

文章编号: 2096-1618(2016)05-0489-05

基于 C# 的混频杂散分析软件

牟俊全, 苗智翼

(成都信息工程大学电子工程学院, 四川 成都 610225)

摘要:为了消除由混频器非线性和非理想的隔离特性等因素引起的杂散响应,需要对混频器输出信号进行滤波处理,将不需要的混频杂散信号滤除。而输出信号中的杂散响应又与输入和本振信号密切相关。因此为了减小后期电路中对滤波器的压力,需要在混频方案设计之初,合理规划变频方案。混频方案设计人员迫切需要一个能在电路设计前期对混频器所产生杂散信号进行预估计算的工具,而本软件正立意在此。软件基于 .NET Framework 框架,利用 C# 语言,根据混频杂散产生的原理编写算法,对输入及本振信号进行多阶组合计算,并利用编程框架中集成的多种控件,将计算结果以直观的图表形式进行实时显示。考虑到软件的可维护性,程序将关键功能以模块化的方式实现,并预留了相应的软件接口,为后续的功能完善提供可行性。软件在实现基本功能的同时,兼顾了易用性和可操作性。利用本软件,工程师可在方案规划前期对混频过程中产生的杂散信号进行计算,通过调整相关输入参数,可以实时、直观地获取杂散的频点、幅度以及杂散信号随频率变化的滚动趋势等信息。从多位一线研发人员的反馈中获知,软件免去了工程师繁琐却不严谨的计算,并且计算快速、准确。相较于某些现有杂散分析工具,软件提供了杂散变化趋势的呈现,且在计算结果的显示方面更加直观和人性化。

关键词:电磁场与微波技术;微波毫米波电路;变频;混频杂散;C#

中图分类号:TP311.5

文献标志码:A

0 引言

在无线电信号的收发处理过程中,经常涉及对信号进行频率变换。在频率转换的过程中,除产生希望得到的输出信号外,还不可避免地产生许多不需要的信号,称之为“杂散信号”^[1-2]。这些信号干扰了后续电路对所需信号的处理。为此,需要尽可能地对这些杂散信号进行抑制^[3]。有两种通行的做法用来消除杂散信号。

首先,可以从电路方面下手,其核心思想主要是通过采用各种平衡电路结构、模拟滤波器对杂散信号进行过滤。例如,通过采用各种单/双/三平衡结构、镜频抑制结构对相关杂散进行抑制^[4]。或如,在混频输出单元中,采用适当的带通、低通、高通滤波器对带外的杂散信号进行过滤^[5]。

上述方法通常是行之有效的,但仍有不足之处。比如,各种混频器结构往往只针对部分杂散信号有抑制效果,而对另一些杂散信号仍然束手无策^[6];又如,当杂散信号混于输出频带内时,则无法用滤波的方式对其进行抑制^[3]。这时,需要从混频方案本身入手,在方案设计之初,预见性地对杂散信号进行预估计算,

并尝试寻找一个最佳的混频方案,将输出频段最大限度地避开杂散信号,进而从理论上避免杂散对输出信号的影响。

目前,行业中普遍做法是:在方案设计前期,根据设计者经验选取若干中频频点。并根据这些频点手动计算相关频段的杂散,最终选出一个相对最优的混频方案。但这存在诸多局限:首先,设计者不同于计算机,无法对所有混频方案进行穷举,通过这种方法找出的“最优混频方案”并不一定是真正“最优”的;其次,由于理论上非线性器件会产生任意阶次的杂散,且阶数越高,幅值也就越低^[7],所以在实际工程应用默认忽略4阶以上杂散信号的影响。然而,根据实践经验,许多混频器在输入较大功率时,输出端产生的高阶杂散的相对幅值仍然十分可观。在某本振为34 GHz、输入18~26.5 GHz的下变频组件的调试过程中,发现在射频输入18~20 GHz时中频带内会出现5阶杂散。当输入-15 dBm信号时,3X2杂散抑制制度仅为20 dBc,而这在方案规划初期并未考虑到。

基于上述考量,对混频基本原理和杂散的计算方法进行了思考,基于C#语言和.NET Framework平台对算法进行了实现,并以易用、人性化的用户界面和形象、实时的数据显示形式作为软件的交互界面。欲为设计人员在工程实践中提供一定的参考。

