清 华 大 学

综 合 论 文 训 练

题目：超高速、高精度数控振荡器(NCO)设计

系 别：电子工程系

专 业：电子信息科学与技术

姓 名：杨一雄

指导教师：杨华中 教授

2017 年 5 月 28 日

中文摘要

数控振荡器，是一种数字信号控制频率、相位的信号发生器。直接数字频率合成(DDS)方法是一种全数字的实现方法，其优点包括精细的率分辨率、极快的转换时间和频率范围广等，应用领域非常广泛。传统的查找表结构DDS存在一个主要问题，导致速度和精度不能兼顾。多年来，很多研究者都致力于解决该问题，并将DDS性能不断提升。综合来看，DDS的技术和本次设计相当契合。

本文采用直接数字合成(DDS)方法，实现了一种高速高精度的数控振荡器的设计和仿真。本文提出了一种ROM-CORIDC混合结构，在噪声量级不变的情况下，查找表地址缩减了一半以上。通过合理资源配置、流水线加速等方法，大大加速了时钟频率。最终在前、后端测试中可满足2 GHz和1.7 GHz的时钟约束，频率分辨率为16 bits，最大杂散分量(SFDR)稳定在100 dBc，而功耗为23.4 mW。在高速DDS电路中，本设计的功耗和频谱纯度均达到领先水平。

关键词： 数控振荡器；DDS；混合结构；高频谱纯度；低功耗

ABSTRACT

Keywords:

目 录

[中文摘要 I](#_Toc483248834)

[ABSTRACT II](#_Toc483248835)

[目 录 III](#_Toc483248836)

[第1章 引言 1](#_Toc483248837)

[1.1 研究背景 1](#_Toc483248838)

[1.2 主要评价指标 1](#_Toc483248839)

[1.3 研究现状 2](#_Toc483248840)

[1.3.1 锁相环式(PLL)频率综合 2](#_Toc483248841)

[1.3.2 直接数字式频率综合(DDS) 3](#_Toc483248842)

[1.3.3 DDS近年工作 3](#_Toc483248843)

[1.4 论文主要工作 4](#_Toc483248844)

[第2章 直接数字式频率综合(DDS)原理 6](#_Toc483248845)

[2.1 DDS的原理和框架 6](#_Toc483248846)

[2.2 查找表压缩技术 8](#_Toc483248847)

[2.2.1 对称性方法 8](#_Toc483248848)

[2.2.2 引入近似结构 9](#_Toc483248849)

[2.3 角度旋转方法介绍 10](#_Toc483248850)

[2.3.1 CORDIC算法原理 11](#_Toc483248851)

[2.3.2 混合结构 12](#_Toc483248852)

[2.3.3 改进算法 13](#_Toc483248853)

[2.4 DDS的频谱特性和误差来源 14](#_Toc483248854)

[2.4.1 频谱分析 14](#_Toc483248855)

[2.4.2 噪声来源分析 15](#_Toc483248856)

[第3章 基于直接数字式的振荡器设计 18](#_Toc483248857)

[3.1 ROM-CORDIC 混合结构设计 18](#_Toc483248858)

[3.1.1 相位累加器(PA)设计 19](#_Toc483248859)

[3.1.2 相位压缩器(PC)设计 19](#_Toc483248860)

[3.1.3 相位幅度转换器(PAC)设计 20](#_Toc483248861)

[3.2 资源配置分析 21](#_Toc483248862)

[3.3 流水线设计 22](#_Toc483248863)

[第4章 数控振荡器实现和仿真 24](#_Toc483248864)

[4.1 功能性仿真平台搭建 24](#_Toc483248865)

[4.1.1 MATLAB数值计算平台仿真 24](#_Toc483248866)

[4.1.2 modelsim仿真平台搭建 26](#_Toc483248867)

[4.2 关键路径优化方法 26](#_Toc483248868)

[4.2.1 电路结构规范 26](#_Toc483248869)

[4.2.2 关键模块重新设计 28](#_Toc483248870)

[4.3 时序仿真结果 28](#_Toc483248871)

[4.3.1 工具介绍 28](#_Toc483248872)

[4.3.2 结果展示 28](#_Toc483248873)

[4.4 性能比较 31](#_Toc483248874)

[第5章 结论 32](#_Toc483248875)

[5.1 主要工作总结 32](#_Toc483248876)

[5.2 未来工作展望 32](#_Toc483248877)

[插图索引 33](#_Toc483248878)

[表格索引 34](#_Toc483248879)

[参考文献 35](#_Toc483248880)

[致 谢 37](#_Toc483248881)

[附录A 外文资料调研阅读报告或书面翻译 38](#_Toc483248882)

# 引言

## 研究背景

当今时代，无线通信系统正在全球急速扩展，生活中方方面面都离不开无线网络、蜂窝信号。新的应用如雨后春笋版出现，基于毫米波的5G移动网络正从实验室走向生活，但是这也对无线系统提出了更高的要求。在工业、军事领域，新型的武器、雷达也依赖于高精度的无线信号。作为系统中的信号源，实现高速、高精度、快切换、低功耗的频率合成器已经迫在眉睫。

信号合成技术已有数十年的发展，最初的振荡器由晶振和滤波器构成，但是鉴于其巨大的功耗、体积以及糟糕的精度，很难实用。而数控振荡器利用高精度的数字信号输入，在可控性以及精度得到了巨大提高。

数控振荡器分主要分为两类：一类是利用锁相环跟踪参考信号，生成指定频率的输出信号；另一类是直接数字合成式，通过相位直接映射成正弦幅度。下面将介绍两种方法的技术特点。

## 主要评价指标

**1) 信噪比(SNR)**

信噪比是信号处理领域的主要评价指标，用于描述信号和噪声能量比值。对于NCO而言，即中心频率能量和非中心频率能量之比。



图 1.1 信号中的底噪和杂散

SNR的数值计算如公式(1-1)。在NCO的设计中，SNR主要用于衡量底噪的大小，图1.1的(a)图显示了输出频谱中的底噪。

(1‑1)

**2) 最大杂散分量(SFDR)**

对于正余弦信号，不仅要降低总噪声的能量占比，减少信号中有色噪声也非常重要，因此使用SFDR用于评价输出的杂散性能。SFDR的定义为基频信号和最大杂散信号的能量之比。

参考图1.1的(b)图，在底噪同样较小的情况下，(b)中的杂散对输出性能起到了主要影响。频率合成领域更重视高次谐波的消除，因此SFDR指标的重要性一般要高于SNR。

## 研究现状

数控振荡器分主要分为两类：一类是利用锁相环跟踪参考信号，生成指定频率的输出信号；另一类是直接数字式，通过数字相位直接映射成正弦幅度。下面将介绍两种方法的技术特点。最后介绍近年DDS的主要发展。

### 锁相环式(PLL)频率综合

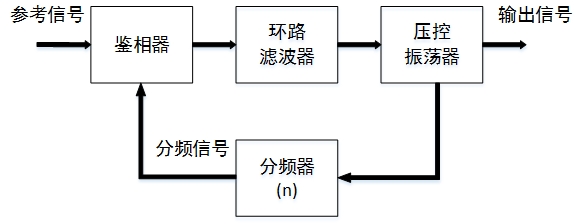


图 1.2锁相环式频率综合原理

锁相环式频率综合技术，又称“间接频率综合”。如图1.2所示，锁相环由鉴相器，环路滤波器，压控振荡器，分频器构成负反馈环路[1](RolandE.Best, 2007){RolandE.Best, 2007 #56}。其原理是通过分频信号对参考信号进行相位跟踪，环路滤波后经压控振荡器放大输出，将生成倍频的输出信号。

改变分频器的分频倍数n，即可控制输出正弦波频率。这种频率合成器的结构成熟且对功耗要求不高，输出信号的杂散性能较好。数字分数锁相环[2]的出现使其各项性能获得大幅提高，成为现今信号发生器的主流选择。

### 直接数字式频率综合(DDS)

DDS[3]的思路是从相位角度出发，将相位直接转化成波形，其基本结构分三块：相位累加器(PA)，相位-幅度转换器(PAC)，数模转换器(DAC)，如图1.3所示。各模块的承担的工作如下：

1）相位累加器用于计算信号相位

2）相位-幅度转换器将相位映射成非线性波形

3）数模转换器将波形转换成模拟信号，低通滤波后输出。

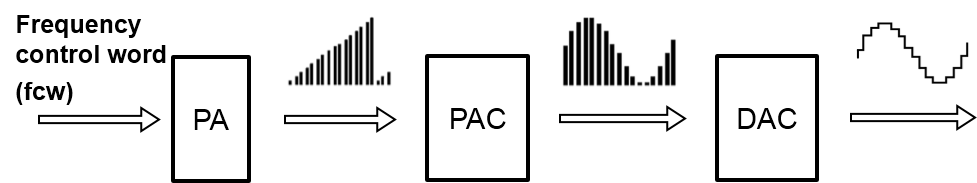


图 1.3 DDS基本结构

由于杂散性能过剩，通常还会在系统中对相位进行压缩，以减小PAC的复杂度。DDS在捷变性能和频率分辨率上相比锁相环式有巨大的优势，最大杂散分量可以达到120 dBc[4]以上。DDS的主要问题是时钟频率相对比较慢，但是在大规模集成电路(VLSI)迅速发展的背景下，DDS的时钟频率也在不断提高。随着高精度信号源的需求增大，不断有新领域开始使用DDS作为信号源。

