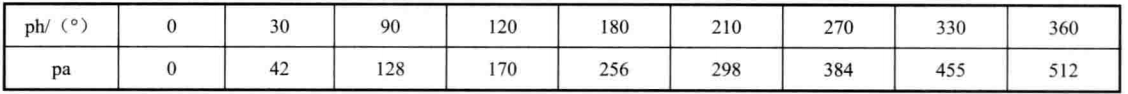


这表明相位累加器的输出 pa 其整数部分只需要 log2512=9 bit 来表示。相位值 ph 与 pa （这里只用到其整数部分）之间的对应关系可表示为



此时 ph 的取值范围为[0°，360°]。根据式 (3.14) 可得表 4.1 所示的 ph 与 pa 之间的具体数值对应关系，这里是在四个象限中各取了一个相位值以便于说明。

表4.1 ph 与 pa 的对应关系



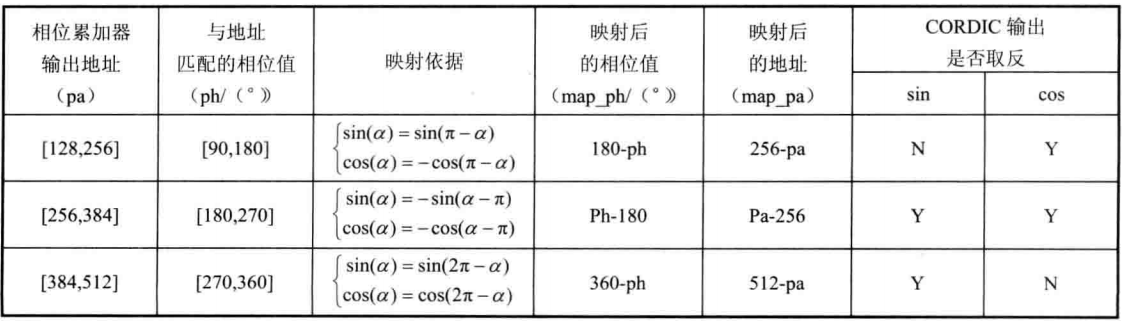
为了正确使用 CORDIC 算法， 需要将 ph 值限制在[-90°,90°]。结合表 4.1 即需要对第二、三、四象限中的相位进行处理，本质上需要对相应的 pa 进行处理。当 ph 处于第二象限时，即 ph 处于区间[90°，180°]，对应的相位累加器的输出地址 pa 范围为[128,256]。以表 4.1中的120°为例，对应的 pa 为 170, 根据式 (4.3) 所示的三角函数公式可知，此时需将 pa 映射为 256-170=86。将 pa 值为 86 代入式 (4.2) 中可知对应的 ph 值为 60° 。对于余弦，由式(4.4)可知，需要将 CORDIC 所求结果取反。





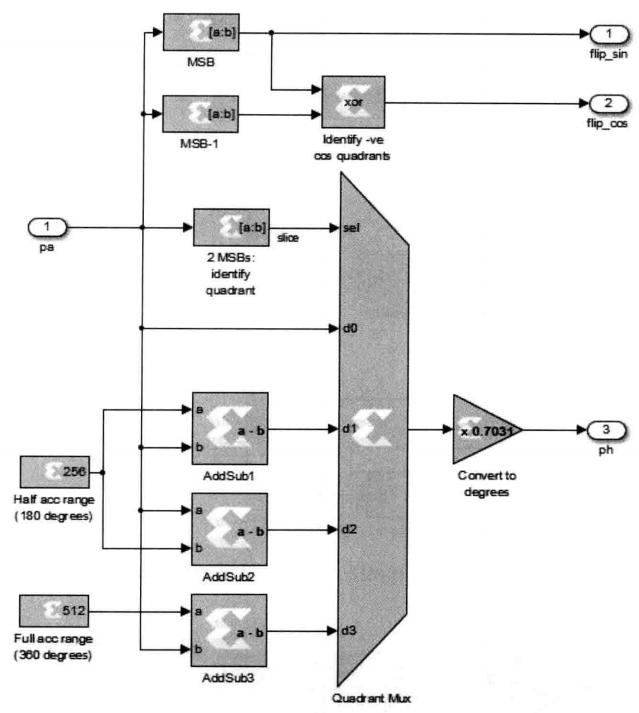
类似地，根据相应的三角函数公式可完成第三、四象限相位值对应的地址映射，如表4.2所示。表中 N (No) 表示 CORDIC 输出无须取反（乘以-1 )，Y (Yes)表示 CORDIC 输出需取反。