收稿日期:2016-08-21

项目基金:国家重点基础研究发展计划(973)资助项目(2014CB441405);国家自然科学基金资助项目(41305005);四川省教育厅科技成果转化重大培育资助项目(16CZ0022)

1 混频杂散产生的原因

1.1 非线性特性的幂级数分析

对于混频器等非线性器件而言,其伏安特性可以表示为

$$i = f(u) \quad (1)$$

若输入信号为

$$u = u_0 + V_1 \cos \omega_1 t + V_2 \cos \omega_2 t \quad (2)$$

将其进行泰勒级数展开,若只考虑前三阶,经过化简整理后可得:

$$i = k_0 + k_1 \cos \omega_1 t + k_2 \cos \omega_2 t + k_3 \cos 2\omega_1 t + k_4 \cos 2\omega_2 t + k_5 \cos(\omega_1 + \omega_2)t + k_6 \cos(\omega_1 - \omega_2)t + k_7 \cos 3\omega_1 t + k_8 \cos 3\omega_2 t + k_9 \cos(2\omega_1 + \omega_2)t + k_{10} \cos(2\omega_1 - \omega_2)t + k_{11} \cos(\omega_1 + 2\omega_2)t + k_{12} \cos(\omega_1 - 2\omega_2)t \quad (3)$$

这里用 k_n 简化表示各项的系数。对于更高阶项,通过分解整理同样可以得到类似规律的结果^[2,8]。

通过分析式(3)可以发现,杂散的幅值随着阶数增加而降低,其斜率随着阶数增加而增大。换言之,两个频率为 ω_1 和 ω_2 的信号通过非线性器件以后,产生 ω_1 和 ω_2 的各阶组合频率信号

$$\omega_{m,n} = |\pm m\omega_1 \pm n\omega_2| \quad m, n = 0, 1, 2, \dots, \infty \quad (4)$$

1.2 混频器的特性

混频器是无线收发设备中的核心组件,广义的混频器就是一个理想乘法器^[2,9]。乘法器是一种特殊的非线性器件,若对其输入(2)式信号,则其输出为

$$v_{OUT} = V_1 \cos \omega_1 t \cdot V_2 \cos \omega_2 t = V_1 \cos(\omega_2 - \omega_1)t + V_2 \cos(\omega_2 + \omega_1)t \quad (5)$$

由此可知,对乘法器伏安特性的泰勒级数展开式化简之后将只剩下二阶组合频率,而其余组合频率的系数均为零^[2]。如果选取“差频”部分作为输出,则有如下频率关系:

$$F_{v_{OUT}} = F_{v_{RF}} - F_{v_{LO}} \quad (6)$$

这样,通过改变本振频率,就可以灵活地将射频信号变频到任意指定的频带内。

然而,实际中的混频器不可能对其他阶次的信号做到完全抑制。由(4)式可知,杂散信号零散地分布在频谱区域。当这些信号出现在输出通带内时,无法用滤波等手段滤除。需要做的,就是选取一个最合适的中频,使得该频段能最大限度的避开杂散信号的干扰。

2 软件思路

2.1 .Net Framework 框架和 C#语言

.Net Framework 是 Microsoft 为开发应用程序而创建的一个具有革命意义的平台。其主要包括一个庞大的代码库,可以在客户语言(如 C#语言)使用这些代码^[10-11]。 .Net Framework 使程序员不必过多地考虑底层功能的具体实现,而将主要精力用于程序的逻辑功能上。

C#是一种表现力强大的编程语言,它是为生成在 .Net Framework 上运行的多种应用程序而设计的。C#简单、功能强大、类型安全,而且是面向对象的^[10,12]。软件基于 C#语言,充分利用了面向对象语言模块化的优点,为后续软件功能的升级、更新提供了便利。

2.2 总体规划

在软件的编写过程中,强烈依赖 OOP(object oriented programming)即所谓“面向对象编程技术”^[13],将程序中各个功能以模块化方式实现,并在顶层进行模块调用。软件总体分为“算法”和“界面控制”两部分。主要的模块组成层次如图1所示,其中每个底层模块都由“类”或“结构”实现,由上层及顶层模块进行调用。