### DDS近年工作

随着CMOS集成电路的成熟，DDS于1971年被Tierney *et al*[2]提出，核心观点是用ROM查表得到波形。这种方法在精度、捷变性能上相比其他频率合成器有突出的优势。但过大的ROM导致了面积偏大、速度较慢等问题。

在90年代，经过的多位学者的改进，引入了相位压缩的方法，以Sunderland[5]，Nicholas[6]为代表。其工作均利用高阶近似条件，实现了查找表多个数量级的减少，加快了DDS的时钟频率。随后又有研究者开始使用线性插值[7]、非线性插值[8]、非均匀差值[9]等方法，使得查找表大小进一步减小。

由于查找表压缩方法的增益收紧，研究者开始使用角度旋转替代查表方法。CORDIC算法[10]是支撑角度旋转的经典算法，通过多级移位、加法操作即可实现角度的旋转。查表方法的问题是ROM大小和地址长度呈指数关系，而角度旋转的优势是资源随相位位数线性增长。角度旋转法的主要优点是SFDR可以达到很高[4]，因为要进行加法计算，速度相比而言会比较慢一点。

近年来，使用非线性DAC实现PAC成为主流[11,12]，这种方法把PAC数字域的操作转移至DAC中完成。相比之前的工作，使用非线性DAC在面积、功耗上取得进步，速度上也有一定的提升[[1]](#footnote-1)。但是引入了DAC设计中的新问题。

表 1.1 近年代表性工作

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DDS结构 | 年份 | 工艺 | 时钟频率 | SFDR | 功耗 |
| Nicholas[6] | 1998 | 0.8 um | 150 MHz | 52 dBc | 0.5 W |
| 非线性内插[8] | 2010 | 0.13 um | 1.0 GHz | 63 dBc | 8.2 mW |
| 角度旋转[13] | 2011 | 0.18 um | 260 MHz | 113 dBc | 16.5 mW |
| 非线性DAC[12] | 2015 | 55 nm | 2.0 GHz | 55 dBc | 130 mW |

表1.1展示了近20年不同DDS结构的代表性工作。在结构创新、工艺升级的推动下，DDS在时钟频率、SFDR、功耗三项主要指标上都取得实质性进步。

## 论文主要工作

在对频率综合技术深入的研究后，结合毕业设计的实际需求、任务挑战度，本文决定使用直接数字合成(DDS)方法实现数控振荡器。为了设计出超高速、高精度直接数字式频率合成器(DDS)，毕设的研究目标有一下三点：

（1）提升输出信号的杂散性能

（2）提高系统的最快时钟频率

（3）降低系统的功耗、面积

本文主要使用查表法和角度旋转法结合的方案，主要工作按5点展开：

（1）调研基于查表法和角度旋转法实现的DDS，了解其原理、特点。

（2）研究DDS的基本构成，建立数学模型并在MATLAB上仿真输出波形

（3）分析DDS的噪声来源，在理论和电路上研究噪声的抑制方法。

（4）结合查表法和角度旋转提出新结构，在速度、精度上取得更好结果。

（5）完成RTL级代码编写和波形仿真，进行综合以及布局布线

在论文的组织安排上：

第一章介绍数控振荡器在重要意义，频率合成技术的两种方法，以及直接数字式频率合成器(DDS)的发展历程，规划了论文的主要工作；

第二章将详细描述DDS的技术原理、基本框架，介绍查找表压缩和角度旋转两种优化方法，提出了改进的算法，最后对误差来源进行分析；

第三章提出了改进的电路结构，分别介绍各电路模块的设计思想，配置资源策略，以及流水线在本设计的实现；

第四章主要是展示实验过程和仿真的结果，包括仿真平台搭建以及实验中总结的优化经验，最后展示时序仿真结果，并与其他论文进行结果比较；

第五章对毕业设计的工作进行简要的总结，对各项结果给予结论，并且探究改设计未来的修改方向以及介绍数控振荡器的发展趋势。

# 直接数字式频率综合(DDS)原理

直接频率合成器是一种典型的数字电路系统，是实现数控振荡器的重要方法。随着近年来DDS性能的稳步提升，学术界对DDS技术的研究仍比较活跃，因此介绍清楚DDS的基本原理很有必要性。本章将首先DDS传统结构，建立其数学模型；第二、三节介绍查找表压缩、角度旋转两种方法，用于优化系统的杂散性能和时钟频率；第四节详细分析DDS输出的频谱特性，并说明系统中噪声的来源。

## DDS的原理和框架

上世纪70年代，信号发生器的主流仍是利用锁相环（PLL）进行模拟信号的跟踪。由于大规模集成电路的出现，Tierney *et al*[3]于1971年提出直接数字合成方法，希望利用相位直接映射到指定波形。DDS的传统结构，如图1.3所示，其核心有三个部分，生成相位的累加器（PA），查表输出波形的相位-幅度转换器（PAC），以及低通的数模转换器（DAC）输出。

DDS系统输入为频率控制字(fcw)和相位控制字(pcw)，均为N位，输出正弦信号如公式(2-1)。其中fcw用于控制输出频率，pcw用于设定初始相位。

(2‑1)

公式(2-1)为连续时间表达式，离散时间表达式为(2-2)，其中是时钟周期。为离散时间的正则化相位，表达式如公式(2-3)

(2‑2)

(2‑3)

输出信号的频率如公式(2-4)，其中N是累加器的计算位数。由于在全相位空间上采样，根据奈奎斯特采样定律，离散采样频率至少为原信号带宽的两倍才能无混叠还原信号，因此对fcw的最大值有限制。同时表明输出的频率最高能达到系统时钟频率的一半。

(2‑4)

公式(2-3)表明，正则相位以固定增量fcw增长，通过累加操作可实现相位计算。由于是通过累加操作获得，在fcw抖动时也能保证了相位的连续性，规避了因相位不连续产生的高频噪声。公式(2-5)显示了输出频率的最小变化量，增加累加器位数N就可以实现非常高的频率分辨率。

(2‑5)

在系统框图2.1中，使用相位累加器(PA)计算正则相位。PA由加法器和寄存器构成。加法器输入一端为fcw，另一端与寄存器相连，加法结果则在下一时钟上升沿写入寄存器。相位生成后，下一步是将其映射到正弦幅度上。

相位-幅度转换器(PAC)是DDS中最为重要的设计，其实现对系统性能的影响最大。使用查找表进行相位-幅度转换是最常见的方式，通过记录所有相位对应的幅度值，实现非线性函数的映射。

(2‑6)

查表法的一个显著优点是实现简单，使用ROM即可完成相位-幅度的转换，在低精度要求下也可以达到较快的速度。但是当精度要求较高时，查表法必须要面临精度和速度的取舍问题。为了在全相位空间寻址，查找表的大小如公式(2-7)，其中M为输出量化位数。根据公式(2-7)，LUT的大小随N指数增长。每提升1bit相位比特数，表的大小都要翻一倍，这将急剧增加ROM的读取开销和面积开销。意味着高分辨率的DDS必须要在时钟频率上让步。

(2‑7)

系统最后通过DAC将数字信号转换为模拟信号。如图2.1所示，在满足奈奎斯特采样定律下，数字信号包含了模拟信号所有的频域信息。



图 2.1 模拟、数字信号频域示意图

观察图1.3，D/A 转换后波形还存在明显的阶梯效应。这是由于在时间离散信号相当于在时间连续信号上采样，因此相在DTFT上当于对连续信号进行周期性重叠，表达式如(2-8)所示。因此使用低通滤波器LPF实现到的转换。

(2‑8)

## 查找表压缩技术

从本章首节的理论中，DDS面临的主要问题是高分辨率和时钟频率的矛盾。要提升系统的速度，关键是提升ROM的访存速度。在实现工艺确定的情况下，一个提升ROM性能的方法是减少其地址的长度，即压缩相位。

### 对称性方法

在文献[5]中，提出了“压缩方法”。其思想是使用地址的后N-2位信息在第一象限查表，前2位用于将查表向量翻转至原有象限。任意向量均可在第一象限水平、竖直翻转至原有向量。该方法能有效减少ROM地址长度2位，使得查找表数据减少为原来的1/4。更进一步的是“压缩方法”[14]考虑到正、余弦函数的对称性，只需要使用后N-3位信息在(0, )区间同时查找正、余弦函数值。相比而言，“压缩方法”存储数据量相同但是查表地址减少了一位。

### 引入近似结构

除利用对称性压缩相位，coarse-fine结构[5]也是减小查找表大小的经典方法，如图2.2所示。



图 2.2 coarse-fine查找表结构

将相位划分成A，B，C三个部分。根据公式(2-9)，(2-10)对公式近似，可建立coarse，fine两个表，分别记录sin(A+B)和cos(A)sin(C)的值，最后将两项相加则得到近似的正弦信号。

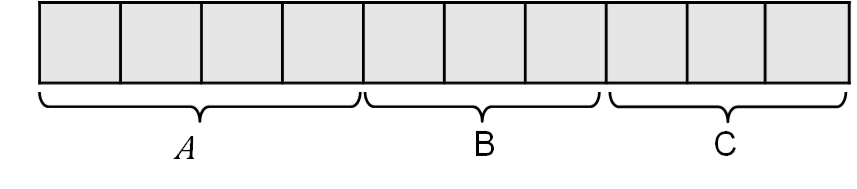


图 2.3 相位划分示意图

(2‑9)