表4.2 相位累加器输出地址映射前后的关系



在 FPGA 设计时，需要根据相位累加器的输出 pa 来判断此时所处区间。以表 4.2 为例， pa 输出最大值为 512, 需要 9bit 二进制表示。这样最高位和次高位就决定了所处区间，同时也决定了是否对 CORDIC 输出取反。从而，形成了如 SysGen 模型 4-1 所示的硬件架构。

SysGen 模型 4-1 地址映射+角度转换电路硬件架构



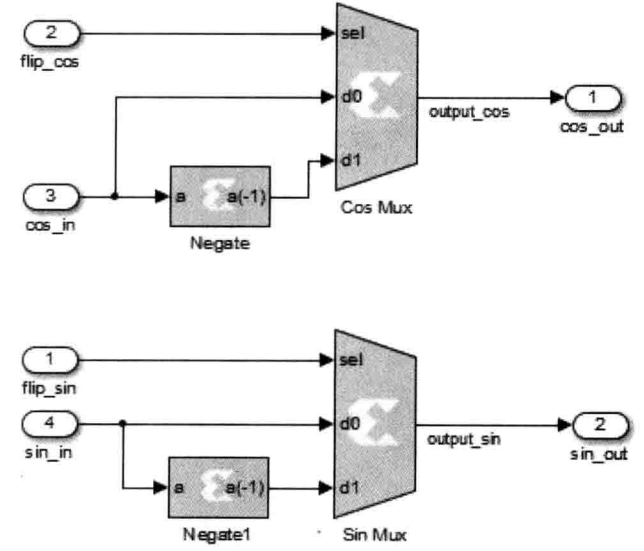
SysGen 模型 4-1 中，模块 MSB 获取 pa 的最高位，MSB-1 获取 pa 的次高位，3 个减法器和一个 MUX 构成了地址映射单元。MUX 的控制端由 pa 的最高位 (MSB )和次高位(MSB-1 )拼接而成。可对 MUX 的输出寄存以减小逻辑延时。此电路同时获得了判断CORDIC模块输出正、余弦结果是否取反的标志信号 flip\_sin 和 flip cos。flip sin 即为 pa 的最高位，flip\_cos 则是 pa 最高位和次高位异或结果。但是，这两个信号在输出之前都需要经过延时处理模块，该模块确定了 flip\_sin 和 flip\_cos 在输出之前需要经历的延时级数。该数值由 CORDIC 算法流水级数决定。

获得了匹配的地址就需要将其转换为相应的角度，这可通过角度转换器完成。如前所述，相位累加器输出的地址值与相位值是一一对应的，这种对应关系体现在式 (4.2) 中。因此，角度转换器就是一个简单的乘法器，且该乘法器的一个输入端为固定常数 pr。SysGen 模型 4-1 中 pr 为 0.7031。

获得了角度值即完成了相位索引到角度的转换，就可以把这个转换结果送给 CORDIC处理器。为了提高计算精度，可适当增加 CORDIC 算法的迭代次数。当然，这会导致资源的增加以及 Latency 增大。