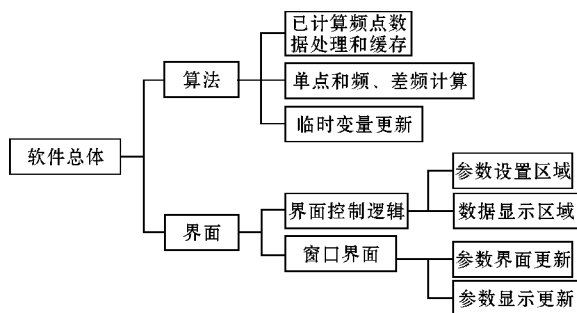


图1 程序层次模块

其中,算法部分的调用流程为:由“临时变量更新类”控制当前频点的输入信息,并交给“和差频计算类”计算当前频点所产生的和、差频。随后“数据处理和缓存类”将“和差频计算类”提供的当前频点的和差频数据进行汇总和格式处理,并提供最终的显示数据以“字段”或“属性”的方式提供给“数据显示”部分调用;界面部分主要由“界面控制逻辑”和“窗口界面”两部分组成,前者负责软件界面更新的逻辑,后者包括了窗口、按钮选项等界面信息。

2.3 固定本振情况算法流程

2.3.1 输入参数的获取

软件默认界面如图 2 所示,用于本振固定情况时的各种变频模式的杂散计算。在界面中,用户可以自行设定输入信号频段、本振频率、变频方式等等参数。

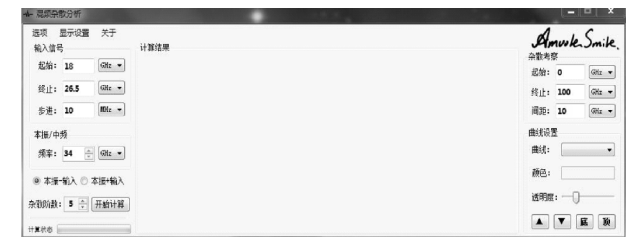


图 2 软件基本界面

开始计算后,软件会先从用户界面获取用户所设定的计算类型以及相应的输入参数等文本值。随后,将这些输入参数的文本值转换为 32 位浮点值并在转换的过程中对参数进行合法性判断,例如,若用户输入参数中包含中文、字母、特殊符号等,则弹出“输入非法”提示。最后,软件利用这些参数值为计算流程中的相关变量进行赋值,并按照预设计算流程进行计算。由于软件的核心计算只涉及基本的代数迭代运算以至于计算量并不大,所以并没有对计算范围进行限制,用户可以根据需求,任意设定所需计算的参数。

默认情况下,软件把杂散信号的幅度根据其阶数从 10 开始依次递减初始化,在这种情况下,计算结果中的幅度仅仅作为一种参考。当然,用户需可以点击

菜单栏“显示设置”中的“幅度设置”选项,然后根据混频器生产商提供的 Datasheet 中的 $M \times N$ 杂散特性在软件中自行进行相应的设置^[14]。



图 3 杂散幅度设置窗口

2.3.2 计算流程

当用户按照图 2 所示进行设定并点击“开始计算”按钮过后,软件先将用户界面所设定的参数缓存到相应的计算变量中。然后根据这些变量展开三层嵌套循环:首先,软件遍历所设定计算阶数中所包含的所有阶数。例如,当用户设定的计算的最高阶数为 5 阶时,软件将 5 阶以下的所有阶数的情况都纳入计算范围。其次,需要遍历当前阶数中的所有 $M \times N$ 阶杂散组合。例如,当计算 4 阶杂散的时候,实际将 $0 \times 4, 1 \times 3, 2 \times 2$ 这些各种频率组合的和差频进行分别计算。最后,软件将输入信号按照所设定的频率步进进行抽样,并且在每个采样点分别计算所生成的杂散频点并赋予其软件预先所设定的幅值。