(2‑10)

coarse-fine结构是分治法在电路设计中的体现。利用ROM大小随地址长度指数下降特点，分而治之可以从中获得收益。但仍需讨论近似的误差，当约束条件满足时，地址最多可减少为原有长度的2/3。

类似于coarse-fine结构，还有线性插值法[7]、二次插值法[8]均对查找表进行更高比例的压缩，一定程度上提升了DDS速度、功耗性能。表2.1比较了各方法的压缩比例和额外计算量

表 2.1 查找表压缩方法比较

|  |  |  |
| --- | --- | --- |
| **压缩方法** | **地址长度变化** | **额外的计算** |
| **coarse-fine** | 2/3 | 加法操作 |
| **线性插值** | 1/2 | 乘法操作 |
| **二次插值** | 1/3 | 平方操作 |

经总结，查找表压缩方法实际上是将查表复杂度转化为计算复杂度。因此查找复杂度的减少和计算复杂度的增加存在一个临界点，越过临界点后继续压缩查找表反而会增加总的时间复杂度。这将限制查找表的压缩程度。



图 2.4 压缩度-复杂度示意图

## 角度旋转方法介绍

上一节介绍了查找表压缩方法，在最后总结中提到了查找表压缩的限制。为了突破查找表的结构的局限性，Grayver [15]在1995年提出使用角度旋转的方法实现相位-幅度转换。角度旋转的思路是利用多次旋转操作逼近真实相位，而旋转操作不需要查找表，因而避免了以上查找表的缺点。

基于CORDIC算法的角度旋转方法不需要查找表结构，并且可以使用流水线进行加速。但是CORDIC算法的缺点是流水级数太长，导致捷变性能严重下降。之后的两节将具体介绍算法的原理及改进。

### CORDIC算法原理

CORDIC算法[10]是实现双曲函数族的一种算法，而三角函数是双曲函数在虚平面的一种特例。在许多论文[11]中已经对其原理进行了详细的推导，因此本文只会简单介绍生成三角函数的CORDIC算法。

对于向量，对绕原点旋转角后得到，如图2.5所示，其变换表达式如(2-11)。

(2‑11)



图 2.5 角度旋转示意图

角度可使用一组正交基表示，其中正交基，则在正交空间的坐标为。

(2‑12)

则公式(2-11)可重新表述为(2-13)。

(2‑13)

进一步得到(2-14),其中。

(2‑14)

代入得到迭代公式(2-15)，(2-16)。从迭代公式(2-14)可以看出，矩阵操作只需要移位和加法操作，在硬件中用加法器即可实现。算法通过公式(2-17)计算正交空间坐标。（2-17）使用了减法操作以及判断操作，故需要额外使用全加器和判断逻辑计算坐标。

1）迭代公式：

(2‑15)

2）初值、末值：

(2‑16)

3）正交空间坐标计算：

(2‑17)

根据之前的分析，可以得到CORDIC算法的电路框图如下：

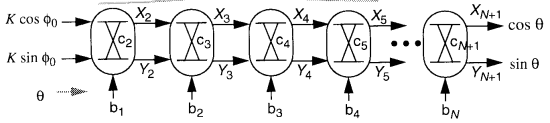


图 2.6 CORDIC算法电路框图[17]

### 混合结构

传统CORDIC算法的主要优点是通过迭代计算，避免使用了查找表，并且可以使用流水线加速。但是相比查找表法，CORDIC算法存在计算量大，流水级数长，占用资源多的情况。

为了提升角度旋转方法的性能，出现了查找表和CORDIC算法结合的方法。混合方法分两步，首先使用查找表得到粗分辨率幅度，再通过角度旋转得到细分辨率幅度。

正交基取为。此处是和原始算法之间呈关系，则公式(2-14)需修改为(2-18)。此时，相比传统CORDIC方法中多次迭代判断和加法操作，的求解的计算量大大减小了。

(2‑18)

考虑到近似条件，可使用一阶近似条件得到公式(2-20)。

(2‑19)

### 改进算法

观察公式(2-20)，旋转角度的计算中有的乘法运算，这将极大降低迭代运算的速度。为了解决此问题，文献[13, 17]均在查找表中额外保存了数据，通过查找操作替代乘法操作，减少了角度旋转的计算复杂度。顺应该思路，在本节提出了改进的算法，同样使用混合结构。

利用近似条件代入(2-13)中获得新表达式(2-21)，其中计算和(2-19)相同。注意(2-13)中已趋近于1，因此在(2-21)中被替换成常数T。

(2‑20)

为消除公式(2-20)中乘法运算，使用文献[17]的方法，在查找表中保存和两个新变量，代入到公式(2-21)中，得到迭代公式如(2-22)，

(2‑21)

另外一点需要注意的是溢出现象。由于移位操作的截断效应，公式(2-22)在局部会略微超过1。通过对常数T的约束(2-23)可以避免溢出的出现。

(2‑22)

## DDS的频谱特性和误差来源

### 频谱分析

对于理想情况，不考虑任何噪声以及输出量化，DDS输出out如公式(2-23)。对输出out进行N点FFT得到频谱如图2.7。

(2‑23)



图 2.7 理想输出信号频谱

从频谱图中可以看出，消除了矩形窗产生的频谱泄漏后，频域的能量主要集中在中心频率附近。但是现实情况中，为了简化PAC的规模，需要对正则相位进行截断，产生了相位截断误差，加上输出量化和中间过程的近似产生的误差，使得信号成为非理想信号。

图2.8展示了非理想情况下输出的频谱，在频率、相位位数相同的情况下，不仅信号的底噪提升了，而且频域上形成了周期性的spurs和多次谐波。



图 2.8 非理想输出信号频谱

### 噪声来源分析

**1) 相位截断误差**

对于一个N 位的相位，在传往PAC之前可能会截断成P 位，截断误差为，则表达式(2-23)修正为(2-24)。由近似公式可得，截断误差传导至输出的噪声。

(2‑24)

由于系统在相位空间上采样，对应的也可看作截断误差分布图2.9(a)图上采样，(b)图显示其频域为周期性线谱。由DFT理论可知，的频谱是截断误差分布频谱的平移叠加。

图2.10(a)绘制了的波形，包络位正弦信号。的频域的分布如图2.10(b)所示，形状如同为包络递减的spurs。每个spurs的频率间隔为采样频率。这也就能解释为什么图2.8中存在周期性的spurs。文献[18]中分析了对输出SFDR的影响，表达式如(2-25)，截断位数每提升一位系统SFDR将提高6.02dBc。

(2‑25)



图 2.9 相位截断误差及其频谱



图 2.10 相位截断噪声及其频谱

**2) 近似计算噪声**

传统的查表法中，相位-幅度的映射是不存在误差的。如2.2节所述，查找表压缩方法中的二阶近似将导致误差的产生。同样，本文的改进算法中由于近似条件的使用，也引入了噪声。

略去高阶分量后得到误差表达式如公式(2-26)。可以发现噪声中存在单频分量，相当于对信号进行放大，并不会影响频谱特性，因而也可略去。最终噪声的表达式为(2-27)。

(2‑26)

(2‑27)

的频谱如前所述，故近似噪声和相位截断噪声类似，也会产生多个周期性的spurs。

**3) 输出量化噪声**

在输出时进行量化会直接引入噪声，根据统计规律，一般认为量化噪声为高斯白噪声，只对频谱的底噪有影响。根据文献[18]的推导，量化噪声对系统SNR的影响如(2-28)所示，量化位数M每提升1bit，系统SNR提升6.02 dB。

(2‑28)



图 2.11 量化噪声频谱

如图2.11所示，量化噪声频谱基本上呈现高斯白的特点，但是由于精度有限还是在部分频率上存在spurs。

# 基于直接数字式的振荡器设计

## ROM-CORDIC 混合结构设计

直接数字合成器在频率分辨率、捷变性能上取得较大成功，但较锁相环式振荡器，传统DDS仍存在时钟频率受限，输出杂散性能不佳等情况。

为了提高DDS的时钟频率和杂散性能，本文采用了2.3.3节中使用的改进算法，将查找表和角度旋转法结合。此举进一步减少了查找表的寻址深度，较传统CORDIC结构所用资源也更省，使系统能以更快的时钟频率获得非常高的精度。