最后一个环节根据地址映射器给出的控制信号对 CORDIC 的输出结果进行调整，调整依据表 4.2 进行。这部分的硬件架构如 SysGen 模型 4-2 所示。

SysGen 模型 4-2 后处理单元硬件架构



n=10、b=l2、*fs* = 100MHz 期望输出频率*fd*= 10MHz时采用 CORDIC 算法迭代 9 次DDS输出频谱如图 4.3 所示。

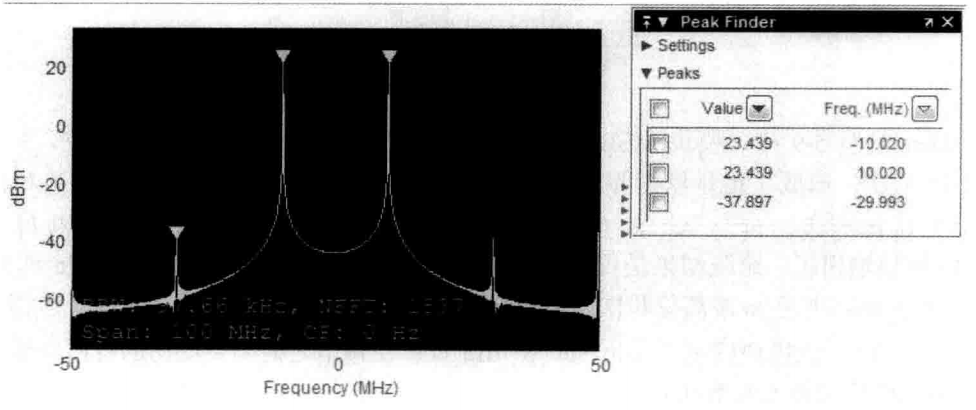


图4.3 n=10、b=l2、*fs* = 100MHz、*fd*= 10MHz时采用 CORDIC 算法迭代 9 次 DDS 输出频谱

5 多通道 DDS

多通道 DDS 旨在通过分时复用技术同时产生多个不同中心频率的正、余弦波，其原理与单通道 DDS 是一致的。以 M 通道为例，假定系统工作时钟频率为 *fclk，*那么每个通道的



这意味着每个通道所能获得的最高中心频率为 *fms*/2 。

多通道 DDS 是在单通道 DDS 的基础上添加分时复用单元而形成的，以 4 通道 DDS 为例，其结构框图如图 5.1 所示。图中相位到幅度映射单元与图 2.15 完全一致，而相位累加器则略有不同，其结构如 SysGen 模型 5-1 所示，这即是滑动累加器。这里重点介绍一下分时复用单元。

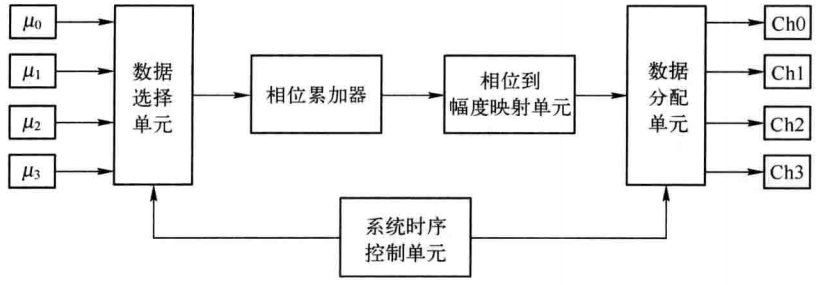
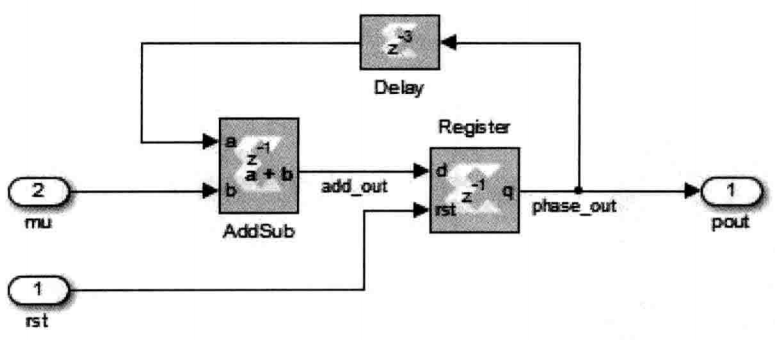