计算流程大致如图 4 所示。

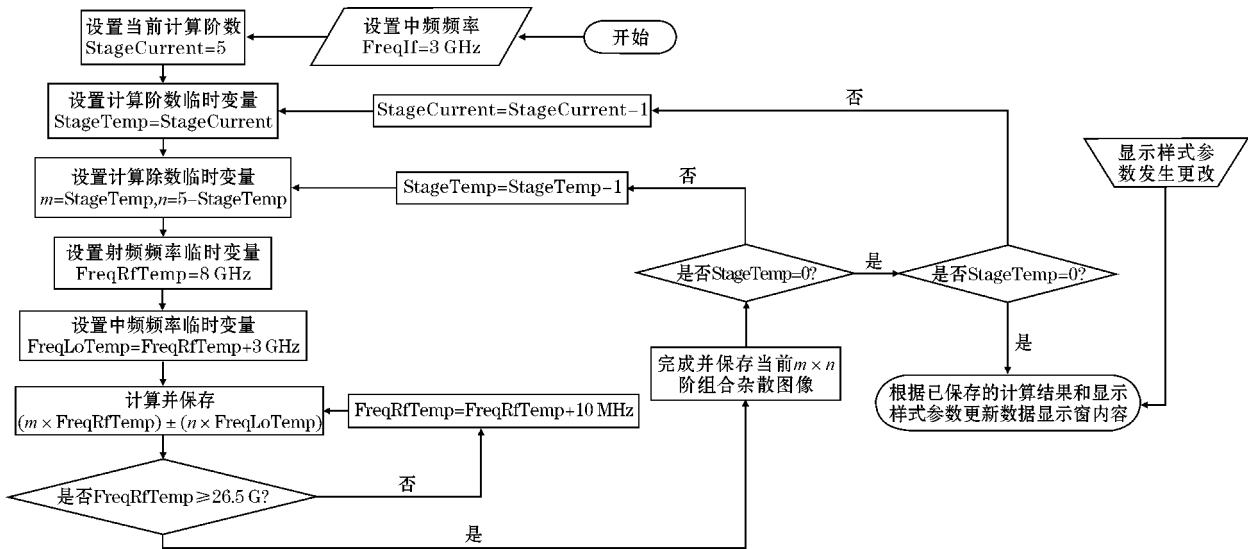


图 4 固定本振求杂散的算法流程

从图 4 可以看出,软件在对杂散进行计算的时候是将输入信号频段以一定的频率分辨率(图中设为 10 MHz)进行采样,然后分别对采用出来的单点进行

差频、和频的计算,最后将计算出来的离散杂散点拟合成一条连续的图像。这样做的缺点是,计算量随着频率分辨率减小而增大;但其好处是算法逻辑相对清晰、

简单,不易出现算法漏洞。

当然,也可以采用纯解析的方法。例如,直接根据输入信号起始频率、终止频率、本振频率这3个参数进行解析计算。这种方法降低很多计算量,但由于需要进行多层逻辑判断,算法逻辑复杂程度更高,并且极易出现算法漏洞,为此摒弃这种方法。

2.3.3 计算结果及显示设置

在计算的过程中,界面左下角的进度条随着计算过程实时更新,最终的计算结果如图5所示。

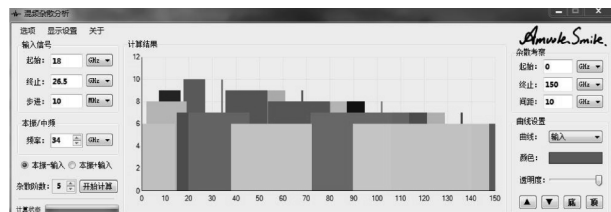


图5 固定本振求杂散的最终计算结果

完成计算后,用户可以点击菜单栏“显示设置”中的“显示项”选项,在弹出的“阶数选择”窗口中(图6),自行选择所需要显示的组合杂散项,并在主窗口右侧的区域对显示方式进行设置。



图6 “阶数选择”设置窗口

为了增加易用性,软件在操作方面进行很多优化。比如,当设置好显示项和显示方式之后,通过调整“本振频率”可以实时地在数据显示框中展现出各个图像随参数变化的运动趋势,为用户提供更直观地参考。

2.4 其他计算类型

除了上述“固定本振”情况下的混频杂散计算外,程序还支持“固定中频”情况下的混频杂散计算。与固定本振的情况不同,在固定中频情况的计算过程中,其本振并非为常量,而是随着输入频率的改变而改变的变量,所以需要在定中频方案的杂散计算步骤增加一级对本振频率更新的步骤,使其本振变量随着每次计算周期不断更新。

需要特别说明的是,在定中频方案下有一个“频分复用”的选项。若未勾选此项,则说明同一时刻只

有一个频道存在信号,其他频道处于无信号状态。软件在此状态下计算混频杂散的时候并不将其他频道产生的杂散考虑在内;若勾选此项,则说明所有频道都同时存在信号^[1,15],此时,软件将整个频带内的所有杂散全部考虑在内。