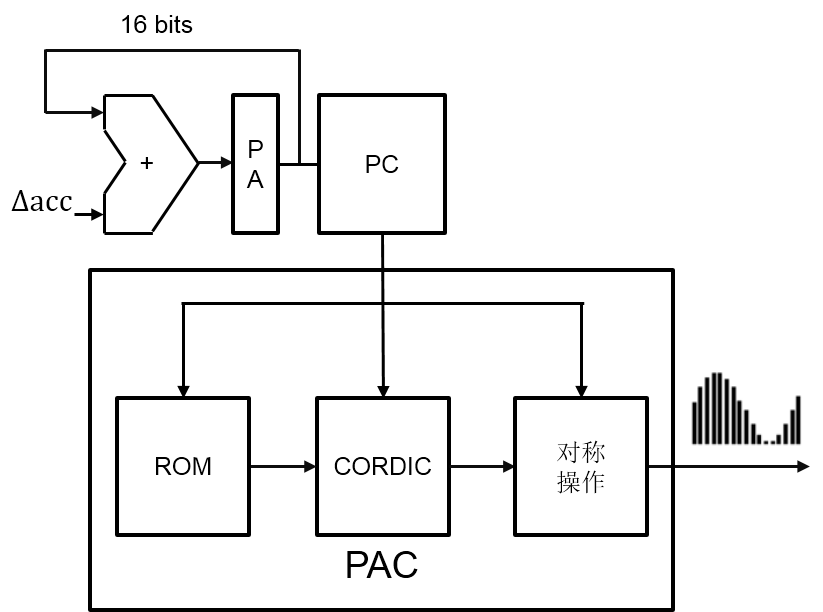


图 3.1 本文使用的DDS结构框图

图3.1显示了DDS的系统框图。它的输入为16 bit的频率控制字(fcw)和相位控制字(pcw)，用于更改系统输出的正弦波频率和初始相位。相位累加器(PA)根据fcw和pcw生成16 bit 的正则相位。

相位压缩模块(PC)可使得系统用(0, 1/4)区间的正、余弦函数值表示(0, 2)的值域。PC提取出的前3 bit作为对称操作控制字(mcw)，后13bit作为伪相位 输出至PAC, 用于查找表寻址和控制旋转角度。

相位-幅度累加器(PAC)包括384-byte的查找表和7组角度旋转电路。的1~6位用于查找表寻址，获得粗分辨率信号。的7~13位对粗分辨率信号进行旋转修正，获得细分辨率信号。

对称操作模块将细分辨率信号从(0,1/4)对称到(0, 2)区间，最后输出。

### 相位累加器(PA)设计

PA接收输入的fcw和pcw，输出线性增长的正则相位，其表达式如(2-3)。图3.2所示，PA由一个16 bit 加法器和16 bit 寄存器构成，输出也为16 bit。



图 3.2 相位累加器框图

根据公式(2-4)和(2-5)，输出信号的频率为，频率分辨率为 ，可见要加强频率分辨率，首先需提高加法器位数。

pcw的作用是在系统重置后对相位赋初值。重置一旦结束，相位初值就不受pcw的影响，故不能随时更改相位。

### 相位压缩器(PC)设计



图 3.3 相位分配结构

图3.3展示了数字相位的区域：其中1 ~ 3位用于对称操作， 4 ~ 9位用于查找表寻址，10 ~ 16用于角度旋转。为了将任意相角的向量映射到(0, 1/4 pi)区间，还需要对上述相位进行变换。

相位的前3位表示向量所在的8分区间，利用区间信息可将该向量对折至(0, 1/4 pi)区间。因此直接取1~3位作为对称操作控制字(scw)，输出至对称操作模块。为计算出伪相角，需利用的4~16位。根据对称关系，得到表达式(3-1)

(3‑1)

的前6位作为查找表地址(address)，后7位作为旋转控制字(rcw)，分别输出至ROM 和旋转电路用于确定细分辨率信号。

(3‑2)

### 相位幅度转换器(PAC)设计

**)A 查找表**

查找表是将相位转化成幅度的重要模块，其作用是输出粗分辨率的信号。

查找表需要输出 。与参考文献[12, 13]相似，查找表额外存储了两组函数，以减少乘法操作。

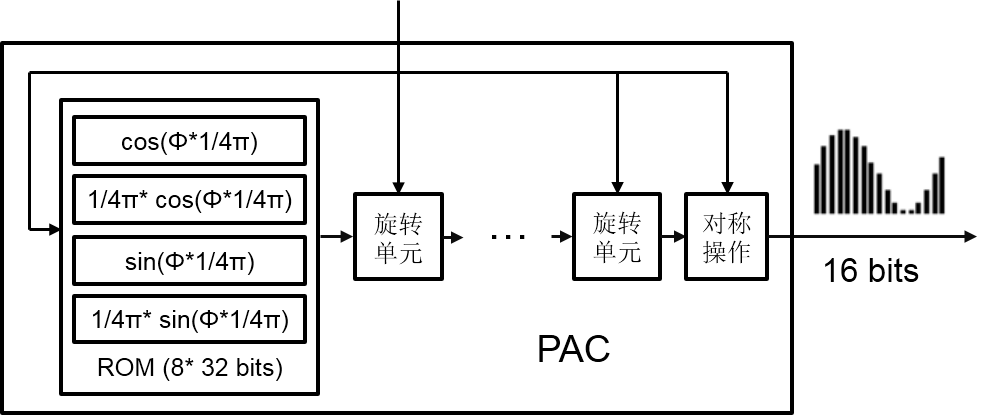


图 3.4 相位-幅度转换器(PAC)结构

根据公式(2-22)，为了防止溢出，实际写入值是。具体设计中，的长度为16 bit，而量化位数为8bit，因此查找表的输出带宽为48 bit。

**)B 旋转电路**

如图3.6，角度旋转电路接收查找表输出和旋转控制字(rcw)，最终将向量旋转至修正值。根据2.4.2中的误差分析，角度旋转将带来杂散性能的增益，rcw每提升一位SFDR提升6.02dBc。

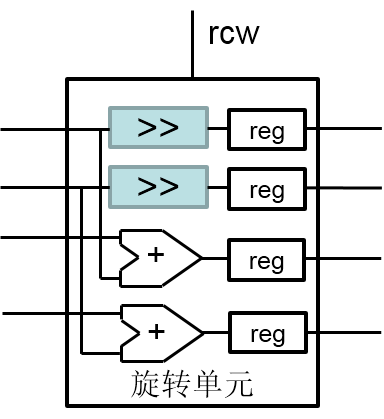


图 3.5 旋转单元电路结构

改进CORDIC算法由多次矩阵乘法迭代产生的，公式(2-21)中的矩阵乘法操作包括两次加法和两次移位，因此在设计旋转电路时，选取两个移位器和两个16 bit加法器构成旋转单元电路。迭代操作可由多个单元电路级联构成，经过7级级联，最终得到(0, 1/4)区间的向量。

**)C 对称操作模块**

(0, 1/4)区间向量需要经过对称操作才能映射到(0, 2)向量。对称操作的逻辑表达式如图3.7，。从公式中看出，利用多路选择器和反相器即可实现对称输出模块。



图 3.6 对称输出逻辑

图3.7展示了“压缩方法”的对称操作，“压缩方法”类似。在查表、角度旋转得到细分辨率信号后，使用rcw控制对称的位置得到输出信号。

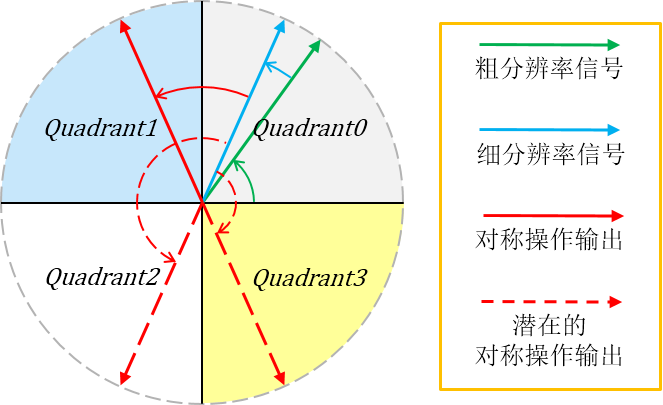


图 3.7 对称操作示意图

## 资源配置分析

1）与量化位数分配

公式(2-21)中有两组变量，其中值域为(0,)，值域为(0, 1)。导致只有前几位没有值。为了方便计算，将替换成()，使得值域由{1, 0}变成了{1, -1}，方法如公式(3-3)。最终只有后N-M-2位有数值。

具体在我的设计中，使用16位保存，而只需要8位保存。

(3‑3)

2）address与rcw位数分配

实验中发现，当约束条件(3-5)不满足时，系统的SFDR，SNR将发生骤降。

为了避免此现象，需要斟酌address与rcw的位数分配。因为的位数是13，那么查找表和旋转电路可分配的为数也为13，故取address位长为6，旋转控制位(rcw)为7位。如此分配能最大程度压缩查找表的大小，从而提升系统的时钟频率，但是会增加流水级数。

## 流水线设计

在之前的章节中，已经提到过CORDIC算法可使用流水线加速，根据迭代公式(2-21)，可使用级联的旋转单元电路实现迭代操作。要实现流水线结构，一个简单的想法是在各级单元电路之间插入触发器，这使得级联电路的时延降低至每个旋转电路和触发器的时延。

由于旋转单元电路使用rcw控制旋转方向，如果旋转方向在时间上不能对应的话，将无法输出正确的结果，因此控制信号也要层层缓存。最终的结构示意图如3.8，各级旋转单元之间插入了D触发器，用于缓存输出数据作为下一级的输入。同样PC输出的rcw控制信号也经过了多级流水，保证了控制信号和操作数据时间上的一致。