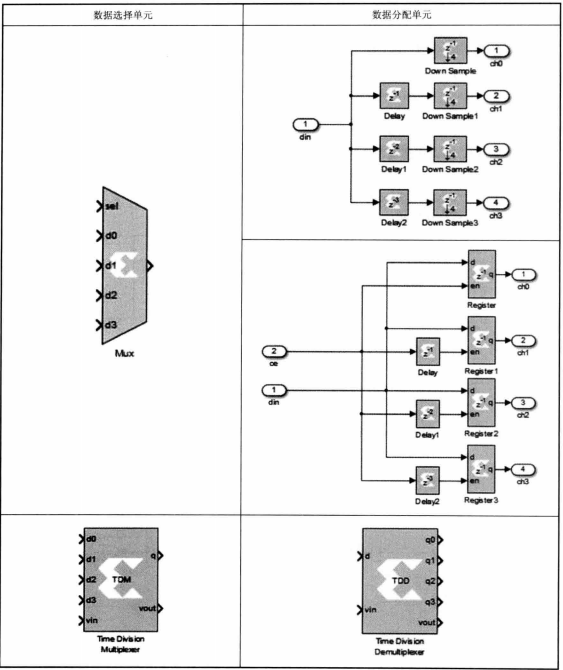
图5.1 4通道 DDS 结构框图

SysGen 5-1 多通道结构下的相位累加器



分时复用单元由两个模块构成：数据选择单元和数据分配单元。在 SysGen 里，分时复用单元可以用不同的模块实现，如表 5.1 所示。若数据选择单元采用 4:1 MUX 实现，则相应的数据分配单元可以采用 Delay+Down Sample 模块或者 Delay+Register(ce)实现；若数据选择单元采用 TDM 模块实现，数据分配单元则可采用与之匹配的 TDD 模块实现。

表5.1 SysGen 里分时复用单元的不同实现方式



6 多路并行 DDS

采用多路并行 DDS 技术旨在等效提高 DDS 的工作时钟频率，扩展 DDS 的输出带宽。单路 DDS 输出频率小于 *fs* /2 为了提高 DDS 的输出频率，就需要提高系统时钟频率。而对 FPGA 而言，系统时钟频率的提升是有限的。多路并行 DDS 技术解决了这一问题。

为便于说明，以 4 路并行 DDS 为例。4 个相位累加器输出时序如图 6.1 所示。可以看出，4 个相位累加器同时输出 4 个等间距的相位码，进而可获得4 个波形幅度数据，这个间距即为频率控制字，图中频率控制字μ=1。对任何一路相位累加器而言，相位步进为频率为频率控制字的 4 倍。在0号相位累加器(p0)输出4时，己经获得了 4 个不同的相位码，分别来自于 4 个不同的相位累加器。这等效于在 0 号相位累加器的相位码 0 和 4 之间内插了 3 个相位码。这意味着采样间隔由原来的 Ts (Ts=Tclk)加快到 Ts/4，也就是采样频率提高为原来的 4倍。换言之，4 路并行 DDS 等效于单路采样频率为 4*fs*的 DDS。与单路采样频率为 *fs* 的DDS 相比，输出带宽由*fs*/2 增大到 2*fs*。

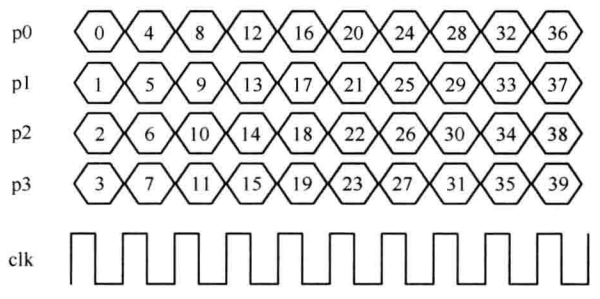


图6.1 4个相位累加器输出时序

对于 m 路 DDS，每一路 DDS 具有相同的规格，即具有相同规格的相位累加器和相位到幅度映射单元。假定相位累加器位宽为n，那么 m 路 DDS 的频率分辨率为



输出频率为：



每一路相位累加器的步进值为 mμ。mμ的最大值为 2n- 1，故μ的最大值为



将其代入式 (6.2) 可得输出最大频率为



对于 4 路并行 DDS 其系统框图如图 6.2 所示。4 个相位累加器的初值由 μ 决定，分别现为，0、μ、2μ和 3μ，步进值均为 4μ。2μ可由μ左移 1 位实现，3μ可由 2μ 和μ相加实现，4μ可由 μ 左移两位实现。在 SysGen 下的实现框图如 SysGen 模型 6-1 所示，图中phase\_gen 的实现框图如 SysGen 模型 6-2 所示。