3 结束语

在射频收发系统的研发过程中,前期混频方案的设计和频谱规划是一项至关重要的工作。软件良好的易用性为方案设计人员提供了极大便利。目前软件已在多个毫米波组件项目的研发中得到实践,其易用的用户界面和直观、实时的数据展示方式获得了一线研发人员一致的好评。其计算结果的准确性也在实践中得到了普遍认可。

软件基于C#语言编写,在编写的过程中充分吸收了面向对象语言的模块化的优点,为后续的功能完善提供了拓展空间。比如,可以增加自动寻找最优中频和混频方案、增加多次变频方案的杂散计算功能等。期间,已经开始对软件后续增加的功能进行了简单规划。相信,随着后续功能的不断更新完善,软件将会在更多地方为设计者提供方便。

致谢:感谢成都市科技惠民技术研发75项目(2015-HM01-00421-SF);成都信息工程大学中青年科研基金(J201401)对本文的资助

参考文献:

- [1] 陈邦媛. 射频通信电路[M]. 北京:科学出版社, 2002.
- [2] 曾兴雯. 高频电子线路[M]. 北京:高等教育出版社, 2003.
- [3] 陈俊喆. 微波宽带低杂散下变频器的研究[D]. 成都:电子科技大学, 2009.
- [4] 舒适. 56M~806M宽频带 Double-Conversion 混频器设计[D]. 上海:复旦大学, 2004.
- [5] 弋稳. 雷达接收机技术雷达接收机技术[M]. 北京:电子工业出版社, 2006.
- [6] Reinhold Ludwig[美]. 射频电路设计——理论与应用[M]. 北京:工业出版社, 2013.
- [7] 黄智伟. 混频器电路设计[M]. 西安:西安电子科技大学出版社, 2009.
- [8] 同济大学数学系. 高等数学[M]. 北京:高等教育出版社, 2007.
- [9] 樊昌信. 通信原理[M]. 哈尔滨:哈尔滨工业大学出版社, 2011.

- [10] KArli Watson, Jon D Reid, Daniel Kemper. C#入门经典[M]. 齐立波, 黄俊伟, 译. 北京: 清华大学出版社, 2008.
- [11] Microsoft Visual Studio 2008 help[Z]. Microsoft Coperation, 2007.
- [12] Nagel. C(美). C#高级编程[M]. 北京: 清华大学出版社, 2008.
- [13] 任宏萍. 面向对象程序设计教程[M]. 北京: 清华大学出版社, 2012.
- [14] HMC689 datasheet [C]. AnalogDevices Company.
- [15] 帅翔. 宽带射频接收技术研究[D]. 成都: 电子科技大学, 2008.

A Mixer Spurious Analyser based on C#

MOU Jun-quan, MIAO Zhi-yi

(College of Electronic Engineering, CUIT, Chengdu 610225, China)

Abstract: In order to eliminate the non-linear and non-ideal isolation characteristics of the mixer and other factors caused by spurious response, we need to process the mixer output signal to filter out the unwanted mixing spurs. While the spurious response in the output signal is closely related to the input and Lo signals we need to plan frequency conversion scheme reasonably to reduce the pressure on the filter in the later circuit. Mixing solution designers urgently need a tool that can predict the spurs generated by the mixer before circuit design, and so do this software. This software is based on .NET Framework and C# language with algorithms based on the generation principle of mixing spurious to calculate the spurious generated by the specified mixer Fin and Flo, meanwhile, real-time display by intuitive chart. We took the software maintainability into account in the process of programming and achieved the software features by using modular method and made it possible to update the follow-up function by reserve a lot of software interface. The software realized the basic functions by giving consideration to the ease of use and maneuverability. Engineers can calculate spurious signals generated in the mixing process before early program planning. It allowed getting the spurious frequency, amplitude and the rolling trends of spurious signals with the frequency changing by adjusting the relevant input parameters. It was said form some Front-line engineers that the software is not only fast but also accurate and eliminated the tedious but not rigorous calculations. The software also provides a trend of spurious changes in the presentation and made the result more intuitive and humane as opposed to some existing spurious analysis tools, that may after all be accepted as an innovation.

Key words: electromagnetic field and microwave technology; microwave millimeter wave circuit; frequency conversion; mixer spurious; C#