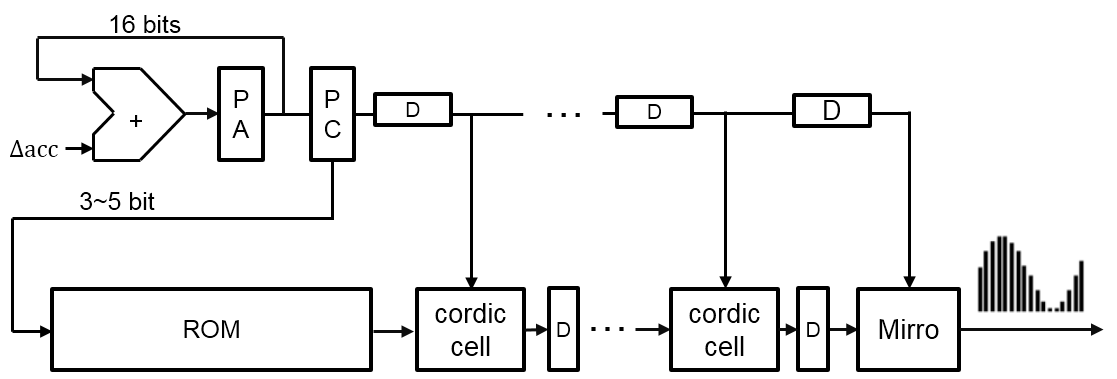


图 3.8 流水线结构

另外，加法器也可使用流水线加速，方案如图3.9所示。在文献[7]中也通过优化得到了更快的加法器。但是本次设计中，加法器流水对系统的时钟频率提升比较小，但是面积功耗会大大提升，综合考虑最终没有使用流水线加法器。



图 3.9 16 bits流水线加法器原理图

# 数控振荡器实现和仿真

基于第3章中提出的ROM-CORDIC混合电路结构设计，本章主要介绍两个实验平台的搭建，综合、布局布线后的时序仿真结果的展示，得到该设计的最终指标并且同前人工作进行比较。

## 功能性仿真平台搭建

本文最基本的功能性仿真使用MATLAB和modelsim实现。

本设计在MATLAB上搭建了二进制计算平台，并在二进制平台上完成了实验的数值仿真。另一项工作是将MATLAB平台上的架构映射成Verilog代码，并在modelsim波形仿真工具进行验证。结果显示，两平台的在输入相同时，输出波形也完全一致。



图 4.1 仿真平台结构图

最后实现了两平台的联合，如图4.1所示。MATLAB生成查找表数据通过文件写入到modelsim的ROM中，将其波形数据导回MATLAB进行频谱分析，构成计算闭环。本节将对MATLAB以及modelsim的平台搭建进行详细说明。

### MATLAB数值计算平台仿真

利用MATLAB进行仿真的突出优点是很方便查看所有内部信号的数值，非常适合进行功能性仿真。但MATLAB中的变量类型是默认是浮点型，而数字电路设计中均使用二进制码进行运算，因此需要在MATLAB上重新搭建二进制计算平台。

在MATLAB数值计算平台上，进行的验证包括：正则相位验证、查找表验证，CORDIC旋转验证。分别在以下具体说明。

（1）正则相位验证

在第三章中已经提到，正则相位由累加器生成，故对相位的验证即验证二进制加法器的正确性。经验证，生成相位均等步长增长无误。

（2）查找表验证

二进制查找表的实现方法实际是使用矩阵保存数据，如图4.2(a)，利用6位地址输出查找矩阵的一行。为了验证查找表方法的正确性，分析粗精度信号的频谱，以验证其精度是否达标。



图 4.2 粗、细分辨率信号频谱

由于粗精度信号仅使用了前9位相位，根据公式(2-25),理论最低SDFR如下所示：的频谱中SFDR值达到54 dBc，高于理论最低值4 dBc，可见查找表方法确实达到理论要求。

(4‑1)

（3）角度旋转算法验证

角度旋转电路结构在3.1.3中已经介绍过，由7级单元电路级联而成，每个单元内部包括两个加法器和两个移位器。根据(1)和(2)的验证，可以认为角度旋转的输入和旋转控制字(rcw)无误，那么只需要分析输出波形，即可验证角度旋转的正确性。

输出波形的频谱如图4.2(b)，可见其SFDR值由54dBc提升至100 dBc，提升为46dBc。根据公式(2-25)，角度旋转修正的SFDR提升值为42.14 dBc，而且考虑到多次旋转中存在移位操作，有限字长效应产生了新的误差，故该提升也超出了最低要求。

### modelsim仿真平台搭建

基于二进制的MATLAB计算平台的算法，本次毕设项目完成了对应Verilog代码的编写，并在modelsim建立了testbench，给出fcw，pcw作为系统输入，并完成ROM数据接口方便文件信息写入。



图 4.3 modelsim RTL级波形仿真

RTL波形仿真如图4.3，将波形结果进行分析，结果和MATLAB二进制计算结果完全相同，表明RTL代码和MATLAB代码具有同一性。

## 关键路径优化方法

本节介绍的实验过程中提高性能的两种手段，包括结构规范合流水线。

### 电路结构规范

由于综合工具对次关键路径缺乏关注，导致多条关键路径的出现。为了解决这个问题，首先需要做的是规范电路的结构，使代码能被更高效综合。在实践中总结了3条基本规则：优化模块划分，实现资源共享和替代电路的使用。

(1) 优化模块划分

优化模块的划分观点之一是，最好能在每一个模块末端加入触发器，避免不同模块通过组合逻辑连接，如图4.4所示。这样能减少模块间接口的连接，而且更易于控制关键路径的长度。



图 4.4 模块划分优化

建立在以上规范，优化划分的第二点是平衡各模块的组合逻辑规模，尽量使得模块内关键路劲长度相似。发现某一个关键路径上的组合逻辑比较大时，尽量将其分成两个部分，并在中间插入触发器。

(2) 实现资源共享

对于同样的真值表，可以存在多种表述，也对应不同的电路复杂度。在设计中应尽量使得资源能被重复利用，用最简单的电路结构实现相同的功能。实践中，我在共享一个加法器后，关键路径取得了比较大的减少。

(3) 替代电路的使用

对于同一个表达式，也能用不同的电路实现。例如if else语句和case语句在部分时刻就存在替代关系。根据参考资料(插入引用)的介绍，对于判断条件不变的情况，使用case 电路较if else电路更快。使用case替换if else后，关键路径也得到不小的优化。



图 4.5 case和if else 电路替代

### 关键模块重新设计

规范电路内部结构对综合结果起到了正面作用，但是关键路径本身的问题还没有解决。这时就需要重新设计关键模块，使它变得更快。

例如本文的关键路径是查找表。在前期工作中，为了方便测试使用RAM记录查找表。为了提升查找表的速度，改用ROM查表，用组合逻辑将ROM输出固定后，综合后查找表的延时大大降低了，最终时钟频率可运行至2 GHz。

在最终的设计方案中，旋转单元电路中的加法器成关键路径所在。我使用流水线加速器以减少加法操作的时间，将时钟频率提升至2.6 GHz，但是如3.3所述，该方法对面积功耗的要求大大提升了。

## 时序仿真结果

功能性仿真的用途是验证逻辑的正确性，但是没有考虑到实际的物理条件。门级电路的延时和布线的延时的差异会引起信号时序的不一致，一旦超出时钟频率约束将导致结果错误。本节的内容首先介绍综合和布局布线的工具，然后展示时序仿真的结果。

### 工具介绍

A）Design Compiler

Design Compiler(DC)是一款综合工具，主要功能是将Verilog代码映射到门级电路，并进行优化。综合的步骤主要包括三个阶段：转换、映射、优化。

首先将HDL代码转化成工艺无关的RTL级网表，然后指定工艺库将RTL级网表映射到前端门级网表上，最后在约束条件下对前端门级网表进行优化。最终的网表中就包含了各门级电路的时间信息，即可进行时序仿真。但是在综合过程中时序中连线延时、时钟扭斜等无法确定。

B）IC Compiler

IC Compiler(ICC)是一款布局布线工具，除了对电路进行布局布线外，ICC还提供时钟树综合、IO和pad的布局等作用。ICC不仅考虑了原件之间连线的延时、时钟不同步等问题，同时提取金属器件之间的寄生电容，使得结果和实际情况更加一致。

### 结果展示

A）DC结果

根据之前的设计进行了DC综合，时钟设置为2.0 GHz仍可满足时序约束条件，见图4.6。图4.7主要展示三个部分，包括输出信号sin\_amp的时序，查找表查找的时序，以及旋转单元内部加法器的时序。



图 4.6 前端时间约束

图4.7主要展示三个部分，包括输出信号sin\_amp的时序，查找表查找的时序，以及旋转单元内部加法器的时序。



图 4.7 前端时序波形图

观察发现，输出信号sin\_amp的滞后大约半个周期，属于正常范围。查找表的输入为A，输出为Q，从信号对应关系看来，查找输出要滞后1个周期。观察黄框标出的旋转单元加法器时序，输出的滞后和抖动超过了3/4个周期，有可能导致超出约束的风险。

B）ICC结果

使用synopsis的ICC对前端网表进行布局布线，得到了后端网表、相关时序信息，最终的版图如图4.8所示。

图4.9和图4.10展示了ICC的时序约束和时序仿真结果。从图4.9看出，电路在0.6 ns(1.7 GHz)时约束得到满足，相比DC综合结果稍慢0.1ns。



图 4.8 后端设计版图



图 4.9 后端时序约束

图4.10中注意到查找表输出Q部分值为蓝色状态，这是因为经过ICC优化后无效连线在物理上被省略了。另外图4.10中黄框标出的加法器延时依然比较大，进一步说明了加法器成为限制电路的瓶颈。