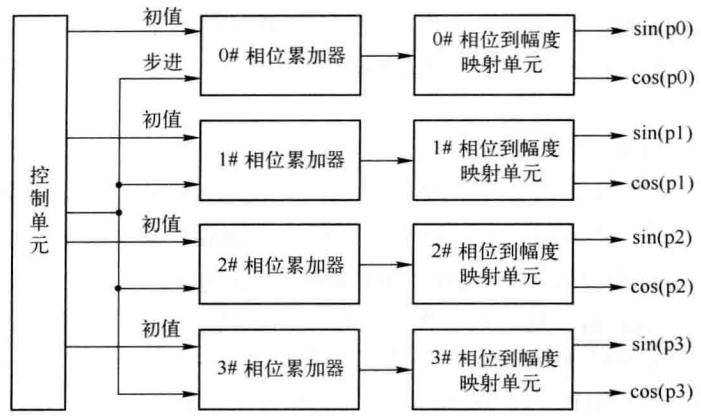
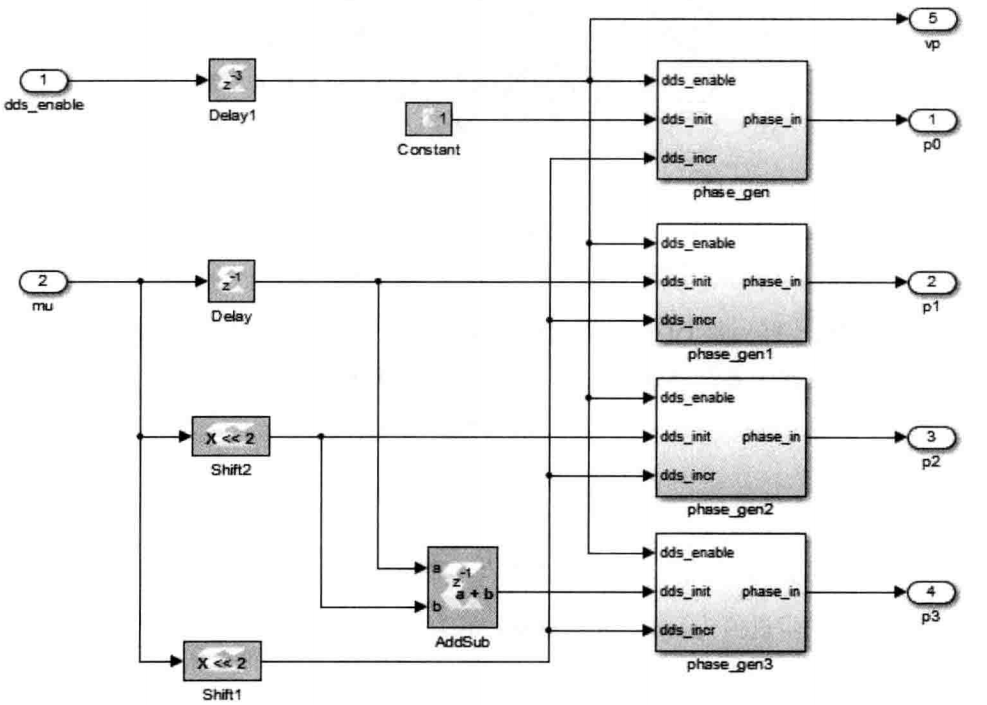
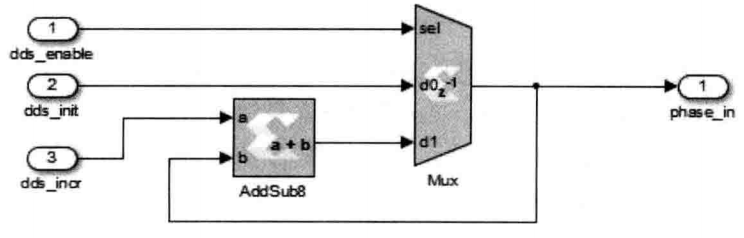


图6.2 4路并行 DDS 系统框图

SysGen 模型 6-1 4 路并行相位累加器实现框图



SysGen 模型 6-2 phase\_gen 的实现框图



4 路并行 DDS 的输出可以通过时钟频率为 4 *fs*的 MUX 转化为串行数据流，或者通过FPGA内部的 OSERDES 给 DAC 也可直接使用即多相并行处理。