图 4.10 后端时序波形仿真

## 性能比较

本文实现的DDS式数控振荡器使用了65 nm TSMC工艺库，输入为16 bits频率控制字，经ICC后端在1.7GHz时钟下可输出频率为2.6kHz~0.85 GHz的正余弦波。输出的量化位数为16 bits，理想输出信号的SFDR为100 dBc，SNR为89 dB。电路设计中，使用1个16位加法器使用了一块6位地址48位字长的ROM和7组旋转单元电路。最终功耗23.4mW。

表 4.1 性能汇总和比较

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **技术**  **方法** | **CMOS**  **工艺** | **时钟**  **[MHz]** | **相位截断**  **[bits]** | **SFDR**  **[dBc]** | **功耗**  **[mW]** |
| **2007 JSSC[14]** | Hybrid-CORDIC | 0.25um |  | 16 | 90 | 154 |
| **2011 JSSC[13]** | Excess-four CORDIC | 0.18um | 260 | 20 | 113 | 16.5 |
| **2014 ISCAS[17]** | Multiplier CORDIC | FPGA | 1000 | 20 | 120 | 54.9 |
| **本设计** | ROM-CORDIC | 65 nm | 1700 | 16 | 100 | 23.4 |

表4.1对近年来DDS的研究进行了总结和比较。使用本设计的仿真结果[[2]](#footnote-2)进行对比，ROM-CORDIC结构在时钟频率、SFDR、功耗上均表现优异，综合性能处于领先水平。

# 结论

## 主要工作总结

直接数字合成(DDS)作为实现数控振荡器的重要方式，随着近年来性能的提升，其使用正变得越来越流行。当前的研究中，查找表结构中的固有问题依然没有彻底解决，导致DDS的用途还受到限制。

本文提出了ROM-CORDIC混合结构，使用角度旋转修正减少了查找表的地址长度，通过并行的ROM省略了旋转中的乘法操作。论文中提出的方法解决了直接数字式结构中的部分问题，分析了输出的误差来源，并且利用MATLAB和modelsim搭建了基础的功能验证平台。根据误差理论指导，对系统结构进行了定量的分析。在电路设计过程中实现了电路的流水线化，最终经过综合、布局布线，系统运行的时钟频率为1.7GHz，输出最大杂散(SFDR)达到100 dBc。相比文献[13]使用FPGA实现的DDS，芯片的整体功耗仅为其42%，但是时钟频率为其170%、ROM大小仅为其25%左右。鉴于在ICC中布局布线的优化尚未展开，电路的时钟频率还有再进一步的空间。

## 未来工作展望

本科毕设期间，由于时间、人力、视野的不充分，使得最终的结果还达不到尽善尽美的标准。主要改进的空间还有方向、算法、电路、工具等。

(1)未来应拓宽视野，跳出传统的结构，观察是否有更高效的结构存在。

(2)在此框架下实现更多算法。因为时间、进程的限制，目前只实现了少数几种算法，新的想法还存在理论中但没有仿真实现。未来可以参考现在的流程对现有算法、框架改进，并与现在的结果比较。

(3)电路上提出创新在未来有广泛的空间。目前使用的都是标准工艺库内的器件，没有提出更适应系统的新电路。如果能对加法器、存储器或者旋转单元等电路上改进，将是巨大的创新。

(4)加强工具的使用将有助于继续提高结果。在针对综合进行基础的优化后得结果得到了提升，应该还有挖掘空间。另外布局布线还没有进行优化，导致后仿结果和前防结果差距比较大，未来熟悉ICC等工具的使用后，应该能使后仿结果更加迫近DC综合结果。

插图索引

表格索引

参考文献

[1] RolandE.Best. 锁相环设计、仿真和应用[M]. 清华大学出版社, 2004.

[2] Zhao Y, Chen Z Z, Du Y, et al. A 0.56 THz Phase-Locked Frequency Synthesizer in 65 nm CMOS Technology[J]. IEEE Journal of Solid-State Circuits, 2016, 51(12):3005-3019.

[3] Tierney J, Rader C M, Gold B. A Digital Frequency Synthesizer[J]. IEEE Transactions on Audio & Electroacoustics, 1971, 19(1):48-57.

[4] Genovese M, Napoli E, Caro D D, et al. Analysis and comparison of Direct Digital Frequency Synthesizers implemented on FPGA[J]. Integration the Vlsi Journal, 2014, 47(2):261-271.

[5] Sunderland D A, Strauch R A, Wharfield S S, et al. CMOS/SOS frequency synthesizer LSI circuit for spread spectrum communications[J]. IEEE Journal of Solid-State Circuits, 2010, 19(4):497-506.

[6] Nicholas H T, Samueli H. A 150-MHz Direct Digital Frequency Synthesizer In 1.25/spl mu/m CMOS With -90dBc Spurious Performance[C]// Solid-State Circuits Conference, 1991. Digest of Technical Papers. Isscc. 1991 IEEE International. IEEE, 1991:42-286.

[7] Langlois J M P, Al-Khalili D. Novel approach to the design of direct digital frequency synthesizers based on linear interpolation[J]. IEEE Transactions on Circuits & Systems II Analog & Digital Signal Processing, 2003, 50(9):567-578.

[8] Ashrafi A, Adhami R, Milenkovic A. A Direct Digital Frequency Synthesizer Based on the Quasi-Linear Interpolation Method[J]. IEEE Transactions on Circuits & Systems I Regular Papers, 2005, 57(4):863-872.

[9] Caro D D, Petra N, Strollo A G M. Direct Digital Frequency Synthesizer Using Nonuniform Piecewise-Linear Approximation[J]. Circuits & Systems I Regular Papers IEEE Transactions on, 2011, 58(10):2409-2419.

[10] Volder J E. The CORDIC Trigonometric Computing Technique[J]. Institute of Radio Engineers Transactions on Electronic Computers, 1959, EC-8(3):330-334.

[11] Mortezapour S, Lee E K F. Design of low-power ROM-less direct digital frequency synthesizer using nonlinear digital-to-analog converter[J]. 1999, SC-34(10):1350-1359.

[12] Yoo T, Hong C Y, Jung Y H, et al. A 2 GHz 130 mW Direct-Digital Frequency Synthesizer With a Nonlinear DAC in 55 nm CMOS[J]. IEEE Journal of Solid-State Circuits, 2014, 49(12):2976-2989.

[13] Willson A, Ojha M, Agarwal S, et al. A direct digital frequency synthesizer with minimized tuning latency of 12ns[C]// IEEE International Solid-State Circuits Conference. IEEE, 2011:138-140.

[14] Caro D D, Petra N, Strollo A G M. A 380 MHz Direct Digital Synthesizer/Mixer With Hybrid CORDIC Architecture in 0.25 um CMOS[J]. IEEE Journal of Solid-State Circuits, 2006, 42(1):151-160.

[15] Grayver E, Daneshrad B. Direct digital frequency synthesis using a modified CORDIC[C]// IEEE International Symposium on Circuits and Systems. IEEE, 1998:241-244 vol.5.

[16] Madisetti A, Kwentus A Y, Willson A N. A 100-MHz, 16-b, direct digital frequency synthesizer with a 100-dBc spurious-free dynamic range[J]. IEEE Journal of Solid-State Circuits, 2002, 34(8):1034-1043.

[17] Bergeron M, Willson A N. A 1-GHz direct digital frequency synthesizer in an FPGA[C]// IEEE International Symposium on Circuits and Systems. IEEE, 2014:329-332.

[18] O'Leary P, Maloberti F. A direct-digital synthesizer with improved spectral performance[J]. IEEE Transactions on Communications, 1991, 39(7):1046-1048.

[8] Yoo T, Hong C Y, Jung Y H, et al. A 2 GHz 130 mW Direct-Digital Frequency Synthesizer With a Nonlinear DAC in 55 nm CMOS[J]. IEEE Journal of Solid-State Circuits, 2014, 49(12):2976-2989.

[9] Caro D D, Petra N, Strollo A G M. A 380 MHz Direct Digital Synthesizer/Mixer With Hybrid CORDIC Architecture in 0.25 um CMOS[J]. IEEE Journal of Solid-State Circuits, 2006, 42(1):151-160.

[10] Essenwanger K A, Reinhardt V S. Sine output DDSs. A survey of the state of the art[J]. 1998:370-378.

[11] Madisetti A, Kwentus A Y, Willson A N. A 100-MHz, 16-b, direct digital frequency synthesizer with a 100-dBc spurious-free dynamic range[J]. IEEE Journal of Solid-State Circuits, 2002, 34(8):1034-1043.

[12] Volder J E. The CORDIC Trigonometric Computing Technique[J]. Ire Transactions on Electronic Computers, 2009, EC-8(3):330-334.

[13] Willson A, Ojha M, Agarwal S, et al. A direct digital frequency synthesizer with minimized tuning latency of 12ns[C]// IEEE International Solid-State Circuits Conference. IEEE, 2011:138-140.

[14] Bergeron M, Willson A N. A 1-GHz direct digital frequency synthesizer in an FPGA[C]// IEEE International Symposium on Circuits and Systems. IEEE, 2014:329-332.

[15] O'Leary P, Maloberti F. A direct-digital synthesizer with improved spectral performance[J]. IEEE Transactions on Communications, 1991, 39(7):1046-1048.

致 谢

附录A 外文资料调研阅读报告或书面翻译

FPAG实现的1 GHz频率合成器

摘要

本文介绍了一种使用FPGA实现的高速直接数字频率合成器（DDFS）的架构与设计。该架构使用的角度旋转算法基于一种新型乘法器，使的正弦和余弦输出的幅度不会失真。这种算法利用DSP在FPGA上效果很好。最终的电路运行在赛灵思Virtex-7上，在时钟为1 GHz时功耗仅为54.9 mW，和使用ASIC实现的性能相当。

1 引言

直接数字频率合成器(DDFS)提出以来，它就已经被用于数字通信系统中。DDFS具有很多特性使得它由于传统的模拟PLL，这些特性包括相位连续，快速的频率切换，精细的频率分辨率，广泛的频率范围和频谱纯度。

大多数DDFS都是基于Tierney，Rader和Gold在1971年首先提出的结构，如图5.1所示。该结构由两个主要部分组成：相位累加器和正弦/余弦映射函数（SCMF）。相位累加器由M位加法器和寄存器组成。频率控制字（fcw）控制累加速率，从而控制输出频率。相位累加器的M位输出在输出SCMF之前，会被截断为W位，以产生φ。SCMF的过去由查找表组成，以将φ映射到输出X和Y，其中X =cosπφ和Y =sinπφ。查找表大小与M呈指数比例关系。因此，在过去40年中为减少查找表的大小已经进行了大量研究。

FPGA在数字电路领域越来越受欢迎，尽管相对于ASIC设计而言，在相同的工艺下时钟速度和功耗开销都处于劣势。由于成本高昂的NRE，公司和研究机构可能要在不同时代的工艺中设计ASIC，而FPGA在这一方面更具吸引力。

不断升级的无线标准正将无线基站从固定实现推向可重构方式，而这种技术通常被称为软件定义无线电。实现可重构的基站的一种方法是在设计中使用FPGA，使得无线标准的发展中，已部署的硬件可重新编程。社会对无线数据的需求越来越大，导致对每个小区基站需要的更多无线容量，基站也要变得更小和更节能。尽管在无线基站中使用DDFS作为数字无线电的一部分，但是很少有工作来优化FPGA中DDFS的实现和功能。这样的数字无线电实施方案倾向于使用提供的供应商提供的DDFS内核，内核中的DDFS为了达到指定的频谱纯度，其查找表结构中使用了功耗较大的RAM和复杂的乘法器。在本文的工作中，我们提出了FPGA中DDFS的最佳实现，主要设计目标是提高比特率和降低功耗。

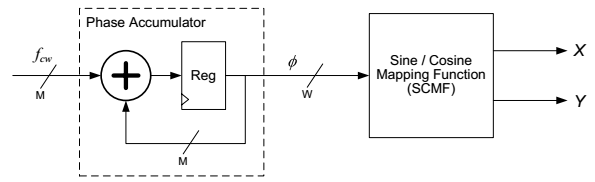
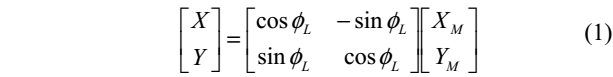


图 5.1 DDFS的基本结构

2 结构

A. Coarse-fine DDFS

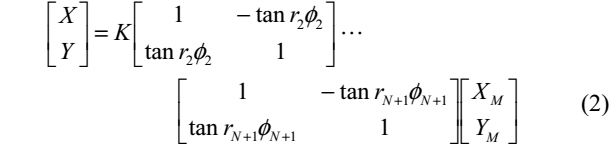
多年来，基于coarse-fine结构的DDFS已经出现了有多个改进版本。 在这种结构中，相位角φ被分解为coarse角度φ\_M和fine角度φ\_L。这允许DDFS分两阶段实现，其中coarse阶段通常在查找表中实现，而fine阶段通过角度旋转，线性插值或多项式插值实现。如下的旋转方式允许我们将DDFS分解成这两个阶段，其中(X\_M,Y\_M)是查找表的输出。



Coarse-fine DDFS中的关键组件通常是fine的部分。本文提出的DDFS的基础架构是基于不会引入幅度失真的角度旋转算法。

B. 角度旋转算法

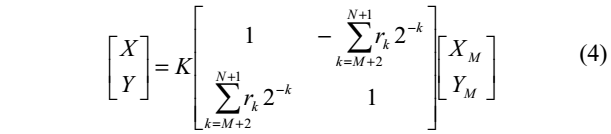
Madisetti使用CORDIC概念来提出了一个角度旋转算法，当该点围绕单位圆旋转时，该角度旋转算法不会使(X,Y)旋转后的向量发生扭曲。



参见(1)中的余弦函数，可被分解为多个子旋转如公式(2)。式中K = cosφ2cosφ3... cosφN+ 1是因式分解结果的比例因子，r\_k∈{-1,1}。其中r通过设置初始相位φ\_F和对φ的b个二进制位重新编码得到。



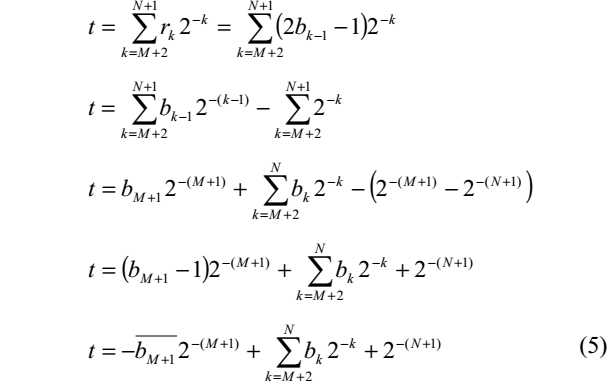
对于足够大的k，tan 2^(-k)≈2^(-k)是可近似的，对于有限信号字长(SWL），它们是完全相等的。Madisetti还表明，当条件k≥(SWL – 1)/ 2满足时，多次旋转可以合并成一次旋转。我们将φ分解为φ\_M和φ\_L，并满足合并条件，使得φ\_L的MSB位于2^-（M + 1）处。最终，次旋转可以重新组合成以下单次旋转，公式如下：



K表示缩放比例，初始相位为φ\_F，(X\_M,Y\_M)是coarse阶段的输出。

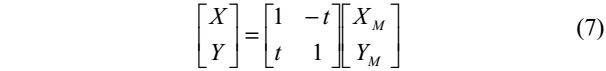
C. 两阶段重新编码

(4)中的求和项表示带有二进制数字r\_k的二进制数。 我们可以通过用b\_k表示r\_k的值，将r\_k重新编码成二进制补码二进制数t\_k∈{0,1}的位。得到公式(5)。





使用公式(5)和(6)和K，对公式(4)改写得到(7)。这些操作使得fine阶段旋转只剩下一次乘法操作。



D. Coarse 阶段

3 片上DNN处理器

A.3.1 整体结构

图A-3展示了片上DNN处理器的硬件结构。16b的RISC是主控制器。有8个PE阵列和200个乘法器单元来实现8个图像块的卷积。每个PE阵列有5x5大小的特征。PE中的加法树在一个周期内把5x5的结果加起来。为了支持最大值合并（pooling）操作，有一个由8个PE阵列共享的最大值ALU。三级加法树支持用同样的硬件实现图A-2（a）中的CNN和MLP操作，实现了高能效。

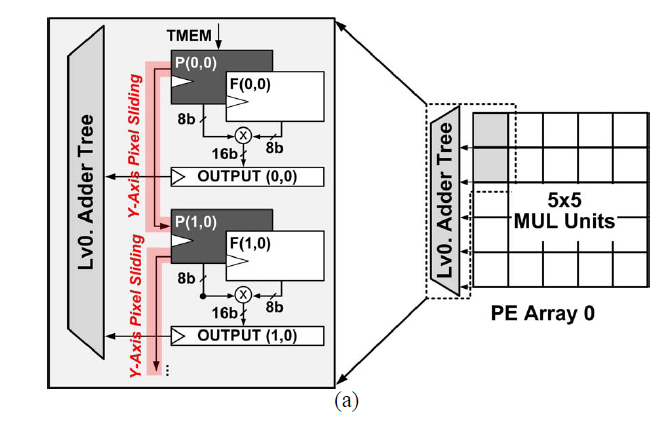
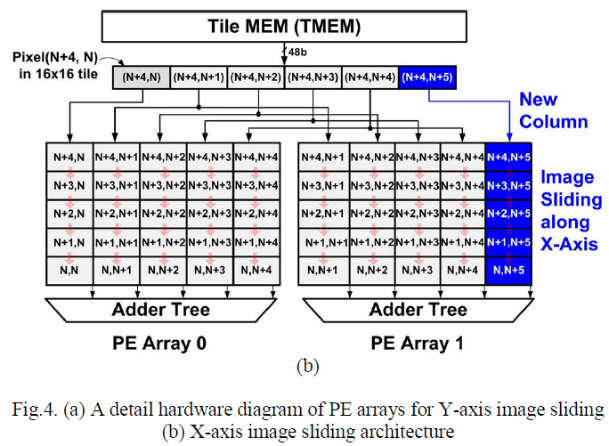
 有3个SRAM存储相关的DNN数据。图像块存储TMEM存储多个16x16的输入图像块，而8KB特征存储RMEM存储所有的离线训练的5x5特征集。因为提出的DNN用了稀疏编码，8kB的特征集可以保证目标之别的精度。8KB的权重存储WMEM存储用来计算MLP的权重。片上DNN处理器的网络接口用通用协议读取16x16的图像块输入。而且发送最后选择的16x16的可能存在目标区域块给其他的IP核，用于进一步的目标识别。