以 m= 4、n=12、*fs* = 320MHz 期望输出频率*fd*= 300MHz为例，此时根据式(6.2)可知μ=960。输出频谱如图6-3所示。

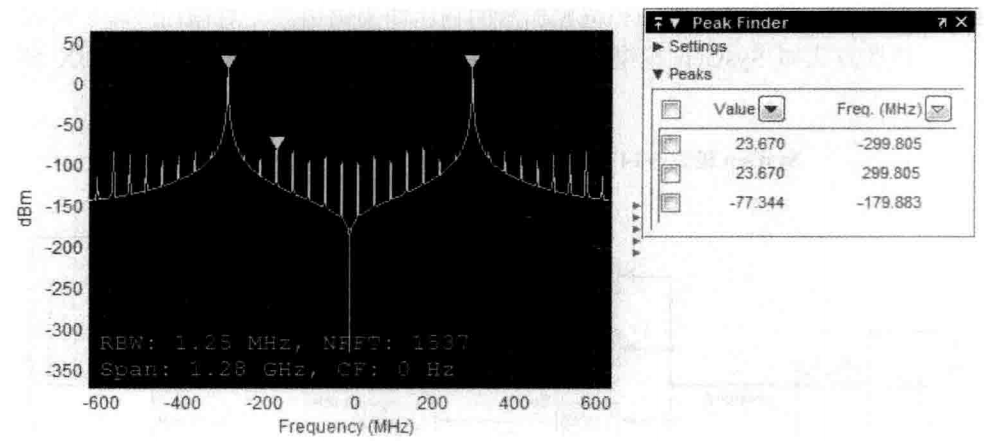


图6.3 m= 4、n=12、fs = 320MHz、fd= 300MHz时并行 DDS 输出频谱

7 产生其他波形

方波、锯齿波和三角波在设计中也会用到。相比于正弦波，它们的产生较为简单。对于方波，可通过计数器产生。以 4bit 计数器为例，模值为 16，输出波形如图 7.1 所示。

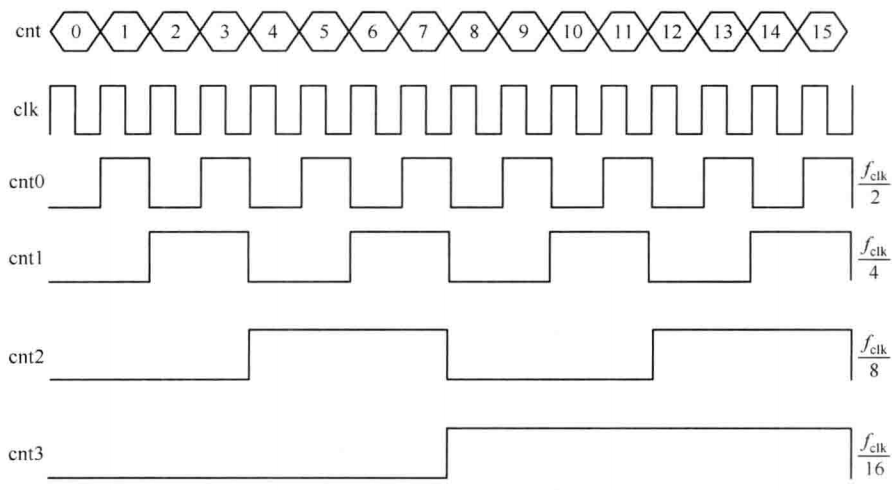


图7.1 4bit 计数器输出波形

图中 cnt0 表示计数器的最低位，cm3 表示计数器的最高位。若计数器工作时钟频率为*fclk*，则计数器第i(i= 0，1，2,…）位输出方波的频率为



因此，n位计数器输出方波的频率范围为[*fclk/2, fclk/2n*]。

锯齿波在DDS中己经用到，相位累加器的输出即为锯齿波。对锯齿波稍加转换即可产生三角波，转换方式如SysGen 模型7-1。图中Counter的输出为锯齿波，MUX的输出为三角波。

SysGen 模型 7-1 产生方波、锯齿波和三角波的系统框图