图 A-4 (a)沿Y方向图像滑动的PE阵列的硬件细节流图 (b)X方向图像滑动结构

A.3.2二维图像滑动窗结构

图A-4(a)和图A-4(b)显示了PE阵列的细节结构，支持二维的图像滑动窗，能够快速载入5x5的图像块。在y轴滑动时，一个8b的像素寄存器连接到相邻的靠近y轴的像素寄存器，如图A-4（a）。因此，只有第一行的5x5单元需要从TMEM中读取像素，减少了图像块载入的消耗。图A-4（b）显示了图像滑动窗沿x轴滑动的情况。当卷积操作结束之后，48b的6个像素数据从TMEM中读取出来。开始的5个像素放入PE0和PE1，剩下的一个像素放入PE1来做X轴滑动，结果是临近x轴的5x5的块可以在一个周期内放到2个PE中。

A.3.3双模式(CNN,MLP)PE阵列集成

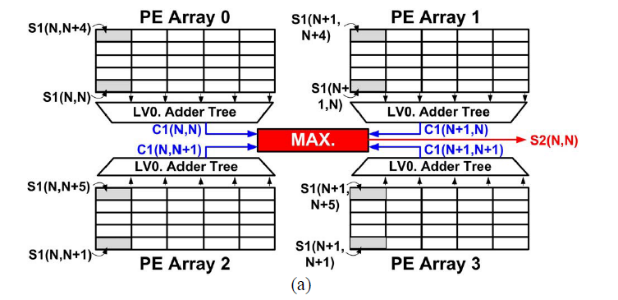
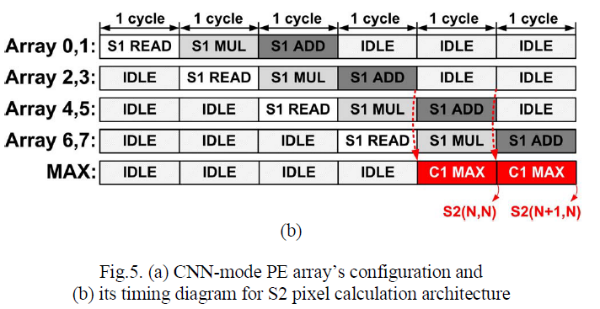
图A-5(a)显示了CNN模式的硬件集成，有4层加法器树。图A-5(a)中的4个PE阵列通过4个相邻的5\*5块并行做卷积操作。2x2的C1值从输出的4个加法器中取最大值用在S2层的计算中。

图 A-5 (a)CNN模式PE阵列的集成 (b)S2像素计算结构的时序图

图A-5(b)显示了8个PE计算两个S2的流水线的时间图表。第一个周期，阵列0和1的第一行寄存器从TMEM读入6个像素数据，其他行从上一行读入像素数据。下一周期，阵列0和1计算5x5的图像块和5x5的特征之间的乘法操作。而阵列2和3从TMEM读取像素数据。第三个周期，阵列0和1用加法器计算两个C1输入值，阵列2、3执行乘法，阵列4、5读取像素数据。在阵列2和3计算出C1数据后，从阵列0-3的4个数据求最大值计算S2层的输入。因此DNN处理器用6个周期计算2个S2像素点。

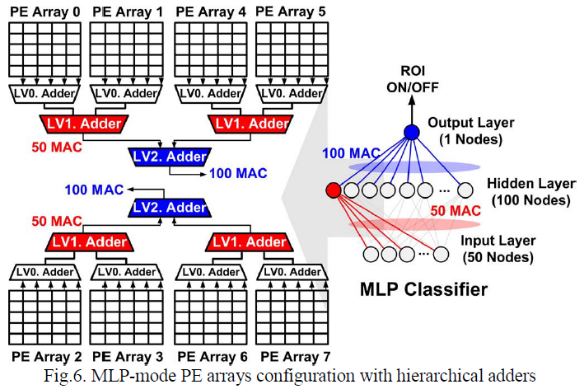
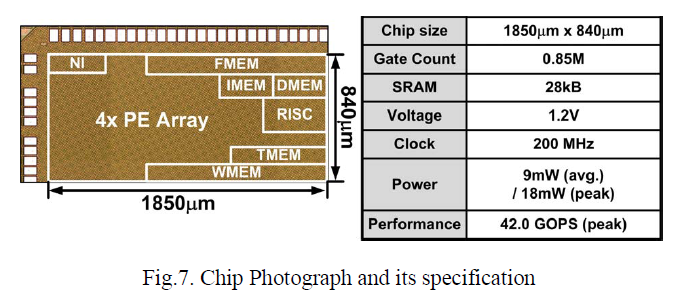
图A-6显示了MLP模式下3层加法器硬件集成。50x100x1的隐藏单层节点输出从两个PE阵列中用级联的0层和1层加法器得到。因此用8个PE可以并行得到4个隐藏节点的值。最后一层输出用4个PE和级联的0,1,2级加法器得到。因此两个输出节点可以由8个PE组成的两个MLP分类器得到。

图 A-6 芯片和具体参数

图 A-7 MLP模式含多层加法器的PE阵列的集成

A.4 完成结果

图A-7显示了片上DNN处理器，65nm工艺，0.85M NAND2等价逻辑门和28KB SRAM。总平均能耗是9mW，峰值功耗18mW，频率200MHz，性能为42.0GOPS，电压1.2V，面积1.554mm2。

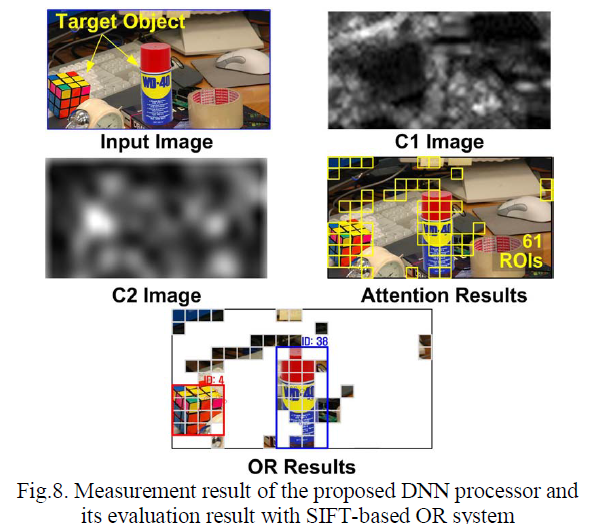
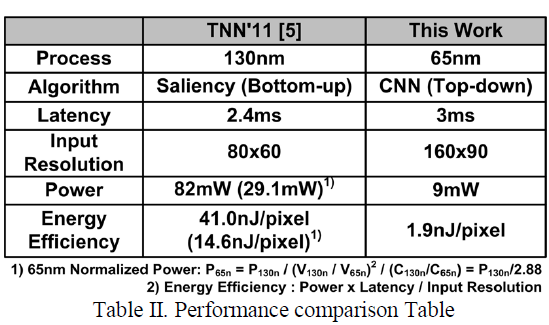
图A-8显示了DNN处理器的测量结果，目标检测评估结果集成在多核时间处理器中。C1图像显示第一个pooling层用了50个稀疏特征，第二个pooling层用了250个稀疏特征。因此只有61个可能存在目标区域被选中。我们用多核处理器和全连接的目标识别算法验证了DNN视觉感知的性能。在目标区域识别选择之后，多核处理器从61个ROI中提取SIFT特征和数据库中的100个数据图像比较。DNN处理器选出的可能存在目标区域显示在图A-8中，目标检测成功快速检测到了目标物体。因此，实现了<33ms的目标识别时间减少了80%的分块计算。

表 A-2 性能比较表

图 A-8 DNN处理器的测量结果和基于SIFT的OR系统的分析结果

表A-2展示了目前最好的视觉感知硬件和自底向上的特征视觉算法的比较。品质因数FOM考虑了延时、输入图像分辨率、能量消耗和处理器归一化参数，用来比较能量效率。本文实现了1.9nJ/像素的能量FOM，比之前的工作小7.7倍。

A.5 结论

本文提出了一种高能效的片上DNN处理器，用于移动视觉芯片的高速的自顶向下视觉感知。我们开发了基于DNN的视觉感知方法，比之前的SIFT方法减少了90%的执行时间。而且我们完成了高并行度的200路PE阵列，二维图像滑动窗结构。而且双模式PE阵列和3级加法器支持CNN和MLP，实现了7.7倍的能量消耗降低。总的来说，我们提出了适应性强且快速的DNN处理器，成功实现了实时移动视觉处理器。

1. 时钟频率达到2 GHz，SFDR=55 dBc [↑](#footnote-ref-1)
2. 这里使用的是仿真结果进行比较，和实际流片性能会存在不一致之处，但相比其他流片的设计仍存在一定优势。 [↑](#footnote-ref-2)